

学術機関リポジトリ構築ソフトウェア  
実装実験プロジェクト  
報告書

平成 17 年 3 月

国立情報学研究所

## 序

平成16年6月から平成17年3月にかけて実施した「学術機関リポジトリ構築ソフトウェア実装実験プロジェクト」の報告書をお届けします。

インターネット利用の普及に伴い、オープンアクセス思潮の興隆をはじめとして、印刷体学術雑誌の出版流通を軸としてきた学術情報流通は大きな流動と変革の時期を迎えています。とくに、電子的情報流通環境下にあっては、電子的学術情報資源は、従来の出版流通では取り扱われなかったような多様な研究教育活動の成果や研究教育に必要な情報資源の発信、流通、横断的活用について多くの可能性が広がっています。その一方で、そのような多様な電子的学術情報資源は、現在の出版流通メカニズムの中では、ともすると散逸しやすく、電子的形態でのみ存在する情報資源の将来にわたる保存と利用という面でも問題が十分に明らかになっていません。したがって、このような多様な電子化された学術情報資源は、大学等の学術機関が、学術機関リポジトリを構築し、その発生源において責任を持ってアーカイブし、必要十分なメタデータとともに世界に、そしてまた後世に伝えていくことがますます重要になってきています。

このような意味で学術機関リポジトリはこれからの学術コミュニケーションの一角を担うものです。他方、学術機関リポジトリの構築は、従来の図書館活動の枠を超えて、大学などの学術機関内での組織作りという新たな側面を含んでいます。本プロジェクトは、学術機関リポジトリ構築ソフトウェアの実装実験を行なうとともに、組織作りという面でも事例の蓄積も進めてまいりました。本報告書が、国内各大学・研究機関における学術機関リポジトリ設立の検討の一助となれば幸いです。

最後に、本プロジェクトに参加を頂いた、

北海道大学  
千葉大学  
東京大学  
東京学芸大学  
名古屋大学  
九州大学

の御協力に感謝いたします。

平成17年3月  
国立情報学研究所教授  
神門典子

## 目次

はじめに.....	3
本報告書の構成.....	10
参加機関.....	11
活動記録.....	12
<国内関連プロジェクトから> デジタルコンテンツ・プロジェクト.....	15
<国内関連プロジェクトから> REFORM（電子情報環境下における 大学図書館機能の再検討）.....	16
講演資料「SPARC の理念と機関リポジトリ／土屋俊・千葉大学文学部」 （NII-IRP ワークショップ, 2004.6）.....	17
講演資料「Institutional Repository の 構築と運用／尾城孝一・千葉大学附属図書館」 （NII-IRP ワークショップ, 2004.6）.....	19
講演資料「機関リポジトリの品質保証／土屋俊・千葉大学文学部」 （中間報告会, 2004.9）.....	24
講演資料「なぜ大学は機関リポジトリをもたねばならないか／土屋俊・千葉大学文学部」 （報告会, 2005.2）.....	26
<b>第 1 部 システム構築.....</b>	<b>29</b>
第 1 章 システム構築.....	30
第 2 章 DSpace の導入と設定.....	33
第 3 章 EPrints の導入と設定.....	64
<b>第 2 部 運用上の諸課題.....</b>	<b>115</b>
第 1 章 学内合意形成と運用体制.....	116
第 2 章 コンテンツ収集・投稿促進.....	122
第 3 章 著作権処理.....	129
第 4 章 学内同種事業との連携.....	136
第 5 章 メタデータ.....	139
<b>第 3 部 各大学の状況.....</b>	<b>148</b>
第 1 章 北海道大学.....	149
第 2 章 千葉大学.....	152
第 3 章 東京大学.....	157
第 4 章 東京学芸大学.....	160
第 5 章 名古屋大学.....	163
第 6 章 九州大学.....	167
第 7 章 金沢大学.....	171

附録 国内対応メーカー・サービス一覧 .....	174
株式会社シー・エム・エス .....	175
インフォコム株式会社 .....	176
ユサコ株式会社 .....	177
富士通株式会社 .....	178
株式会社NTTデータ九州 .....	179
株式会社シー・エム・エス .....	180

## はじめに

### 本報告書について

本報告書は、平成16年6月から平成17年3月にかけて国立情報学研究所と国内の大学図書館が共同で実施した「学術機関リポジトリ構築ソフトウェア実装実験プロジェクト」について報告するものである。

同プロジェクトは国内における学術機関リポジトリ構築先行事例の研究と、オープンソースの学術機関リポジトリ構築用ソフトウェアの稼働実験を通じて、その構築と運用に関するノウハウを蓄積することにより、知識の共有化を図り、国内各大学での円滑な学術機関リポジトリ構築・運用の一助とすることを目的として行われた。プロジェクトに参加した機関・組織を以下に示す。

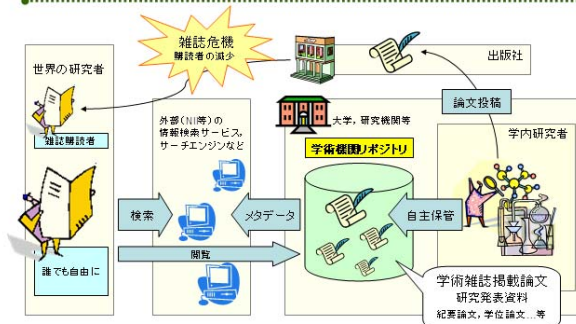
北海道大学	附属図書館
千葉大学	附属図書館
東京大学	情報基盤センター
東京学芸大学	学術情報部
名古屋大学	附属図書館，情報連携基盤センター
九州大学	附属図書館，情報基盤センター

### 学術機関リポジトリとは

学術機関リポジトリ (Institutional Repository) は、大学等の学術機関 (以下、「大学」という。) で生産された知的生産物を保存・公開することを目的とした、電子アーカイブシステムである。学術機関リポジトリの対象としては、学術雑誌掲載論文、プレプリント、科学研究費補助金成果報告書、テクニカルペーパー、学会発表スライド、紀要論文、学位論文、電子教材、データセット、マルチメディア・コンテンツなど、さまざまな電子コンテンツの搭載が考えられる。

大学の研究成果等を扱うデータベース・システムとしては、研究業績を集約した二次情報データベースや、紀要・学位論文・図書館所蔵資料を電子化・収載した電子図書館システムなどがある。学術機関リポジトリは、一次コンテンツ本体を搭載し、その保存とアクセス保証を担う点で前者と異なり、収録対象を学内刊行物に限定せず、雑誌掲載論文を含め教育・研究の成果物を広く扱う点で後者とも一線を画する。また、紙媒体資料のスキヤニングに大きく依存することなく、ボーン・デジタルな資料を視野の中心に据える点も特徴である。

## 学術機関リポジトリの概要



学術機関リポジトリの概要 (NII Library Week 2004 スライド)

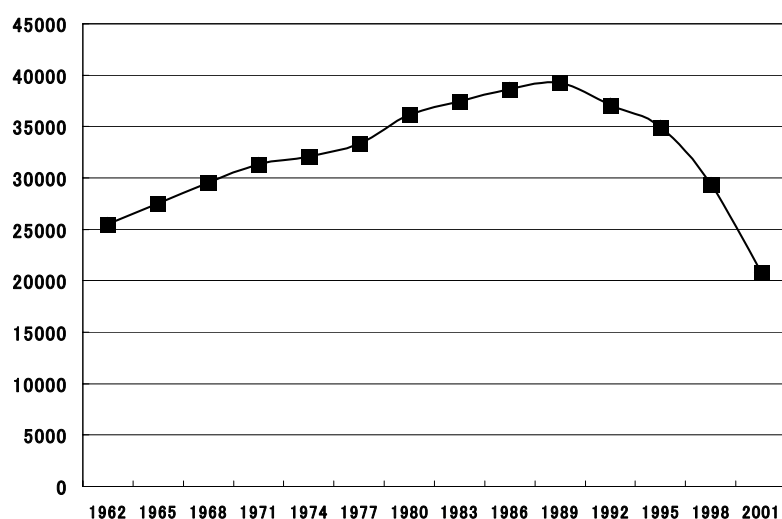
学術機関リポジトリ設立の目的や用途としては、研究成果の可視性向上、大学の電子資産の恒久的保管、教育活動のための電子教材の配布、研究紀要の純電子ジャーナル化のインフラ等、多様な可能性が考えられる。以下では、図書館の重要な任務のひとつでもある学術情報基盤の整備と学術コミュニケーション支援の観点から、研究成果の可視性向上を目的としたセルフ・アーカイビングの基盤としての側面を中心に、学術機関リポジトリの背景、意義について示す。

### 学術機関リポジトリの背景～学術コミュニケーション不全

学術研究活動の成果は主として学術論文として結実する。研究者は他の研究者の論文に示された新たな知見を自分の研究活動に役立て、その成果として新たな学術論文を生み出す。この情報流通サイクルを追って、総体として研究が進展する。これが学術コミュニケーションの基本的な流れである。

印刷出版時代を通じてこの学術情報流通サイクルを支えてきたのは学術雑誌であった。学術雑誌は、17世紀半ばに誕生して以来今日まで、専門家による査読 (peer review) によって品質が保証された学術論文を安定的に学術コミュニティに還元する媒体として、学術研究成果の公表のための重要な位置を占めてきた。

20世紀中盤以降、学術研究活動のスピードと量は急激に増加し、学問分野は多様化と深化の途をたどった。科学技術、生命科学 (STM: Science, Technology and Medicine) の分野を中心に、研究者によって生産される論文の数は増加し、学術雑誌はそのタイトルと刊行頻度を増やしていった。出版社同士の併合が相次ぎ、徐々に学術出版市場の寡占化が進行するとともに、販売部数の減少とさらなる価格上昇の悪循環の中で、雑誌危機 (Serials Crisis) と呼ばれる状況が現出した。



この図は、我が国の大学等における外国学術雑誌の総受入タイトル数の推移を示したグラフである。ピークであった1980年代後半には約4万タイトルに達していた受入タイトル数であるが、購読中止による減少が続き、以降わずか10年程度の間半減してしまっていることがわかる。

この状況に更なる一石を投じたのがインターネット利用の普及である。学術論文は電子ファイルの形で流通可能となり、学術出版社はアクセス対価と引き換えにオンラインで論文を閲覧できるサービスを開始した。電子ジャーナルの成立は一括契約の傾向を助長し、Big Dealと呼ばれる大規模パッケージ契約の購読形態を生み出した。タイトルごとの取捨選択に基づいていた大学等研究機関の学術雑誌購入スタイルは一部崩れ、出版社単位のオール・オア・ナッシングの判断を迫られるケースも増加した。

### オープン・アクセス思潮

雑誌危機には、各館にとっての収集資料の減少、すなわち、研究者にとって研究活動に必要な学術情報資源を十分に得られないという側面のほかに、もうひとつ重要な意味がある。すなわち、このように雑誌が行き渡らない状況下にあっては、研究成果の流通が限定されるために、自身が発表する学術論文も望みうる十分な読者を得られないという事態を意味するのである。

こうした状況からオープン・アクセスという思潮が生まれた。オープン・アクセスとは、学術研究の成果へ誰もが障壁なくアクセスできるようにすることを意味し、研究者にとって、論文の読み手としては研究資源の増大、書き手としてはより広範な読者獲得による研究インパクト向上のメリットがある。

オープン・アクセスの実現には二通りの主な戦略が提案されている。

- ・ 無料雑誌（オープン・アクセス・ジャーナル）  
無料オンライン公開される学術雑誌を研究成果公表の場とするもの。
- ・ 自主保管（セルフ・アーカイビング）  
論文著者が自身乃至所属機関の Web サイト（＝学術機関リポジトリ）等において自著論文を無料オンライン公開するもの。

オープン・アクセスの効果についてはまだ完全に実証がなされているわけではないが、計算機科学分野においては、オープン・アクセス論文はそうでない論文に比べ 2.6 倍の被引用率を持つとの報告がなされており<sup>1</sup>、計算機科学以外の学問分野についても現在同様の検証作業が進められている。これについて、オープン・アクセス活動の推進者であるスティーブン・ハーナッドは、「所属機関が論文を掲載した雑誌のアクセス費を捻出できないためほかの方法では論文にアクセスできないユーザを取り込むことで、オープン・アクセスは論文の潜在ユーザの数を劇的に増加させている」と述べている<sup>2</sup>。

学術情報流通の一方の要である学術出版社のオープン・アクセスへの対応についても SHERPA/Romeo プロジェクトが継続的な調査を行っている。平成 17 年 1 月現在、エルゼビア社を含む 92%の海外学術出版社が著者による自主保管を許容しており<sup>3</sup>、学術出版界においても一定の理解が広まっていると見ることができる。一方、国内については、千葉大学による 39 の学協会を対象とした調査によれば 42%の学協会が自主保管を認めている<sup>4</sup>。

また、論文の著者である研究者自身については、JISC/OSI による英国の状況の調査レポート<sup>5</sup>では、大半の著者が自主保管に賛同するという調査結果が示されており、北海道大学による学内研究者へのアンケート調査でも回答が得られた研究者の 70%が自主保管に賛意を示していることがわかった<sup>6</sup>。

以上のような状況に鑑み、ハーナッドは「ボールは投げ返された（It is hence clear that the ball is now in the universities' court）」、すなわち、自主保管戦略に基づいてオープン

---

<sup>1</sup> Steve Lawrence. "Online or Invisible?"

<http://citeseer.ist.psu.edu/online-nature01/>

<sup>2</sup> Stevan Harnad. "Comparing the Impact of Open Access (OA) vs. Non-OA Articles in the Same Journals"

<http://www.dlib.org/dlib/june04/harnad/06harnad.html>

<http://www.nii.ac.jp/metadata/irp/harnad/>（日本語）

<sup>3</sup> SHERPA. Publisher copyright policies & self-archiving

<http://www.sherpa.ac.uk/romeo.php>

<sup>4</sup> 千葉大学附属図書館. 国内学会等刊行誌掲載論文の著作権調査について（報告）

[http://mitizane.ll.chiba-u.jp/curator/about/local\\_societies\\_research.pdf](http://mitizane.ll.chiba-u.jp/curator/about/local_societies_research.pdf)

<sup>5</sup> JISC/OSI. "Journal Authors Survey Report"

[http://www.jisc.ac.uk/uploaded\\_documents/JISCOAreport1.pdf](http://www.jisc.ac.uk/uploaded_documents/JISCOAreport1.pdf)

<sup>6</sup> 北海道大学附属図書館. 学術情報の発信に関するアンケート調査（集計）

<http://www.lib.hokudai.ac.jp/item/enq/kekka.html>



ン・アクセスを推進する上で次は大学が行動する番である、と指摘している。

## 学術機関リポジトリの狙い

学術機関リポジトリの狙いは次の2点に集約される。

- ・ 研究インパクトの回復・維持・向上（研究者にとってのメリット）
- ・ 学術機関としての研究成果コレクションの構築（機関にとってのメリット）

学術機関リポジトリは、大学を基盤としたオープン・アクセス推進のインフラストラクチャとなるものである。研究者は、学術機関リポジトリに自著論文を掲載することにより、より多くの読者を獲得できる（必ずしも商業出版社の有料学術雑誌への論文投稿を中止し、専ら独自に発信していくということを意味するものではなく、むしろ経済的理由などで論文掲載誌を購読できない研究者へも自著論文を届けることが主眼である）。

一方、学術機関リポジトリを運営することによって、大学には研究成果の電子コレクションが自然に形成されていくことになる。成長した学術機関リポジトリは、学術研究活動の側面における大学の「顔」として、既存の大学ホームページと並び、学術研究活動の成果を社会へ還元する窓口の役割を果たすことができる。

## 図書館の役割

学術機関リポジトリの設立と運営には、研究者を含めた大学全体としての協調的連携が不可欠であるが、海外では大学図書館がその主導的な役割を果たしている例が多い。

我が国でも、国立大学図書館協議会学術情報委員会デジタルコンテンツ・プロジェクトによる国立大学附属図書館を対象とした「デジタルコンテンツに関するアンケート」によれば、科研費報告書は82%、博士學位論文は72%、紀要は88%の図書館が収集を行っており、大学図書館は学術機関リポジトリを担うにふさわしい学術情報流通支援の実績とポテンシャルを有しているといえる。

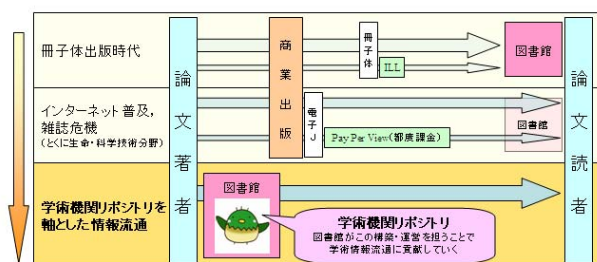
電子論文の収集・蓄積は、資料収集という意味において従来の図書館活動の延長として位置づけられる。また、学術機関リポジトリには、著作権処理をはじめとして、これまで図書館が蓄えてきた知識と経験が必要とされる側面が少なくない。

さらにもっとも重要な点として、情報の出し手である論文著者と情報の受け手である論文読者を結ぶ仲介者の役割は、資料の収集とサービスを通じて図書館が元来果たしてきた任務であり、電子的情報流通環境化においては、情報発信者の側から学術コミュニケーションに一定の貢献を果たしていくことは今後の図書館活動の重要な一角を占め得る。

とくに、学術機関リポジトリの構築と運営に主体的に取り組むことにより、パッケージ購入の電子ジャーナルを主体とした現代の学術情報流通においてやや希薄化し不明瞭となった図書館の存在意義を明確に学内に示し、図書館自身の可視性を向上していくことも重要と考えられる。

## なぜ図書館が担うのか

### • 学術コミュニケーションへの寄与



なぜ図書館が担うのか (NII Library Week 2004 スライド)

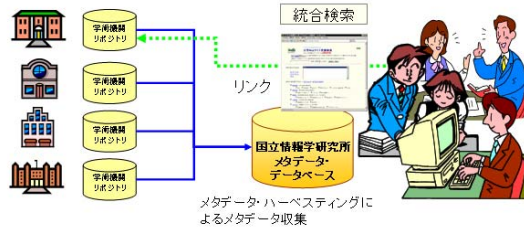
## 国立情報学研究所の役割

平成 14 年 10 月、大学からの情報発信支援を重要な目的のひとつとして、国立情報学研究所はメタデータ・データベース共同構築事業を開始した。同事業は、各大学の研究者等がネットワーク上に公開している学術情報を対象とした共同分担入力方式によるデータベース構築事業であるが、ネットワーク上の情報資源は安定度が低く、リンク切れ等により、常に継続的なメンテナンスが必要となっている。学術機関リポジトリは、大学の研究成果の固定・保存におけるこうした課題を解決するものでもある。

メタデータ・データベースは、自動的なメタデータ流通のための仕組みである OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting) に準拠したメタデータ収穫ソフトウェアを備え、学術機関リポジトリとの全自動相互運用が可能となっている。このシステム間連携により、学術機関リポジトリに収録された電子論文のメタデータは自動的にメタデータ・データベースへ集約され、メタデータ・データベースは国内各学術機関リポジトリへのポータル機能を担うことができる。

## 国立情報学研究所 公開への支援

- メタデータ・データベースを通じた、国内学術機関リポジトリへの統合的アクセスの実現



### 国立情報学研究所 公開への支援 (NII Library Week 2004 スライド)

本研究所は、メタデータデータベース共同構築事業における学術情報資源記述の方式として、ダブリンコア・メタデータ記述要素応用プロファイルである「NIIメタデータ記述要素」を提案している。OAI-PMHを通じたシステム間連携のために学術機関リポジトリは同プロファイルへの対応を図る必要があるが、DSpace及びEPrintsにおける方法についても本報告書を参照されたい。

## 本報告書の構成

第 1 部では、学術機関リポジトリを構築するためのオープンソース・ソフトウェアとして本プロジェクトにおいて各大学が実装実験を行った DSpace 及び GNU EPrints について、その導入・設定手順、日本語対応手順、NII メタデータ記述要素への対応手順について記す。

第 2 部では、学術機関リポジトリの運営上の課題について、海外先行事例などを参考に考察を加える。

第 3 部では、プロジェクトに参加した各大学における学術機関リポジトリ構築への取り組みの現況について紹介する。併せて、ほかに学術機関リポジトリ構築の検討をすすめている金沢大学の事例についても紹介する。

巻末に、学術機関リポジトリの構築に利用可能な国内商用ソフトウェアや、海外オープンソースの導入支援業務を扱うシステム開発メーカ等についての資料を付す。

第 1 部～第 3 部の各章はそれぞれ下記参加機関が分担執筆し、国立情報学研究所が監修した。

(執筆担当)

第 1 部	第 1 章	システム構築	九州大学
	第 2 章	DSpace の導入と設立	名古屋大学
	第 3 章	Eprints の導入と設立	東京大学
第 2 部	第 1 章	学内合意形成と運用体制	千葉大学
	第 2 章	コンテンツ収集・投稿促進	千葉大学
	第 3 章	著作権処理	千葉大学
	第 4 章	学内同種事業との連携	北海道大学
	第 5 章	メタデータ	東京学芸大学
第 3 部			各大学

## 参加機関

### 北海道大学

金子敏 附属図書館情報システム課システム管理係長  
鈴木雅子 附属図書館情報システム課システム管理係

### 千葉大学

加藤晃一 附属図書館情報管理課雑誌・電子情報係長  
阿藤品治夫 附属図書館情報管理課雑誌・電子情報係

### 東京大学

本多玄 情報基盤センターデジタル・ライブラリ係長  
小山憲司 情報基盤センターデジタル・ライブラリ係

### 東京学芸大学

綾部輝幸 学術情報部情報管理課電子情報係長  
山崎みどり 学術情報部情報管理課電子情報係

### 名古屋大学

郡司久 附属図書館情報システム課長  
山本哲也 情報連携基盤センター学術電子情報掛

### 九州大学

瓜生照久 情報基盤センター電子図書館掛長  
小野真由美 附属図書館情報システム課電子情報掛

### 国立情報学研究所

神門典子 ソフトウェア研究系教授  
宮澤彰 人間・社会情報研究系教授  
大場高志 開発・事業部コンテンツ課長  
山西秀幸 開発・事業部コンテンツ課課長補佐  
杉田茂樹 開発・事業部コンテンツ課学術情報形成第二係長

## 活動記録

(詳細はプロジェクト・ホームページ <http://www.nii.ac.jp/metadata/irp/> を参照のこと)

NII-IRP ワークショップ (平成 16 年 6 月 14～19 日 NII, 軽井沢国際高等セミナーハウス)

- ・ 各大学の現況についての情報交換, 先行事例報告
- ・ DSpace, GNU EPrints の実験機へのインストール
- ・ 日本語対応, NII メタデータ記述要素対応手順の分析調査

参加者: プロジェクト参加機関

株式会社シー・エム・エス

株式会社インフォコム

講師: 土屋俊 (千葉大学文学部)

尾城孝一 (千葉大学附属図書館)

鈴木敬二 (北海道江別市)

鳥越直寿・日高正志 (株式会社インフォコム)

中間報告会 (平成 16 年 9 月 7 日 NII)

- ・ 各大学の状況報告
- ・ 国立大学図書館協会学術情報委員会デジタルコンテンツ・プロジェクト及び研究グループ「REFORM」(研究代表者: 千葉大土屋教授。大学図書館の情報発信機能について研究課題とする) との情報交換・意見交換

参加者: プロジェクト参加機関

荘司雅之 (早稲田大学図書館)

研究グループ「REFORM: 電子情報環境下における大学図書館機能の再検討」

土屋俊 (千葉大学文学部)

栗山正光 (常磐大学人間科学部)

逸村裕 (名古屋大学大学院情報科学研究科)

国立大学図書館協会学術情報委員会デジタルコンテンツ・プロジェクト

米澤誠 (東北大学附属図書館総務課情報企画係長)

片山俊治 (群馬大学附属図書館情報サービス課長)

木村晴茂 (岐阜大学附属図書館情報サービス課長)

郡司久 (名古屋大学附属図書館情報システム課長)

富田健市 (筑波大学附属図書館情報サービス課長)

篠塚富士男（筑波大学附属図書館情報管理課課長補佐）

講師：土屋俊（千葉大学文学部）

#### 各参加機関での試験構築

- ・ 東京大学（GNU EPrints）との間のメタデータ・ハーベスティング実験（10月）
- ・ 東京学芸大学（DSpace）との間のメタデータ・ハーベスティング実験（11月）
- ・ 北海道大学（DSpace）との間のメタデータ・ハーベスティング実験（12月）

SPARC&SPARC Europe “Institutional Repository：The Next Stage” 出展（平成16年11月18～19日 ワシントンD.C.）

- ・ 本プロジェクトの概要を紹介するポスターを出展

スティーブン・ハーナッド氏との懇談会（平成16年11月24～25日 NII, パシフィコ横浜）

- ・ オープン・アクセス、学術機関リポジトリについて、研究グループ「REFORM」と合同で懇談会を実施（24日）。スティーブン・ハーナッド氏は、ケベック大学カナダ政府基金講座教授兼サウサンプトン大学教授で、オープン・アクセス運動の積極的な推進者。
- ・ 第6回図書館総合展（パシフィコ横浜）『学術コミュニケーション最先端：オープン・アクセスとセルフアーカイブ』（講師：スティーブン・ハーナッド氏）聴講（25日）。

参加者：プロジェクト参加機関

荘司雅之（早稲田大学図書館）

研究グループ「REFORM：電子情報環境下における大学図書館機能の再検討」

土屋俊（千葉大学文学部）

竹内比呂也（千葉大学文学部）

松村多美子（図書館情報大学（名誉教授））

栗山正光（常磐大学人間科学部）

佐藤義則（三重大学人文学部）

逸村裕（名古屋大学大学院情報科学研究科）

尾城孝一（千葉大学附属図書館）

酒井由紀子（慶應義塾大学医学メディアセンター）

細川聖二（国立情報学研究所開発・事業部コンテンツ課学術コミュニケーション係長）

講師：土屋俊（千葉大学文学部）

NII Library Week 2004 出展 (平成 16 年 12 月 7～17 日 札幌, 東京, 名古屋, 京都, 福岡)

- ・ スライド「学術機関リポジトリ : Institutional Repository」を出展

報告会 (平成 17 年 2 月 10 日 NII)

- ・ 各大学の状況報告
- ・ 報告書に関する意見交換

参加者 : プロジェクト参加機関

内島秀樹 (金沢大学情報部情報企画課課長補佐)

守本瞬 (金沢大学情報部図書館サービス課医学情報サービス係)

荘司雅之 (早稲田大学図書館)

今村昭一 (早稲田大学図書館)

研究グループ「REFORM : 電子情報環境下における大学図書館機能の再検討」

土屋俊 (千葉大学文学部)

松村多美子 (図書館情報大学 (名誉教授))

酒井由紀子 (慶應義塾大学医学メディアセンター)

国立大学図書館協会学術情報委員会デジタルコンテンツ・プロジェクト

西原清一 (筑波大学システム工学研究科教授)

米澤誠 (東北大学附属図書館総務課情報企画係長)

郡司久 (名古屋大学附属図書館情報システム課長)

富田健市 (筑波大学附属図書館情報サービス課長)

篠塚富士男 (筑波大学附属図書館情報管理課課長補佐)

国際学術情報流通基盤整備事業

永井裕子 (社団法人日本動物学会事務局長)

細川聖二 (国立情報学研究所開発・事業部コンテンツ課学術コミュニケーション係長)

ユサコ株式会社

日本ヒューレット・パッカー株式会社

講師 : 土屋俊 (千葉大学文学部)



<国内関連プロジェクトから>

## デジタルコンテンツ・プロジェクト

主査：植松貞夫（筑波大学附属図書館長）

デジタルコンテンツ・プロジェクトは、国立大学図書館協会（以下「国大図協」）第51回総会で設置された学術情報委員会の小委員会として発足しました。委員は主査以下7名で、オブザーバー1名と事務局2名で活動しています。活動内容としては、学術機関リポジトリ及び電子 Book に関するものが中心ですが、ここでは前者についてだけ紹介させていただきます。

親委員会である学術情報委員会は平成16年7月1日の総会で設置されましたが、小委員会である本プロジェクトの組織及び任務が確定し、国大図協に報告されたのは7月27日でした。したがって、発足からほぼ半年が経過したことになります。当初の学術機関リポジトリに関する任務は、「学術機関リポジトリの実務レベルの検討」であり、その内容としては、「大学における学術機関リポジトリ・ソフトウェアの実装」及び「具体的なコンテンツの導入整備」が挙げられていました。

8月に入ってメーリングリストを立ち上げ、活動内容等についての検討を行っていましたが、9月7日に開催された NII-IRP の中間報告会に本プロジェクトのメンバーも参加させていただいた際、ソフトウェアの実装については多くの部分で重複することが判明しました。そこで、実装についての検討は必要最小限とすることにし、本プロジェクトでは啓発活動や管理方針の策定に力点を置くこととしました。

具体的活動としては、平成16年12月に国大図協のメンバー館に対して、アンケート調査を実施しました。これは、学術機関リポジトリは学術情報の収集と公開に実績のある図書館が担当するのが相応しいといわれていることを具体的な数字で裏づけることと、各館への啓発活動の一環として行ったものです。平成17年2月14日までに88機関から回答がありましたが、科研費報告書は82%、博士学位論文は72%、紀要は88%の機関が収集しており、学術機関リポジトリを担うにふさわしい学術情報流通支援の実績とポテンシャルを有しているということができると思います。

ただし、ポストプリント、プレプリントについては、これからの課題であることがはっきりとしました。啓発活動の面からは、学術機関リポジトリへの対応状況が、すでに運用（1）、試験運用（6）、計画あり（4）、策定中（31）、計画なし（43）、無回答（3）となりました。計画なしのうち、半数以上はこれから検討を行うということであり、いくつかの問題点がクリアされれば運用が本格化していくものと期待されます。

今後の活動としては、現在中間報告書のまとめにかかっており、その中でアンケートに現れた実施にあたっての問題点を洗い出し、それを克服していくことを計画しています。主な問題点として、予算・人員・研究者の理解／協力・著作権処理・学内合意等がすでに挙がっていますが、それらに対してどのような方策があるのかを検討していきたいと考えています。

本プロジェクトは、1年では終了せず継続される可能性が高いと考えていますので、NII-IRP の成果も取り込ませていただきながら、啓発活動を続けていきたいと思っています。今後とも皆様のご支援とご協力をお願いいたします。

<国内関連プロジェクトから>

## REFORM（電子情報環境下における大学図書館機能の再検討）

学術情報発信班主査 栗山正光  
（常磐大学人間科学部助教授）

REFORM(Reengineering of the Functionalities of Research Libraries in the Digital Milieu)は、平成16～18年度科学研究費補助金基盤研究(B)による研究グループです。研究代表者は土屋俊教授（千葉大学文学部）で、他に研究分担者7名、主要研究協力者7名で構成されています。研究の目的は、電子情報が学術コミュニケーションを大きく変化させつつある時代にあって大学図書館がどのような機能を果たすべきか、望ましい基準やガイドライン、あるいはマニュアルといった具体的な形で、提言を行うことです。

この目的に向けて、REFORMでは「マネジメント」、「情報サービス利用」、「情報発信」の三つの研究班を設置しました。研究メンバーはこの三つの研究班のいずれかに属しています。もちろん、これらは互いに関連しあう研究であり、メンバーも時には自分の属する班を越えて活動に参加しています。

最初のマネジメント班では、大学図書館における学術情報マネジメント機能についての実証的研究を行っています。手始めに、わが国の学術情報政策に関わる資料の収集・整理や、関係者への聞き取り調査を開始しています。

次の情報サービス利用班では、大学図書館における電子情報サービス利用についての実証的研究を行っています。具体的には、電子ジャーナルやILLの利用統計の解析を通して、電子情報サービス利用の実態に迫りたいと考えています。

そして、情報発信班では、大学図書館における学術情報発信についての基礎的研究を行っています。ここでは特に学術機関リポジトリに注目し、先進事例や文献の調査を通して、その意義や出現の背景を探っています。学術機関リポジトリは、学術コミュニケーションの新しい担い手として、世界各国の大学図書館がその構築に乗り出しています。日本においても、90年代に開始された電子図書館プロジェクトは学術機関リポジトリ的な要素を持っており、その経験やコンテンツを新しいシステムで生かすことは十分可能であると考えられます。

この情報発信班の研究は「学術機関リポジトリ構築ソフトウェア実装実験プロジェクト(NII-IRP)」と特に密接に関わっていると言えます。NII-IRPでは、全国の各大学図書館で実際にシステムの構築に携わっていらっしゃる方々の知識・情報を持ち寄ることで、実戦的なノウハウが蓄積されています。これに対して、REFORMは、図書館が大学の研究成果を電子的に収集・発信することの意味や妥当性（あるいは必然性）を明らかにすることで、各大学における学術機関リポジトリの構築・運用を側面からサポートしていきたいと考えています。少々乱暴に図式化すれば、REFORMが理論、NII-IRPが実践という形で車の両輪となって、わが国の大学図書館による学術機関リポジトリの推進に資する、ということになるでしょうか。

幸い、NII-IRPのメンバーの方にも、研究協力者としてREFORMに加わっていただいています。両プロジェクトの連携・協力を密にし、相乗効果で成果をあげることができると期待しております。

講演資料「SPARCの理念と機関リポジトリ／土屋俊・千葉大学文学部」  
(NII-IRP ワークショップ, 2004.6)

**SPARCの理念と機関リポジトリ**

土屋俊(千葉大学)

図 1

**SPARC理念の変質、  
または、1998年の皮肉**

- もともとのSPARCはPrint雑誌ベース
  - 商業出版社は、競争市場を破壊している
  - 競争を再現すべきである(80年代モデル?)
  - 学会誌によって競争を実現しよう
- 1998年: 学術コミュニケーションの電子化元年
- 現在のSPARCは、E-Journalベース
  - オープン・アクセス(開放的利用)
  - 機関リポジトリ

研究者と図書館の役割の強調は一貫

図 2

**SPARCの現状認識と提案**

- 電子ジャーナル化によって、
  - 研究成果の量的増大は依然として続くものの
  - 開放利用への需要が増大
  - 従来型価格・市場モデルへの不満(特に図書館)
  - 学術成果の「アーカイブ」に関する不安
- これに対応するためには、機関リポジトリで
  - 図書館は、advocacyだけでなく、自分でリポジトリ運営を → **新しい学術出版モデルの創出**
  - 研究者は、学会よりも所属機関を重視を → **機関の知名度と威信の向上**

図 3

**新しい学術出版モデルとは**

- 学術コミュニケーション4機能の分散実施
  - 従来は、「垂直統合」であった

従来型 出版社 図書館	雑誌への投稿	IR型 機関
	ピア・レビュー	
	選書・予約購読	IR搭載
	製本・配架・管理	機関保証 IRを基礎 学術機関が実施

- 登録 (registration)
- 承認 (certification)
- 認知 (awareness)
- 保存 (archiving)

図 4

**IR型学術情報流通のモデル**

<b>登録</b> 別途開拓する必要	<b>保証</b> 学会保証でなく 機関保証? (overlay journalの 可能性ーただし、 夢)	<b>認知</b> リンク 抄録 Overlay Journal etc
相互運用可能		
<b>サービス層</b>	<b>コンテンツ層</b>	
蓄積・保存 相互運用可能 → メタデータ共有・メタデータ収集協力 共同運用 長期保存 → 図書館間協力		

図 5

**機関リポジトリの4要件**

- 機関における明確な位置づけ
  - すべての機能について必要
- 学術資料への限定
  - 当面はここが大事だから
- 蓄積的かつ恒久的
  - 機関が責任を(限定的であれ)もつ
- 開放的かつ相互運用可能
  - アクセスへのバリアを下げるために協力が必要

図 6

## 影響の評価とコスト負担モデル

- ステークホルダーの増加
  - 研究者: それなりの作業、必要な自覚
  - 学会: いろいろできるが、蜜月は終わったかも
  - 大学等: 責任とおそらく資金の主体
  - 図書館: 実施の主体
  - 出版社: 本当はやることなくはるはず(しかし、エルゼビア社の緑化の問題提起は?)
  - 資金提供団体: コミュニケーションへの直接支援が必要
- 価格モデルからコスト負担モデルへの概念的变化

図 7

## ばら色よりも自覚の喚起

- 金が出続けるだろうか？
  - Digital divide 激化も覚悟の上(「一流大学」だけのエリート・オープン・アクセス?)
- 研究者・図書館員は「自覚」のみで仕事を続けるだろうか？
  - 70年代以降の商業化は反対の示唆
- 社会は好むだろうか？
  - どこまで信用してもらえるかという問題が残る。

図 8

講演資料「Institutional Repositoryの構築と運用／尾城孝一・千葉大学附属図書館」（NII-IRP ワークショップ, 2004. 6）

**Institutional Repositoryの  
構築と運用**  
 千葉大学附属図書館  
 尾城 孝一  
 ojiro@ll.chiba-u.ac.jp

2004/6/14      NII-IRPワークショップ      1

図 1

**アウトライン**

- 概説
  - Institutional Repositoryとは
  - 機関リポジトリの現状
  - 存在意義
  - 代表的なリポジトリ, プロジェクトの紹介
- 構築と運用の課題
  - 学内合意形成
  - 運用方針の策定
  - システム実装
  - 登録の促進
- 今後の展望

2004/6/14      NII-IRPワークショップ      2

図 2

**Institutional Repositoryとは**

- 日本語では
  - 学術機関リポジトリ, 機関リポジトリ, 機関レポジトリ
- 定義
  - 電子コンテンツを捕捉し, 保存し, 発信するための, ウェブベースのデータベース
- 電子コンテンツの要件
  - 学術的価値
  - ある学術機関において生産されたコンテンツ(vs. 主題ベースのリポジトリ)
  - 累積的かつ永続的
  - オープン(誰もが無償でアクセス可能)かつ相互運用可能

2004/6/14      NII-IRPワークショップ      3

図 3

**存在意義**

- SPARC
  - "The Case for Institutional Repositories: A SPARC Position Paper" (2002)
  - 学術コミュニケーション・システムの改革
    - 世界的なリポジトリのネットワーク
    - 無料でアクセスできる学術論文の増加
    - 商業出版社による寡占システムの代替
- Clifford A. Lynch
  - Institutional Repositories: Essential Infrastructure for Scholarship in the Digital Age (2003)
  - デジタル時代における学術研究に不可欠のインフラストラクチャ
    - 大学とその構成員によって創造されたデジタル情報を管理し配信するための一連のサービスを提供

2004/6/14      NII-IRPワークショップ      4

図 4

**オープン・アクセス運動における  
位置づけ**

- 学術論文に対する障壁のないオンライン・アクセスをめざした運動
- BOAI(Budapest Open Access Initiative)の2つの戦略
  - オープン・アクセス誌の創出
  - セルフ・アーカイビング
- セルフ・アーカイビング
  - 著者が, プレプリント(査読前論文) and/or ポストプリント(査読後論文)を個人サーバ, 分野別サーバ, あるいは大学(図書館)が運営するサーバに蓄積し, それを無料で公開する行為

2004/6/14      NII-IRPワークショップ      5

図 5

**機関リポジトリの現状**

- Institutional Archives Registry (Eprints.org)
  - 204リポジトリ(主題ベースのリポジトリも含まれる)
  - <http://archives.eprints.org/eprints.php>
- "PALS Pathfinder Research on Web-based Repositories Final Report"
  - 45リポジトリ
  - [http://home.q00.itscom.net/ojiro/table\\_IR.xls](http://home.q00.itscom.net/ojiro/table_IR.xls)

2004/6/14      NII-IRPワークショップ      6

図 6

## CODA(カリフォルニア工科大学)

- <http://library.caltech.edu/digital/>
- カリフォルニア工科大学の各種リポジトリの集合体
- 13種のリポジトリが公開され、5種が準備中(2004年6月9日現在)
- ソフトウェアは、サウサンプトン大学で開発されたEPrintsを使用

2004/6/14 NII-IRPワークショップ 7

図 7

## DSpace@MIT

- <https://dspace.mit.edu/index.jsp>
- 2002年11月にサービス開始
- 現在、5つの研究コミュニティ(学部、学科、センター等)が参加
- DSpaceソフトウェアをヒューレット・パッカード社と共同開発→オープンソース化
- DSpace連合の結成(ケンブリッジ大学、コロンビア大学、コーネル大学、ロチェスター大学、オハイオ州立大学、トロント大学、ワシントン大学)

2004/6/14 NII-IRPワークショップ 8

図 8

## eScholarship(カリフォルニア大学)

- <http://repositories.cdlib.org/escholarship/>
- CDL(California Digital Library)が2002年4月にサービス開始
- 研究論文、ワーキングペーパー、テクニカルペーパー、プレプリント等を格納
- 登録ペーパー数:2,980(2004年6月9日現在)
- 約10,000件/週のダウンロード実績
- ソフトウェアはバークレイ校で開発されたbepressを使用

2004/6/14 NII-IRPワークショップ 9

図 9

## プロジェクト

- ARROW
  - <http://arrow.edu.au/>
  - 機関リポジトリのベスト・プラクティスを支援するソフトウェアやソリューションの実証実験を目的とするオーストラリアのプロジェクト
- CARL
  - <http://www.carl-abrc.ca/projects/ir/>
  - カナダ研究図書館協会を中心とした、学術機関リポジトリのパイロット・プロジェクト
- DARE
  - <http://www.darenet.nl/en/foon>
  - オランダの全国規模の分散リポジトリ構築計画
- FAIR
  - [http://www.jisc.ac.uk/index.cfm?name=programme\\_fair](http://www.jisc.ac.uk/index.cfm?name=programme_fair)
  - 英国JISC(Joint Information Systems Committee)のプログラム。学術機関リポジトリの普及をめざし、14プロジェクト(50機関)が活動中

2004/6/14 NII-IRPワークショップ 10

図 10

## 学内合意形成

- なぜ機関リポジトリが必要なのか？
  - 存在意義について理解を求める
- なぜ図書館が運営するのか？
  - 従来の図書館機能の延長(学術情報の収集、組織化(メタデータ、主題分析)、利用提供、保存)
  - 著作権及び学術コミュニケーションをめぐる諸問題に関する専門家
  - 技術的なノウハウの蓄積

2004/6/14 NII-IRPワークショップ 11

図 11

## 運用方針の策定

- コンテンツ・ガイドライン
  - 登録可能な投稿者(誰が登録できるのか?)
  - 登録可能なコンテンツの種類(論文、教材、ソフトウェア、データセット等々)
  - 登録可能なコンテンツの形態
  - 品質管理(査読に相当する品質管理のプロセスが必要か?)
  - 登録したコンテンツの削除(取り下げ)
- 利用許諾契約書
  - コンテンツをリポジトリに蓄積し公開するための非排他的権利の譲渡を求める

2004/6/14 NII-IRPワークショップ 12

図 12

## システムの実装

- オープン・ソース
  - A Guide to Institutional Repository Software v 2.0
    - <http://www.soros.org/openaccess/software/>
  - 機関リポジトリ構築ソフトウェアガイド(上記ガイド第1版の翻訳)
    - [http://www.nii.ac.jp/metadata/oai-pmh/osi\\_ir\\_software\\_guide.html](http://www.nii.ac.jp/metadata/oai-pmh/osi_ir_software_guide.html)
- 商用ソフトウェア
  - bepress (Berkeley Electronic Press)
  - インフォコム
  - CMS

2004/6/14                      NII-IRPワークショップ                      13

図 13

## 登録(投稿)の促進

- 既存リポジトリのコンテンツ数
  - 45のリポジトリの収録コンテンツ数のメジアン(中央値)=290("PALS Report"による)
- 段階的な促進
  - 第1フェーズ
    - 初期データの構築(デモンストレーション)
  - 第2フェーズ
    - 研究者の自己登録(セルフ・アーカイビング)を促す

2004/6/14                      NII-IRPワークショップ                      14

図 14

## 初期データの構築(有用性の実証)

- 学内外のサーバ上で既に公開されているコンテンツの登録
  - 学内:個人, 学部・学科
  - 学外:eプリントアーカイブ(arXiv, ADS)
- NIIの紀要ポータル事業によって電子化されたコンテンツ
- 既に電子化されているが, サーバ上では未公開のコンテンツ(CD-ROM化された学位論文等)
- リポジトリへの登録を許可する雑誌に掲載された, 学内研究者による論文

2004/6/14                      NII-IRPワークショップ                      15

図 15

## 考えられる障壁

- インセンティブの欠如
  - 自分のウェブサイトで既に公開している
  - どんなメリットがあるの?
  - 登録しなくても何のペナルティもない
- 登録行為に対する抵抗感
  - 登録に手間がかかる
  - 時間がない
- 著作権に関する懸念
  - (特に学術誌掲載論文の場合)登録する権利があるの?

2004/6/14                      NII-IRPワークショップ                      16

図 16

## 除去するための方策

- インセンティブの欠如
  - メリットの強調(アメ)
  - 強制力(ムチ)
- 登録行為に対する抵抗感
  - 使いやすい簡易な登録インターフェイスの提供
  - 図書館員による登録支援
- 著作権に関する懸念
  - 出版社のポリシーの報知

2004/6/14                      NII-IRPワークショップ                      17

図 17

## メリットの強調(アメ)

- 無料でアクセスできるオンライン論文の被引用率
  - オフライン論文に比べて2.6倍多く引用されている(Lawrence, Steve. "Online or invisible?" *Nature*. Vol.411, No.6837, p.521, 2001.)
  - 自らの研究成果の可視性の向上
- 研究成果の長期保存・利用の保証
- 成果一覧リストの出力

2004/6/14                      NII-IRPワークショップ                      18

図 18

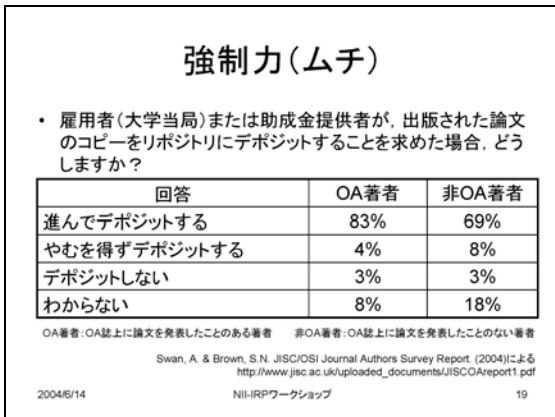


図 19

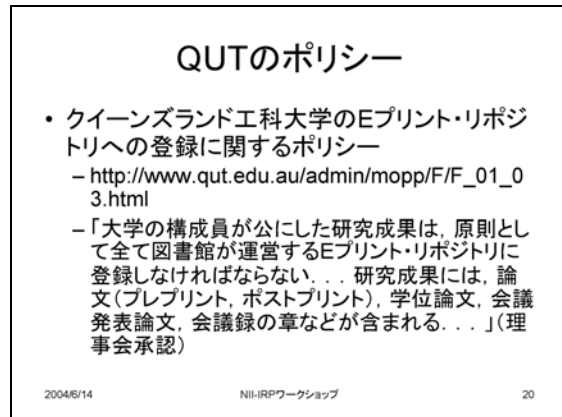


図 20



図 21

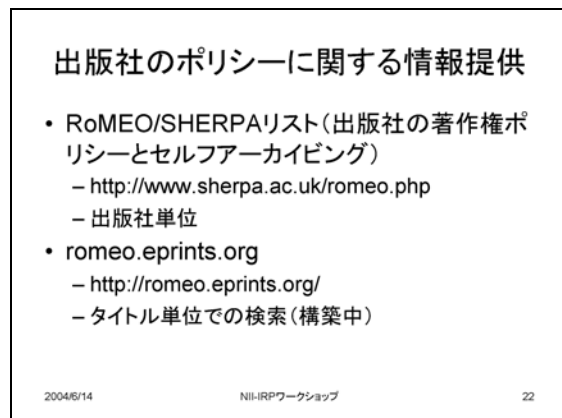


図 22

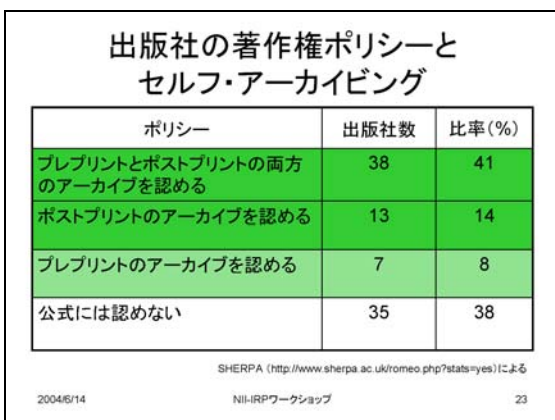


図 23

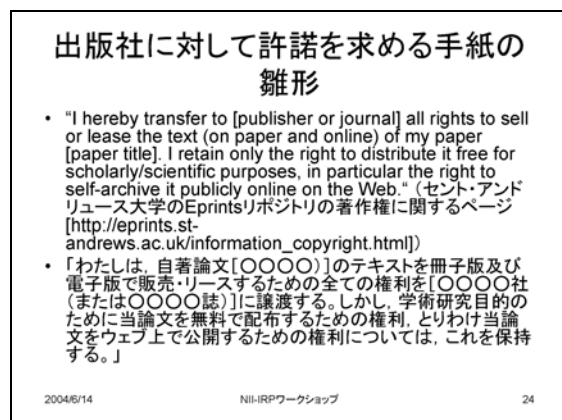


図 24



## 今後の展望

- リポジトリ共同体
  - 課題解決に向けた知識, 経験, 技術の共有
- 大学図書館とNIIのコラボレーション
  - 大学図書館
    - リポジトリの構築と運用
    - 学内の学術情報を収集, 保存(データ・プロバイダ)
  - 国立情報学研究所
    - 各大学のリポジトリ構築支援
    - 各大学のリポジトリに対する総合的窓口機能(サービス・プロバイダ)

2004/6/14                      NII-IRPワークショップ                      25

図 25

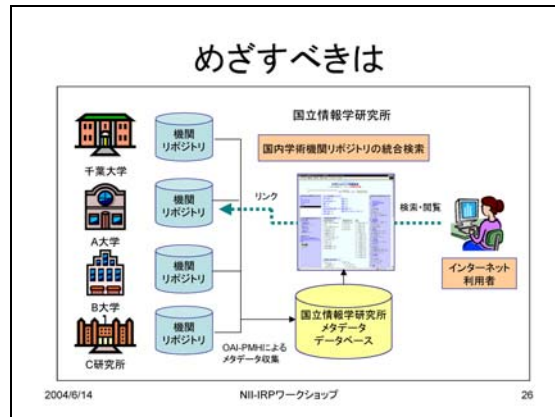


図 26

講演資料「機関リポジトリの品質保証／土屋俊・千葉大学文学部」  
(中間報告会, 2004.9)

機関リポジトリの品質保証

土屋俊  
(千葉大学)

図 1

話題

- 克服すべき課題の分類(あくまで暫定)
  - 技術的問題
    - サイト構築
    - 連携(メタデータ・ハーベスティング)
    - マイグレーション
  - 制度的問題
    - 機関リポジトリの「機関」としての認知
    - 人的資源問題
    - 搭載コンテンツのジャンルと品質
    - 可視性と誘導
    - 継続性(マイグレーション)
    - 権利処理

図 2

サイト構築

- 既成のソフトウェア
  - Eprints
  - DSpace
  - さらにまたいろいろのopen source
  - Endeavor(?)
- オーダーメイドのソフトウェア
- ハードウェア
  - サイトのネットワーク上の位置(バックボーン? ラン上?)
  - セキュリティとバックアップ

図 3

メタデータハーベスティング

- メタデータの作成
  - 著者作成
  - 図書館作成
- 自分のカタログの位置づけ
  - それそのものとして目録化すること
  - 業績データベースとの連携
- ハーベスティング
  - 中央ハーベスティング
  - 相互ハーベスティング

図 4

マイグレーション

- 著者よりも生き延びなければならない
  - 退職、転職との関係(制度的問題?)
  - 著者の個人ホームページよりも長く
- 出版者よりも生き延びなければならない
  - プラットフォームの変遷
    - 覚悟する必要がある
    - 文書規格の遵守の必要性
- ひょっとして機関よりも生き延びなければならない?
  - 統合、譲渡の許容

図 5

機関としての認知への努力

- 図書館の気まぐれの事業であってはならない
- 「大学」レベルでの理解
  - 経営者への説得
  - 研究者への説得(学内、学会)
  - 図書館への説得
- 研究協力との連携
  - 学内各種データベースとの連携
  - 外部資金獲得との連携
- 評価事業との連携

図 6

### 人的資源問題

- どのような人的資源が必要か
  - コンピュータ操作
  - 制度的側面
  - 対外・対内関係
- どれだけの人的資源が必要か(機関認知的観点からも重要)
  - 専任?
  - 片手間?
  - 外注?
  - その他?

図 7

### 可視性と誘導

- どのようにして利用者に認知させるか
  - 学内利用者は著者である
  - 学内利用者はたまたま見つけた人である  
⇒ 訴えかける対象の特定が困難
- どのようにして対象に誘導するか
  - 主題?
  - キーワード?
  - 全文検索?
  - 著者名?

図 8

### 継続性

- 機関としてのコミットメントの確保の必要性
  - 位置づけの理念化
    - 大学がそれを必要とするということの明確な表明
    - そのための資源投下の必要性の明確な表明
  - 担当者の制度化(図書館から独立でもよい)
  - 「マンネリ化」の必要性
- 研究者サイドでの関心の持続
  - 習慣化すること
  - 自覚化すること

図 9

### 権利処理

- 著者による著作権管理の不可能性
  - 出版者への譲渡は理不尽ではない
  - しかし、理不尽でもある
  - 必要なのは、「権利処理担当者」
  - 大学が権利処理をする可能性
- 「緑化」問題
  - 認識の共有へ
  - 機関の支援(機関独自統一契約書)
  - 雇用契約レベル

図 10

### 品質保証

- 品質保証を機関リポジトリはできない
- したがって、品質保証ずみのものを搭載する必要がある
- それは可視性向上との連携でもある
  - 既公刊論文
  - 他で発表されたもの(上演、講演などなど)
  - 外部資金によるデータベース、報告書
- 筑波大学の電子図書館はなぜ現状のようになっているのか?

図 11

講演資料「なぜ大学は機関リポジトリをもたねばならないか／土屋俊・千葉大  
学文学部」（報告会, 2005.2）

なぜ大学は機関リポジトリを  
もたねばならないか

土屋俊  
(千葉大学)  
2005年2月10日国立情報学研究所にて

図 1

機関リポジトリをめぐる諸説紛々をちょっと整理したい

- オープン・アクセスからの議論
  - セルフ・アーカイビング派 (Harnad, UK 下院 Select Cmmtt等)
  - オープン・パブリッシング派 (CDL, Boston College, SPARC白書等) ⇒ 新型大学出版会?
- コレクション構築からの議論
  - Uva流古典的議論 (= 日本の「電子図書館」)
  - コーネル大学流 (arXiv.org, Euclid—ちょっと変則?)
- e-Learningからの議論
  - たとえば, OCW(⇒DSpace), ELI@Yale
- 大学論的アプローチ
  - Caltech流

図 2

(残念ながら) やっぱりLynch流定義か?

a university-based institutional repository is a set of services that a university offers to the members of its community for the management and dissemination of digital materials created by the institution and its community members.

ie. Organizational commitment to stewardship of digital materials:

- organization
- access or distribution
- long-term preservation

ARL Bimonthly Report 226, February 2003  
Institutional Repositories: Essential Infrastructure for Scholarship in the Digital Age  
by Clifford A. Lynch, Executive Director, Coalition for Networked Information

図 3

要するに、機関リポジトリとは

- 機関の責任において設置され、運営される
- 機関の資源によって設置され、運営される
  - したがって、文書を利用することによって、利用者が費用を負担することはない
- (原則として) 機関内で作成された文書類を保存する
- 保存は電子的に行う
  - 電子的に保存された文書は、インターネットを経由して誰でも利用可能である

図 4

機関リポジトリの分類基準

- Accessibility
  - Open access
  - Controlled access(registration/paid)
- Contents
  - Research (Preprint/Postprint/Database/...)
  - Pedagogy(Course material/...)
  - Etc
- Metadata(Harvestable/...)
- Management system

図 5

「機関」とは？

- 「大学」「研究機関」を指す
- 学部、学科、研究室、個別研究者は？
  - 上述機関からの委任によって運営の一端を担う
  - つまり、ボトムアップではない
- JSPSやJSTのような資金提供団体は？
  - 研究者・教育者と共通の目的を有する必要がある
- 国立国会図書館や国立情報学研究所は？
  - 研究者・教育者と共通の利害を有する必要がある

図 6

### 文書類とは？

- 研究成果物
  - プレプリントとポストプリント
  - データベース、テキストベース、校訂本、音声、画像・映像資料
  - 博物資料
  - 特許???
- 教育素材
  - テキスト、参考書、参考資料、演習・試験問題、図版、実験シミュレーション等
- 文書館的機能の対象
  - 行政文書、個人メモ、写真等

図 7

### 「電子的に」とは？

- 現状では、
  - デジタル化された資料として保存
  - なんらかのマネージメントシステムによって管理
  - メタデータをハーベスト可能な状態として
  - World Wide Webを基礎として利用可能
- それゆえに生ずるさまざまな技術的課題
  - 保存(マイグレーション・エミュレーション)
  - 文書等形式の標準化と管理システム
  - メタデータ形式の標準化(利用用・管理用)
  - 利用可能性の向上とインテグリティの確保
  - 重複努力の回避

図 8

### 「誰でも」「費用負担なしに」とは？

- アクセス・コントロールを行わない
  - ただし、ライセンスによる導入資料の場合が問題
  - 利用実績のカウントは問題
- 課金を行わない
  - コストの回収を行わない。つまり、機関がコストをすべて負担する
- いわゆる「オープン・アクセス」の状態になる

図 9

### 大学になんのメリットがあるのか？

- 機関のアイデンティティの確立
  - 研究と教育に関する社会的説明責任の履行
  - RAE・QAAの評価およびア Krediteーション的評価への対応
  - 歴史的アイデンティティの創造と継承
- 研究者にとっての研究インパクトの増進
- 教育者としての社会的責任
  - 教育は教材につきない
- 大学資源の社会還元

図 10

### 今の北米は実は、90年代日本と同じ？

- ELI@Yale
  - Special collectionの電子化
- CalTechのリポジトリ
  - 「筑波大学電子図書館」?
- アリゾナ州立大学の文書館
  - さまざまな展望
- ただしライセンス導入できるコンテンツが決定的に不足
- CMSがほとんど使われていない
- 大学自体の教育への取り組みの変化が必要

図 11

### 日本の学会誌の対応

- 千葉大学による予備調査
  - 全体としてまだ十分な自覚がない
  - 「機関」についての関心の不足
    - 日本の学会の歴史的事情
  - 学会の「収入」に関する(いわれなき)危惧
- NII/ELSにおける電子図書館
  - 過去のもの電子化
  - 課金を許容(半数の学会が課金)

図 12

## 図書館が運営しなければならない理由

なぜならば、

- ほかに図書館がすることがなくなるから
  - 学術雑誌は全部電子ジャーナルになる
  - 参考図書、専門図書はすべて電子化される
  - 検索はすべてGoogle(の将来の姿)でユーザ自身が行う(レファレンスサービス不要論)
- 図書館だけがもっている機能が必須であるから
  - 資料管理・組織化(コレクション構成)
  - メタデータ付与とその管理・利用可能化

図 13

## 「コレクション」の新しい意味

- これまでのコレクション
  - 外部からの導入:「ものを買う」
  - 選書:すでにあるものを選ぶ→ライセンスへ
  - ビッグディールによる変化(けっして悪くなかった?)
- これからのコレクション
  - 自分のキャンパスでできたものを集める(「発信」?むしろ保存(とアクセス))
  - 選書でなく、企画(教育・研究状況の把握・連携)
    - arXiv.org、Project Euclidなどのコーネル大学のコレクション
    - 「出版会」との関係(スタンフォード大学の場合)
    - デジタル・リソースの管理の問題

図 14

## 結論

- 大学は、研究・教育・社会還元に関して、情報化社会における機関としてのアイデンティティ維持のために機関リポジトリを持たねばならない
- 機関リポジトリは、(研究と教育を還流する)学術情報の生産・流通・保存に寄与する
- 機関リポジトリの寄与は、大学構成員の義務である
- 機関リポジトリは、図書館が運営することにすべきである

図 15

## 結論からの帰結

- 学内における「アドミニストレーション」へのアプローチの重要性
- 大学間コンソーシアムの重要性
  - 大学図書館のプライオリティ
  - 大学間連携へのアプローチ
  - NIIのからみ方
- 国際連携のあり方

図 16

## 第 1 部 システム構築

# 第1章 システム構築

## 1.1 はじめに

学術機関リポジトリを構築するためのシステムは数多く存在する。それらは、それぞれ固有の設計思想を有し、また、技術的要件もそれぞれ異なる。そのためどのシステムを採用するかは、各機関の「達成すべき目的」、「状況」にあわせて選択する必要がある。

ここでは数あるシステムの中から、今回のプロジェクトで採用した EPrints 及び DSpace の機能と特徴及び両システムの技術的側面を簡単に述べる。

## 1.2 機能と特徴

EPrints と DSpace に共通する機能は、(1) OAI-PMH データプロバイダ機能 (2) メタデータ形式はダブリンコアを採用 (3) Web 検索システムの提供である。これらの機能は、学術機関リポジトリの最大の目的である、「研究成果物の公開」を容易に実現する。

特に学術機関リポジトリにとって OAI-PMH 準拠にすることは、世界中に分散したアーカイブに相互運用性を与える。

(表 1) は、EPrints と DSpace の比較対照表である。以下ではそれぞれの特徴を簡単に述べる。

EPrints は、システム構成の簡便さがその特徴であり、論文リポジトリ指向のシステムであるといわれている。また、比較的早くから開発が進められてきたことで、世界中の多くの機関が採用しており、新規利用機関をサポートする大規模なネットワークが形成されている。

DSpace は、その機関を構成するコミュニティ (例えば、学部、学科、研究グループなど) 独自の運用ルールに合わせて、コンテンツ管理やユーザ認証、知的所有権処理に対処できるように設計されている。そのことは、多くの学問分野を擁する大規模総合大学の知的生産物を保存するデジタルリポジトリとして最適であると言える。

(表 1)

	EPrints	DSpace
<b>ユーザー管理</b>		
ユーザ種別によるアクセス制限	あり	あり
ファイル/オブジェクト単位のアクセス制限	あり	あり
<b>コンテンツ管理</b>		
同一システムでの複数コレクションの定義	あり	あり
コレクション毎に異なる登録パラメタの設定	なし	あり
コレクション毎のホームページ管理	なし	あり
登録処理の役割	利用者, 編集者, 管理者	投稿者, 査読者, 是認者, 編集者
配布ライセンスの要求	なし	あり
コンテンツと配布ライセンスの保存	なし	あり
利用統計作成	なし	あり
利用報告書作成	なし	あり
URL からのアップロード	あり	なし
メタデータ及びオブジェクトの一括入力	あり	あり
<b>メタデータ</b>		
サポートするメタデータスキーマ	ダブリンコア	限定子付きダブリンコア
拡張されたメタデータのサポート	あり	カスタマイズ可能
メタデータの査読サポート	受付, 編集, 差戻し(要変更), 削除	あり



メタデータ出力	カスタム XML スキーマ	METS およびカスタム XML スキーマ
メタデータハーベスティングの禁止	あり	あり
メタデータフィールドの追加・削除	あり	あり
Unicode 文字セットのサポート	あり	あり
<b>ユーザインターフェース</b>		
多言語インターフェースのサポート	あり	あり
ディスカッションフォーラムのサポート	あり	なし
<b>検索能力</b>		
フルテキスト検索	あり (Text, HTML, PDF)	あり (HTML, PDF, MSWord)
全メタデータ検索	あり	あり
検索結果のソート	あり	なし
<b>永続的なドキュメント識別</b>		
識別子のシステム付与	あり	あり
CNRI ハンドル	なし	あり
<b>データ保存のサポート</b>		
定義されたデジタル保存戦略	なし	あり
保存用メタデータのサポート	なし	あり

### 1.3 EPrints 及び DSpace の技術的側面

EPrints は、(表 2) にある通り、Perl+MySQL+Apache という構成である。DSpace は、サーブレット/JSP による Web アプリケーションであるため技術的には比較的高度であるが、その多機能性から大規模総合大学に適したシステムと言える。

どちらにしても、各機関の事情に則したカスタマイズを行なう上では、プログラミングを始めとするシステム構築技術が必須となる。

各機関が置かれた状況によっては、後にも紹介する商用システムを導入することもひとつの選択肢である。(商用システムについては、巻末を参照のこと)

(表 1)

	EPrints	DSpace
開発元	サウサンプトン大学	MIT, HP
最新版次	2.3.8 (2005 リリース)	1.2.1 (2005.1 リリース)
ソースライセンス	GNU GPL (注 2)	BSD (注 1)
サポートする OAI-PMH の版	OAI-PMH 2.0	OAI-PMH 2.0
<b>ソフトウェア</b>		
(検証済の)OS	UNIX/Linux/MacOSX	UNIX/Linux/MacOSX/Windows
プログラム言語	Perl	Java
データベース	MySQL	PostgreSQL
Web サーバ	Apache	Jakarta Tomcat (Apache との連携可)
検索エンジン	該当なし	Lucene
<b>カスタマイズ</b>		
インターフェースの日本語化	日本語化キットあり	JSP ファイルの翻訳
日本語メタデータ検索	あり (第 1 章 3 項)	あり (第 1 章 3 項参照)
日本語コンテンツ全文検索	フィルタープログラムの日本語化	フィルタープログラムを日本語化

注 1) <http://www.opensource.org/licenses/bsd-license.php>

注 2) <http://www.gnu.org/licenses/gpl.html>

(参考)

- 学術機関リポジトリ構築ソフトウェアガイド (第 3 版) (日本語訳)  
[http://www.nii.ac.jp/metadata/irp/osi\\_guide\\_3/](http://www.nii.ac.jp/metadata/irp/osi_guide_3/)
- DSpace システム説明書 (日本語訳)  
<http://www.ndl.go.jp/jp/aboutus/dspace/>
- e-print 機関アーカイブのセットアップ (日本語訳)  
<http://www.nii.ac.jp/metadata/oai-pmh/eprints/>
- グラスゴー大学における e-プリント機関アーカイブの展開 (日本語訳)  
<http://www.nii.ac.jp/metadata/oai-pmh/nixon1/>

## 第 2 章 DSpace の導入と設定

### 2.1 DSpace の導入手順

DSpace のインストール手順について説明する。

一般的な情報源として、<http://www.dspace.org/> を必要に応じて参照すること。

インストール作業をするにあたっては、基本的な UNIX の操作をある程度習得していることが望ましい。また、下の操作はすでに完了しているものと仮定する。

適切な OS (Linux 等) がインストールされている。  
適切な IP アドレスやルーティングテーブル、またサーバー名が設定されている。  
サーバーから電子メールが発行できるよう設定されている。  
Web サービスを公開できるよう、適切なポートを開放する等の設定がされている。  
(DSpace は、tomcat 等を使って Web サービスを行う)

最初に UNIX 上で dspace グループ、および dspace ユーザーを作成する。普段の作業はこのユーザーで行う。(下のコマンドはスーパーユーザーで行う)

```
# groupadd Δ dspace
# useradd Δ -g Δ dspace Δ dspace
# passwd Δ dspace
# (新しいパスワードを 2 回入力)
```

インストールに必要なとなるソフトウェアは、次のようなものである。

#### Java SDK

Java のソースコードをコンパイル・実行するための基本的なツール群。無償。

#### Apache Ant

コンパイルやその他の操作を常に手順正しく行うためのツール。Java で記述されている。無償。

#### PostgreSQL

データベース (RDB) を管理するソフトウェア。無償。

#### Jakarta Tomcat

WEB 上で動作する、Java サーブレットというアプリケーションを動作させるためのツール (サーブレットエンジン) のひとつ。Java で記述されている。無償。

#### DSpace

学術機関リポジトリを実現するためのソフトウェアで、Java サーブレットで記述されている。今回の例では、Jakarta Tomcat の上に載って、PostgreSQL のデータベースを読み書きしながら動作する。

BSD ライセンスに基づいて配布されているので、無償で利用でき、自由にカスタマイズすることが許される。

次の 2 節では、Java の基本的なツールのインストールと、ビルドツールである Apache Ant のインストールを順に行う。

##### 2.1.1 Java SDK

Java のバージョン 1.3 以降がインストールされている場合は、このステップを飛ばして

よい。調べるには、コマンドラインから

```
$ java △ -version
```

と入力する。エラーが出る場合、または低いバージョンしかない場合は、下の手順でインストールを行う。

#### 必要なファイルの取得

<http://java.sun.com/j2se/1.4.2/download.html> から Download J2SE SDK を選択し、J2SE をダウンロードする。ダウンロードは、通常の Web ブラウザで行うか、Linux の wget コマンドなどが使えるようなら、それを使う。

#### インストール方法

ここでは、/usr/local にインストールする。

```
$> su
Password : ← root ユーザーのパスワードを入力する。
#> mv △ j2sdk-1_4_2_04-linux-i586.bin △ /usr/local
#> cd △ /usr/local
#> chmod △ 755 △ j2sdk-1_4_2_04-linux-i586.bin
#> ./j2sdk-1_4_2_04-linux-i586.bin
#> exit
$>
```

#### 環境変数の設定

環境変数を設定するため、dspace ユーザのホームディレクトリで「.bash\_profile」を編集し、下の記述を追加する。

(ここでは bash を使っているときの例を用いる。)

```
export JAVA_HOME=/usr/local/j2sdk1.4.2_04
export CLASS_PATH=$JAVA_HOME/lib/tools.jar
export PATH=$JAVA_HOME/bin:$PATH
```

追加修正後は、

```
$>. △ ~/.bash_profile
```

で環境変数を即時に変更できる。

コマンドラインから再度 java のバージョンを調べて、下のような返事が返ってくれば成功である。

```
$ java △ -version
java version "1.4.2_04"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.2_04-b05)
Java HotSpot(TM) Client VM (build 1.4.2_04-b05, mixed mode)
```

#### 2.1.2 Apache Ant

すでに適切なバージョンの Apache Ant がインストールされている場合は、この過程を飛ばしてよい。

### 必要なファイルの取得

<http://ant.apache.org/bindownload.cgi> から `apache-ant-1.6.1-bin.tar.gz` を選択し、ダウンロードする。

### インストール方法

圧縮されたファイルを展開する。ここでは圧縮ファイルが `/usr/local/src` に格納されているものとし、`/usr/local` に Apache Ant を展開する例を示す。

```
$> cd △ /usr/local/src
$> su △ -
Password : ← root ユーザーのパスワードを入力する。
#> gzip △ -d △ apache-ant-1.6.1-bin.tar.gz
#> tar △ xvf △ apache-ant-1.6.1-bin.tar △ -C △ /usr/local
```

### 環境変数の設定

環境変数を設定するために、`dspace` ユーザーの「`.bash_profile`」に下記を追加する。

```
export ANT_HOME=/usr/local/apache-ant-1.6.1
export ANT=$ANT_HOME/bin/ant
export PATH=$ANT_HOME/bin:$PATH
```

追加修正後は、

```
$>. △ ~/.bash_profile
```

で環境変数を即時に変更する。

コマンドラインから下のように入力して、返事が返ってくることを確認する。

```
$ ant △ -version
Apache Ant version 1.6.1 compiled on February 12 2004
```

次に、PostgreSQL のインストールと、DSpace のインストールに備えた設定までを行う。

### 2.1.3 PostgreSQL

コンパイルオプションが重要なので、すでにパッケージとしてインストールされている PostgreSQL を使うのは動作が保証できない場合がある。また、ここからの方法では `dspace` ユーザーの権限で PostgreSQL をインストールするので、できれば既存の PostgreSQL をアンインストールするか、又は別のポート番号で接続できるように棲み分けてインストールするべきである。

### 必要なファイルの取得

<ftp://ftp.jp.postgresql.org/source/v7.4.2/> から `postgresql-7.4.2.tar.gz` を選択しダウンロードする。

取得したアーカイブから、ソースファイルを `/usr/local/src` に展開する。

```
$> su △ -
Password :
#> mv △ postgresql-7.4.2.tar.gz △ /usr/local/src
#> cd △ /usr/local/src
#> gzip △ -d △ postgresql-7.4.2.tar.gz
```

```
#> tar Δ xvf Δ postgresql-7.4.2.tar
#> chown Δ -R dspace.dspace postgresql-7.4.2
(下の手順に続く)
```

#### コンパイル・インストール

postgreSQLをインストールするディレクトリを作成する。ここでは「/usr/local/pgsql」にインストールする例を示す。

```
(上の手順から続く)
#> mkdir Δ /usr/local/pgsql
#> chown Δ dspace.dspace Δ /usr/local/pgsql
(下の手順に続く)
```

次にソースのコンパイルとインストールを行う。スーパーユーザーを終了していること。

```
(上の手順から続く)
#> exit (スーパーユーザーを終了する)
$> whoami (現在のユーザーが dspace であることを確認する)
dspace
$> cd Δ /usr/local/src/postgresql-7.4.2
$> ./configure Δ --prefix=/usr/local/pgsql Δ --enable-multibyte
Δ --enable-unicode Δ --with-java
$> make
(この間、時間がかかる)
$> make Δ install
```

※ この際、環境変数 ANT を設定していないと、configure を実行した時点でエラーとなる場合がある。

#### 環境変数

dspace ユーザーの「.bash\_profile」に下記を追加して、即時に反映しておく。(以前までの設定で、\$JAVA\_HOME, \$ANT\_HOME は設定してあること)

```
export POSTGRES_HOME=/usr/local/pgsql
export PGLIB=$POSTGRES_HOME/lib
export PGDATA=$POSTGRES_HOME/data
export MANPATH=$MANPATH:$POSTGRES_HOME/man
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$PGLIB
export PATH=$POSTGRES_HOME/bin:$JAVA_HOME/bin:$ANT_HOME/bin:$PATH

$>. Δ ~/.bash_profile
```

#### データベースの初期化

データベースの初期化を行う。

```
$> whoami (現在のユーザーが dspace であることを確認する)
dspace
$> initdb
```

#### 認証・ソケット接続の設定

/usr/local/pgsql/data に格納されている postgresql.conf と pg\_hba.conf を編集して、

PostgreSQL がソケット接続できるよう設定する。

```
$> cd △ /usr/local/pgsql/data
$> vi △ postgresql.conf
(手順 A)
$> vi △ pg_hba.conf
(手順 B)
```

(手順 A) postgresql.conf の 30 行目付近に、次の一行を追加する。

```
tcpip_socket = true
```

(手順 B) pg\_hba.conf の最後に、次の一行を追加する。

```
host dspace dspace 127.0.0.1 255.255.255.255 md5
```

#### postmaster の起動

postmaster を起動する。

```
$> whoami (現在のユーザーが dspace であることを確認する)
dspace
$> pg_ctl △ start
```

環境変数 PGDATA が設定されていないと、エラーが発生することがある。  
正常に起動したかどうかは、netstat コマンドを使って確認することができる。

```
$> netstat △ -ln
...
tcp    0    0    0.0.0.0:5432      0.0.0.0:*    LISTEN
...
```

上のような一行が含まれていれば、DB が接続待ちの状態であることがわかる。

#### データベースユーザーと、データベースの作成

PostgreSQL に DSpace 用のデータベースを作成する。この作業は、postmaster が起動された状態で実行すること。

```
$> whoami (現在のユーザーが dspace であることを確認する)
dspace
$> createdb △ -U △ dspace △ -E △ UNICODE △ dspace

( dspace という名前のデータベースを作成する。
 所有者は、UNIX 上の dspace ユーザー。 )

CREATE DATABASE
$>
```

initdb コマンドを実行したユーザーは、自動的にそのデータベースのスーパーユーザーとして登録されている (UNIX のスーパーユーザーとは別)。このため、UNIX の dspace ユーザーで initdb を流した場合、DB 上の dspace ユーザーができており、改めて createuser を実行するとエラーになる。それでもインストール全体には支障はない。(つ

まり、この下のコマンドを実行する必要はない。)

```
$> createuser Δ dspace ←PostgreSQL にユーザを作成 (ユーザ名 dspace)
```

#### 2.1.4 Jakarta Tomcat

JAVA サブレットを実行するための環境である Jakarta Tomcat のインストールと、DSPACE のインストールに備えた設定までを説明する。

##### 必要なファイルの取得

<http://jakarta.apache.org/site/binindex.cgi> から jakarta-tomcat-4.1.30.tar.gz を選択し、ダウンロードする。

##### インストール

jakarta-tomcat-4.1.30.tar.gz を展開する。この例では、一旦 /usr/local/src 上に移動してから展開している。展開先は、ここでは /usr/local。

```
$> su Δ -  
Password : ← root ユーザーのパスワードを入力する。  
#> mv Δ jakarta-tomcat-4.1.30.tar.gz Δ /usr/local/src  
#> cd Δ /usr/local/src  
#> gzip Δ -d Δ jakarta-tomcat-4.1.30.tar.gz  
#> tar Δ xvf Δ jakarta-tomcat-4.1.30.tar Δ -C Δ /usr/local  
#> chown Δ -R Δ dspace.dspace /usr/local/jakarta-tomcat-4.1.30  
#> exit (スーパーユーザーを終了する)  
$>
```

##### 環境変数の設定

dspace ユーザーの「.bash\_profile」に下記を追加して、即時に変更しておく。これはスーパーユーザーから抜けて行う。

```
export JAVA_OPTS="-Xmx512M -Xms64M -Dfile.encoding=UTF-8"
```

```
$>. ~/.bash_profile
```

##### 動作確認

Tomcat が動作することを確認する。

```
$> whoami (現在のユーザーが dspace であることを確認する)  
dspace  
$> cd Δ /usr/local/jakarta-tomcat-4.1.30/bin  
$> ./startup.sh
```

正常に起動したかどうかは、netstat コマンドを使って確認することができる。

```
$> netstat Δ -ln  
...  
tcp    0    0  0.0.0.0:8080      0.0.0.0:*    LISTEN  
...  
...
```

上のような一行が含まれていれば、Web サービスが接続待ちであることがわかる。ブラウザを起動させて以下のアドレスを入力し、Tomcat の初期画面が表示されるかを確



認する。まだ特別な設定をしていないので、Jakarta Tomcat のデフォルトのページなどが表示される。

`http://このコンピュータの名前:8080/`

動作が確認できたら、いったん tomcat を終了しておく。

```
$> cd △ /usr/local/jakarta-tomcat-4.1.30/bin
$> ./shutdown.sh
```

### 2.1.5 DSpace

#### 必要なファイルの取得と展開

`http://prdownloads.sourceforge.net/dspace/dspace-1.2.tar.gz?download` から `dspace-1.2.tar.gz` をダウンロードして、その後展開する。

(※2005/1/31 現在、`dspace-1.2.1beta4` が最新のリリースとなっている。いくつかのバグが修正されているので、これか、またはより新しいバージョンをダウンロードしてインストールを進めるのが望ましい。その際、以下に続く `dspace-1-2` というディレクトリ名は適宜読み替えること。)

```
$> su △ -
Password : (パスワードを入力)
#> cd △ [ダウンロードしたディレクトリの場所]
#> mv △ dspace-1.2.tar.gz △ /usr/local/src
#> cd △ /usr/local/src
#> gzip △ -d △ dspace-1.2.tar.gz
#> tar △ xvf △ dspace-1.2.tar
#> chown △ -R △ dspace.dspace △ /usr/local/src/dspace-1.2-source
#> exit
$>
```

#### Java ライブラリーの追加

以下、いくつかの Java ライブラリーを、`/usr/local/src/dspace-1.2-source/lib` に追加格納してゆく。執筆時現在の最新バージョンを記しているが、それより上のバージョンでも通常は問題ないはずである。

#### activation.jar

`http://java.sun.com/products/javabeans/glasgow/jaf.html` から [JavaBeans? Activation Framework](#) をクリックし、`jaf-1_0_2-upd.zip` をダウンロードする。`jaf-1_0_2-upd.zip` を展開し、生成される `activation.jar` を対象ディレクトリにコピーする。

```
$> cd △ [展開したディレクトリ]
$> cp △ activation.jar △ /usr/local/src/dspace-1.2-source/lib
```

#### servlet.jar

`http://java.sun.com/j2ee/ja/jsp/download.html` から `Java Servlet 2.3 and JSP 1.2` をクリックし、`jsp-1_2-fcs-classfiles.zip` をダウンロードする。`jsp-1_2-fcs-classfiles.zip` を展開し、生成される `servlet.jar` を対象上記ディレクトリにコピーする。

```
$> cd △ [展開したディレクトリ]
$> cp △ servlet.jar △ /usr/local/src/dspace-1.2-source/lib
```

mail.jar

<http://java.sun.com/products/javamail/downloads/index.html> から JavaMail 1.3.1 をクリックし、javamail-1\_3\_1-upd.zip をダウンロードする。javamail-1\_3\_1-upd.zip を展開し、生成される mail.jar を上記ディレクトリにコピーする。

```
$> cd △ [展開したディレクトリ]
$> cp △ mail.jar △ /usr/local/src/dspace-1.2-source/lib
```

JDBC ドライバ (postgresql.jar)

PostgreSQL JDBC ドライバを、上記ディレクトリにコピーする。以前に行った PostgreSQL のコンパイルの途中で postgresql.jar というファイルがすでに生成されているはずなので、それを対象ディレクトリにコピーする。

```
$> cd △ usr/local/src/postgresql-7.4.2/src/interfaces/jdbc/jars/
$> cp △ postgresql.jar △ /usr/local/src/dspace-1.2-source/lib
```

インストール先ディレクトリの準備

DSpace をインストールするディレクトリを作成する。ここでは /usr/local/dspace にインストールする例を示す。

```
$> su △ -
Password : (パスワードを入力)
#> mkdir △ /usr/local/dspace
#> chown △ dspace.dspace △ /usr/local/dspace
#> exit
$
```

dspace.cfg の設定

DSpace のインストールのために、設定ファイル dspace.cfg の変更を行う。

```
$> cd △ /usr/local/src/dspace-1.2-source/config
$> vi △ dspace.cfg
```

以下の項目を探して、環境にあわせて設定を変更する。

dspace.dir

DSpace をインストールするディレクトリを設定する。

```
dspace.dir = /usr/local/dspace
```

dspace.url

DSpace にブラウザからアクセスする際の URL を設定する。

```
dspace.url = http://マシン名:8080/dspace
```

dspace.hostname

DSpace のホスト名を設定する。

```
dspace.hostname = 127.0.0.1 (サーバーのホスト名があるときはそちら)
```

#### dspace.name

DSpace のサイトのタイトルを設定する。この設定は初期状態のままでも動作するが、サイトのタイトルにも表示されるので、各機関の名前に変えるとよい。(日本語名を表示させたい場合はそのままでは文字化けするので、`native2ascii` コマンドでエンコードした文字列を設定する必要がある。たとえば「名古屋大学」は `¥u540d¥u53e4¥u5c4b¥u5927¥u5b66` とエンコードされる。)

```
dspace.name = DSpace at My University
```

#### Destinations for configuration files for other tools 部分

```
##### Destinations for configuration files for other tools #####  
  
# Comment out any lines corresponding to files you don't need, so they  
# don't get copied
```

とコメントされている部分の設定に関しては、パス中の `/dspace` となっている部分を、各環境にあわせて変更する。下に、`/usr/local/dspace` と変更する場合の例を示す。

```
config.template.apache13.conf=/usr/local/dspace/config/httpd.conf  
config.template.log4j.properties=/usr/local/dspace/config/log4j.properties  
config.template.log4j-handle-plugin.properties=/usr/local/dspace/config/  
log4j-handle-plugin.properties (実際は一行)  
config.template.oaicat.properties=/usr/local/dspace/config/oaicat.properties  
config.template.oai-web.xml=/usr/local/dspace/oai/WEB-INF/web.xml
```

#### db.url

データベースの接続先を設定する。PostgreSQL にアクセスするためのポート番号を初期状態から変更している場合は、"5432"の部分を適宜変更すること。`/dspace` の部分は、接続するデータベースの名前を入れる。ここでは、ローカルマシンに接続している。

```
db.url = jdbc:postgresql://localhost:5432/dspace
```

#### db.driver

データベースドライバのクラス名を設定する。ここは変更する必要はない。

```
db.driver = org.postgresql.Driver
```

#### db.username

PostgreSQL に接続するユーザー名を設定する。

```
db.username = dspace
```

#### db.passwd

PostgreSQL に接続するユーザーのパスワードを設定する。

```
db.password = dspace
```

#### mail.sever

利用できる SMTP メールサーバーを設定する。DSpace からの各種通知に用いられる。

```
mail.server = (利用できる SMTP サーバーの IP または名前)
```

mail.???

DSpace が各所で利用するメールアドレスを設定する。各環境にあわせて設定する。

```
mail.from.address = (メール送信の際, From: に表示されるアドレス)
feedback.recipient = (フィードバックフォームからのメールを受け取るアドレス)
mail.admin = (各所に表示される, 管理者のメールアドレス)
```

assetstore.dir 他

アップロードしたファイルを格納するディレクトリを設定する。初期状態では /dspace/assetstore のように設定されているので、 /dspace の部分を各環境に合わせて設定すること。この例では

```
assetstore.dir = /usr/local/dspace/assetstore
```

のように設定する。以下の history.dir, search.dir, log.dir も同様に環境に合わせて設定すること。

```
history.dir = /usr/local/dspace/history
search.dir = /usr/local/dspace/search
log.dir = /usr/local/dspace/log
```

upload.temp.dir

データをアップロードする際の一時ディレクトリを設定する。通常、ここを変更する必要はない。

```
upload.temp.dir = /tmp
```

upload.max

アップロードするファイルのサイズの上限を設定する。巨大なファイルを扱う場合は、ここを適宜変えることができる。(デフォルトでは, 536870912 = 512MB)

```
upload.max = 536870912
```

handle.prefix

ハンドルを設定する。この設定はさしあたり変更の必要はないが、将来ハンドルサーバーに登録したときは、割り当てられた値をここに設定する必要がある。本稿ではハンドルサーバーについて説明しないが、詳細については、<http://www.handle.net/> 等を適宜参照すること。

```
handle.prefix = 123456789
```

handle.dir

ハンドルサーバーファイルをインストールするディレクトリを設定する。パスを適宜書き換える。

```
handle.dir = /usr/local/dspace/handle-server
```

webui.site.authenticator

この設定は初期状態のままでよい。

webui.cert.ca

パスを各環境に合わせて設定する。ここでは

```
webui.cert.ca = /usr/local/dspace/etc/certificate-ca.pem
```

と設定する。

以下の項目は初期状態のままでよい。

```
webui.cert.autoregister
webui.submit.blocktheses
default.language
```

### DSpace のインストール

DSpace を解凍したディレクトリに移動する。ant コマンドでコンパイルを実行し、インストールまでを行う。

```
$> cd △ /usr/local/src/dspace-1.2-source
$> ant △ compile
```

ant compile コマンドの実行中に、

```
[javac] 注：一部の入力ファイルは推奨されない API を使用またはオーバーライドしている。
[javac] 注：詳細については、-deprecation オプションを指定して再コンパイルすること。
```

と表示されることがあっても、インストール作業に支障はない。

```
$> ant △ fresh_install
```

これで指定したディレクトリ下に DSpace がインストールされる。

この作業は、データベースサーバーである postmaster が起動した状態で行う必要がある。Jakarta Tomcat は、この時点で起動している必要はない。

### DSpace の初期設定コマンド

DSpace をインストールしたディレクトリの bin に移動し、index\_all コマンドで、検索インデックスを初期化する。また、create-administrator コマンドで、DSpace の管理者ユーザーを作成する。

```
$> cd △ /usr/local/dspace/bin
$> ./index-all
$> ./create-administrator
E-mail address: (管理者のメールアドレスを入力)
First name: (管理者の First name)
Last name: (管理者の Last name)
Password: (管理者のログインパスワード)
Again to confirm: (確認にもう一度パスワード)
Is the above data correct? (y or n): (y と入力)
```

この作業は、`postmaster` が起動した状態で行う必要がある。まだ `Jakarta Tomcat` が起動している必要はない。

### Tomcat から DSpace を実行する設定

DSpace をコンパイルしてできあがったものを、Tomcat が実行できるように設定する。できあがったものをひとまとめにした `war` ファイルが `build` ディレクトリの下に生成されているので、これを Tomcat が管理するディレクトリにコピーする。ふたつある `war` ファイルのうち、ひとつは DSpace の WEB サービス機能、もうひとつは OAI-PMH プロバイダの機能である。

```
$> cd △ /usr/local/src/dspace-1.2-source/build
$> cp △ dspace.war △ /usr/local/jakarta-tomcat-4.1.30/webapps/
$> cp △ dspace-oai.war △ /usr/local/jakarta-tomcat-4.1.30/webapps/
```

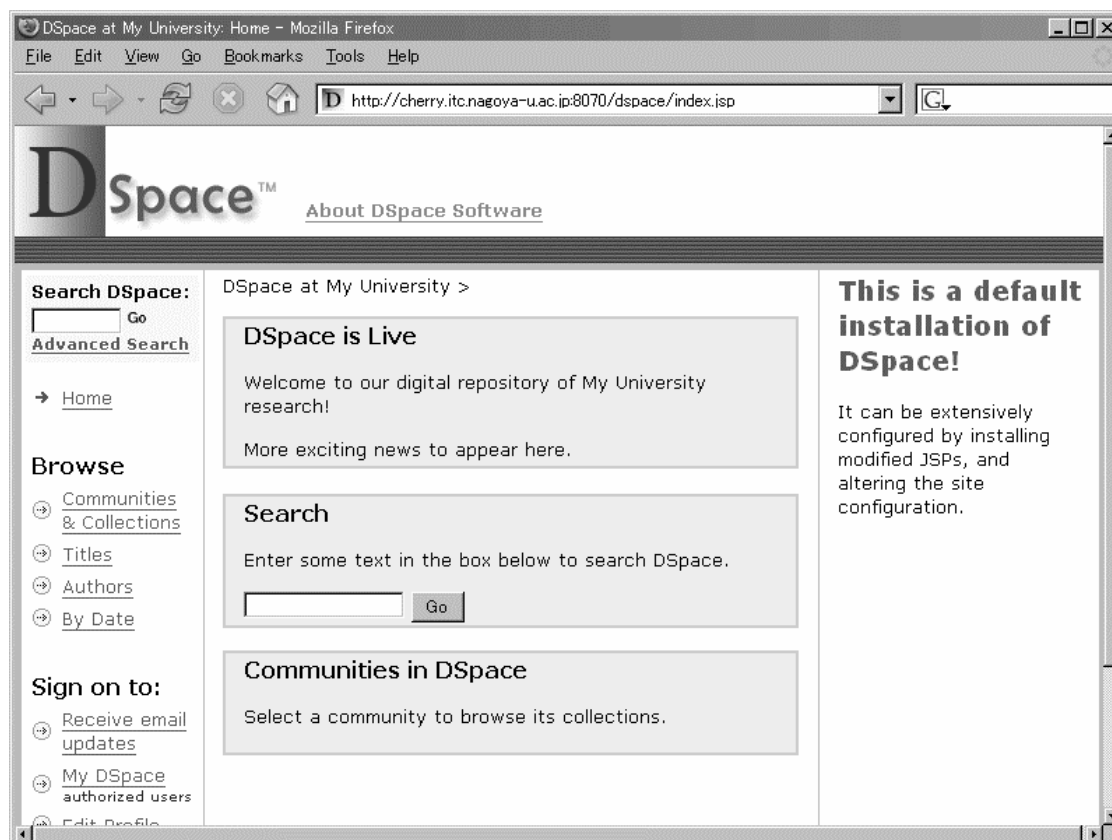
### 起動

Tomcat を起動する。`war` ファイルは自動的に展開されて、サービスが開始される。

```
$> whoami (現在のユーザーが dspace であることを確認する)
dspace
$> cd △ /usr/local/jakarta-tomcat-4.1.30/bin
$> ./startup.sh
```

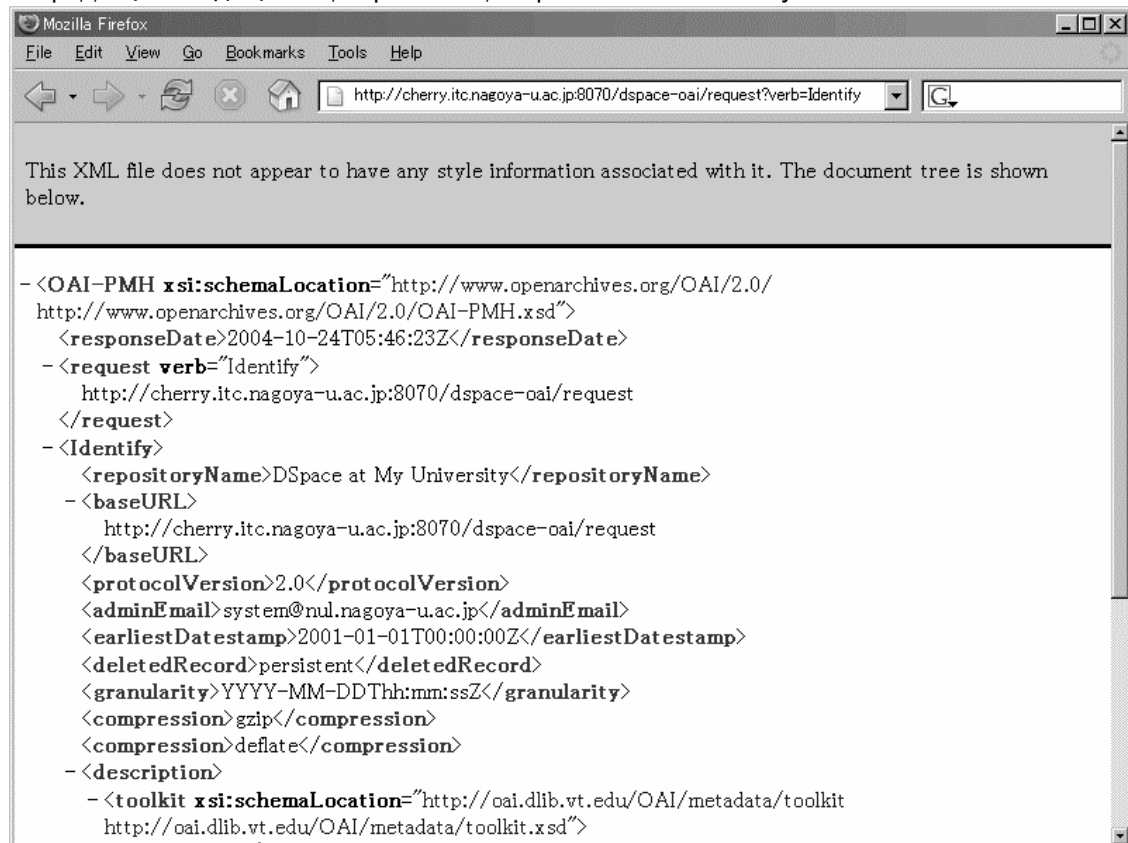
### 動作確認

ブラウザにて DSpace の URL を入力し、動作確認をする。  
`http://ホスト名:ポート/dspace/`



上のような画面が表示されたらインストールは完了である。

また、下のような URL も入力し、oai プロバイダ機能が動作するかも確認しておく。  
http:// ホスト名:ポート/dspace-oai/request?verb=Identify



## 2.1.6 補足資料

最終的にできあがる .bash\_profile の例

```
export JAVA_HOME=/usr/java/j2sdk1.4.2_04
export CLASS_PATH=$JAVA_HOME/lib/tools.jar
export JAVA_OPTS="-Xmx512M -Xms64M -Dfile.encoding=UTF-8"
```

(Apache ANT のインストール場所と ant コマンドの場所を設定)

```
export ANT_HOME=/usr/local/apache-ant-1.6.1
export ANT=$ANT_HOME/bin/ant
```

(PostgreSQL の環境を設定)

```
export POSTGRES_HOME=/usr/local/pgsql_d
export PGLIB=$POSTGRES_HOME/lib
export PGDATA=$POSTGRES_HOME/data
export MANPATH=$MANPATH:$POSTGRES_HOME/man
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$PGLIB
```

(コマンドの探索パスを設定。OS に最初からインストールされているバージョンのコマンドが優先的に実行されないように、\$PATH は最後にくっつける)

```
export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$POSTGRES_HOME/bin:$PATH
```

インストールに最終的に必要だったアーカイブファイル一覧

インストールする時点によって、ファイルのバージョン部分は多少変化することがある。

```

$ cd △ /usr/local/src
$ ls
(Apache ANT のアーカイブ)
apache-ant-1.6.1-bin.tar.gz
(DSpace のアーカイブ)
dSPACE-1.2.tar.gz
(JAVA のインストーラー)
j2sdk-1.4.2_04-linux-i586.rpm
(JAVA Activation Framework。activation.jar をここから得た)
jaf-1.0.2-upd.zip
(Jakarta Tomcat のアーカイブ)
jakarta-tomcat-4.1.30.tar.gz
(Java mail。mail.jar をここから得た)
javamail-1.3.1-upd.zip
(JSP API。servlet.jar をここから得た)
jsp-1.2-fcs-classfiles.zip
(PostgreSQL のアーカイブ)
postgresql-7.4.2.tar.gz

```

#### DSpace サーバー自動起動設定

サーバーの自動起動設定を、適宜行なう。例を参考に起動ファイルを作成すること。  
root ユーザーになって、以下のコマンドラインを実行する。

```

#> cd △ /etc/init.d
#> vi △ dSPACE

```

ここで下のように入力し、保存する。保存後、次の操作を行なう。

```

#> chmod △ 755 △ dSPACE
#> cd △ ../rc3.d
#> ln △ -s △ ../init.d/dSPACE △ S99dSPACE
#> cd △ ../rc6.d
#> ln △ -s △ ../init.d/dSPACE △ K99dSPACE

```

起動ファイル (dSPACE) の例

```

#!/bin/bash
# Startup script for the DSpace Server

export JAVA_HOME=/usr/local/j2sdk1.4.2_04

export POSTGRES_HOME=/usr/local/pgsql
export PGLIB=$POSTGRES_HOME/lib
export PGDATA=$POSTGRES_HOME/data
export MANPATH=$MANPATH:$POSTGRES_HOME/man
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:"$PGLIB"
export PATH=$PATH:$POSTGRES_HOME/bin

TOMCAT_BASE=/usr/local/jakarta-tomcat-4.1.30
POSTGRE_BASE=/usr/local/pgsql

```



```

case "$1" in
'start')
## -- DSpace Start -- ##
# Tomcat Start...
$TOMCAT_BASE/bin/startup.sh
echo "Tomcat is Starting..."

# PostgreSQL Start
su dspace -c $POSTGRE_BASE'/bin/pg_ctl start'
echo "PostgreSQL is Starting..."
;;

'stop')
## -- DSpace Stop -- ##
# Tomcat Stop...
$TOMCAT_BASE/bin/shutdown.sh
echo "Tomcat Stop..."

# PostgreSQL Stop
su dspace -c $POSTGRES_BASE'/bin/pg_ctl stop'

echo "PostgreSQL Stop..."
;;

*)
echo $"Usage: $0 {start|stop}"
exit 1
;;

esac
exit 0

```

## 2.2 DSpace 操作方法

DSpace は主に、論文などのコンテンツを投稿する機能、またその投稿されたコンテンツを承認して登録する機能、また、その登録されたコンテンツを検索・閲覧する機能で構成される。

以下に基本的な機能について述べる。応用的な機能については、適宜 <http://www.dspace.org/> 等をあたること。

### 2.2.1 トップページ

DSpace のトップページは、標準的には `http://ホスト名:ポート/dspace/` を指定する。ホスト名やポートは、WEB サーバー (Jakarta Tomcat) 等の設定に依存する。

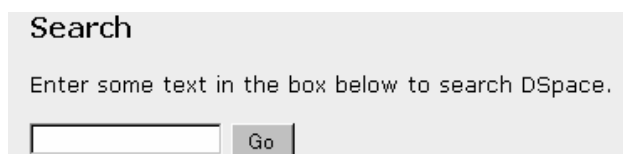
管理者用のトップページは、`http://ホスト名:ポート/dspace/dspace-admin` を指定する。

管理者ページを表示するには、管理者権限のユーザーでログインする必要がある。管理者アカウントは、DSpace のインストール直後に、(DSpace のインストールパス) `/bin/create-administrator` というコマンドを発行することで作成できる。

### 2.2.2 閲覧・検索

トップページから、メタデータによる検索が可能。検索語を入力して Go ボタンを押すことで検索が実行される。

(DSpace 1.2 から、登録コンテンツの全文検索もできるようになっている。)



また、登録されているコンテンツを一覧 (browse) することができる。並び順は、著者順、タイトル順、登録日付のうちから選択できる。ひとつのアイテムに複数の著者が登録されている場合、それぞれの著者ごとに同一タイトルが現われる。

DSpace 内のアイテムは、コミュニティという大枠で分けられ、ひとつのコミュニティの下はさらに複数のコレクションに分けて管理することができる。一覧表示は、コミュニティ、またその下のコレクション内のものがそれぞれ可能で、またそれらの中に区切ってメタデータの検索が可能。(コミュニティやコレクションの追加管理は、後述の管理機能で行う)

### 最初のコミュニティ

#### Community home page



### 2.2.3 ユーザー登録

DSpace の機能のいくつかを利用するには、ユーザー登録がされていて、そのユーザーとしてログインしていることが必要。認証されていない状態でログインが必要な機能を選ぶと、目的の画面に飛ばずに Log in DSpace という画面が表示される。

## Log In to DSpace

**New user?** [Click here to register.](#)

Please enter your e-mail address and password into the form below

E-mail Address:

Password:

Have you [forgotten your password?](#)

新しく利用者登録をする場合、この画面中の **Click here to register** というリンクをたどって、そこから電子メールのアドレスを入力して送信ボタンを押す。その後、DSpaceから自動送信されたメールを受け取り、そこに表示された URL にアクセスすることでユーザー登録が完了する。

## User Registration

If you've never logged on to DSpace before, please enter your e-mail address in the box below and click "Register".

E-mail Address:

### 2.2.4 プロフィールの編集

(この機能を利用するには、DSpace にログインしている必要がある)

**Edit Profile** というリンクをたどって、自分自身の情報とパスワードの変更ができる。変更後は更新ボタンを押して完了する。

## Registration Information

Please enter the following information. The fields marked with a \* are required.

First name\*:

Last name\*:

Contact telephone:

Please choose a password and enter it into the box below, and confirm it by typing into the second box. It should be at least six characters long.

Password:

Again to Confirm:

### 2.2.5 更新通知（購読）状態の確認

(この機能を利用するには、DSpace にログインしている必要がある)

DSpace の利用者は、あらかじめ指定したコミュニティにあたらしくアイテムが登録されたときに通知メールを自動的に送信してもらうことができる。

**Receive email updates** というリンク をたどって、更新情報をメールで通知するよう設

定されている（「購読している」）コミュニティの一覧を確認することができる。購読の設定そのものは、コレクションの閲覧画面など別の場所から行う。Unsubscribe ボタンで、購読のキャンセルを行うことができる。

## Your Subscriptions

To subscribe to a collection, visit the collection's home page, and click on the "Subscribe" button.

Below are the collections you are subscribed to. You will be sent an e-mail each day detailing new items that have become available in these collections. On days that no items have appeared, no e-mail will be sent.

テストコレクション	Unsubscribe
-----------	-------------

Remove All Subscriptions

[Go to My DSpace](#)

### 2.2.6 My DSpace

（この機能を利用するには、DSpace にログインしている必要がある）

ここには、ユーザーごとにカスタマイズされた情報が一覧表示される。

ユーザーが「投稿者」である場合、投稿を開始してまだ編集途中のアイテム（Unfinished Submissions）、投稿を完了してワークフローを通過中（承認を待っている状態）のアイテム（Submissions In Workflow Process）、投稿が承認されてデータベース上に登録されたアイテムの一覧（Accepted Submissions）が表示される。

### Unfinished Submissions

Title	Submitted to		
その他のメモ	テストコレクション	Resume	Remove...

### Submissions In Workflow Process

Title	Submitted to
DSpace1.2 インストールメモ	テストコレクション

## Your Submissions

Below are listed your previous submissions that have been accepted into the archive.

There is 1 item in the main archive that was submitted by you.

Date of Issue	Title	Authors
6-Nov-2004	DSpace1.2 インストールメモ	Yamamoto, Tetsuya

ユーザーが「購読者」である場合、See Your Subscriptions リンクから、現在購読中のコレクション一覧を表示させることができ、必要に応じてそれらのキャンセルができる。（「更新通知（購読）状態の確認」の項で説明したのと同じ）

ユーザーが「承認担当者」である場合、承認候補のアイテムでまだ誰も作業を受け持っていないもの (Tasks in the Pool), すでにこのユーザーが受け持ちを宣言して作業中のもの (Owned Tasks) の一覧が表示される。アイテムの承認過程のことを、DSpace ではワークフローと呼ぶ。

## Tasks in the Pool

Below are tasks in the task pool that have been assigned to you.

Task	Item	Submitted To	Submitted By	
Review Submission	DSpace1.2 インストールメモ	テストコレクション	sample user	Take Task

## Owned Tasks

Below are the current tasks that you have chosen to do.

Task	Item	Submitted To	Submitted By		
Review Submission	その他のメモ	テストコレクション	sample user	Perform This Task	Return to Pool

ユーザーがこれらの利用形態を同時に持っている場合、それらがすべて同一の My DSpace 画面に表示される。

### 2.2.7 投稿・ワークフロー

(この機能を利用するには、DSpace にログインしている必要がある)

投稿するには、My DSpace から [start submission] ボタンを押して投稿を開始するか、コレクション内のコンテンツ一覧から [submit to this collection] ボタンを押して投稿を開始する。前者の場合、投稿しようとするコミュニティを選ぶ画面が表示されるので、適宜選択する。投稿は必ずひとつのコミュニティに向けて行う。

投稿するには、投稿しようとするユーザーがそのコレクションに対して投稿が許可されていることが必要である。これらの権限は、後述の管理画面から行う。

投稿を進めるためには、各ステップで必要な項目を順に入力していく必要がある。いくつかは必須項目で、適切に入力してはじめて次のステップに進むことができる。入力途中でいったん中断し、日を改めて入力を継続することもできる。このときは、次回にログインしたときに、My DSpace から投稿途中のアイテム (Unfinished submissions) を選んで編集を継続することができる。下の図は、投稿が完了するまでの過程を示している。Describe (メタデータの記述), Upload (コンテンツ本体のアップロード), Verify (確認), License (ライセンスの承諾) という順になっている。



投稿プロセスの最後にはライセンスに承諾する画面があり、投稿者は [I grant the licence] ボタンを押すことで投稿を完了する。ライセンス文は、利用する機関ごとにカスタマイズする必要がある。(管理者向け:DSpace インストールパス/config/default.licence というファイルを書き換えることで、このライセンス文を変えられる)

I Grant the License

I Do Not Grant the License

投稿のプロセスが完了すると、投稿したアイテムがコレクションに登録されるための準備が完了する。ただし、そのコレクションにワークフロー（承認プロセス）が設定されている場合、これらのコレクションを通過させる権限をもったユーザーが許可するまで実際の登録は行われず。ワークフローの通過を待っているアイテムが現在いくつあるかは、投稿者の My DSpace から確認することができる。

管理者は、投稿されたアイテムが、最大で3つのワークフローを通過するように、コレクション単位で設定ができる。このうちのいくつか、または全部を省略するような設定も可能である（全部省略した場合、投稿して即登録される）。

それぞれのワークフロー上にアイテムが移動したときには、該当ワークフローの担当者に通知メールが自動送信される。ワークフロー担当者は、それぞれの My DSpace からこれらのアイテムを「受け持つ (Take Task)」かどうかを選択して、ワークフロー作業を開始する。

ひとつめのワークフローは、許可/却下 を判断するものである。ここで許可されたものはふたつめのワークフローに進み、却下されたときは投稿者の My DSpace 上に戻される。

ふたつめのワークフローは、許可/却下/データ編集 を判断するものである。ここで許可されたものはみつつめのワークフローに進み、却下されたときは投稿者の My DSpace 上に戻される。

みつつめのワークフローは、許可/データ編集 の判断をするものである。ここから却下を行うことはできない。

どのワークフロー上でも、[Do Later] ボタンと、[Return Task in Pool] ボタンが用意されている。前者は作業をあとで行うためにいったん中断するもので、後者は、ワークフロー上であって、特定の担当者が受け持っていない状態 (in the Pool) に戻すものである。

ワークフローをすべて通過すると、投稿者には登録完了の通知メールが送信される。ワークフロー内で却下処理をされた場合は、却下の理由を書かれたメールが送信され、投稿者の My DSpace 上に、編集途中のアイテムとして戻ってくる。投稿者は、このアイテムを破棄するか、内容を見直した上であらためて投稿することができる。

これ以降は、管理者用の機能について説明する。

## 2.2.8 管理メニュー：コミュニティの編集

左のメニュー内の Communities & Collections をたどって該当コミュニティを表示させたとき、管理者権限でログインしているときは左に Admin tools という枠が現れるので、そこからコミュニティの追加や編集ができる。

[Create Top-Level Community] ボタンで、コミュニティの新規作成を行う。

[Edit] ボタンで、コミュニティの情報を編集、または削除する。

[Create collection] ... コミュニティの下に新規にコレクションを作成する。

[Create Sub-community] ... コミュニティの下に入れ子になったコミュニティの作成をする。(DSpace バージョン 1.2 から)

**Admin Tools**

Create Top-Level Community...

[Admin Help...](#)

**Admin Tools**

Edit...

Create collection

Create Sub-community

[Admin Help...](#)

## Edit Community 123456789/1

[Help...](#)

Delete this Community...

Name:

Short Description:

Introductory text (HTML):

Copyright text (plain text):

### 2.2.9 管理メニュー：コレクションの編集

コミュニティ下のコレクションを表示させているとき、管理者でログインしているときは、コミュニティのときと同じく、[Edit] ボタンが現われる。ここから、コレクションの情報やワークフローを編集する。

## Edit Collection 123456789/2

Delete this Collection...

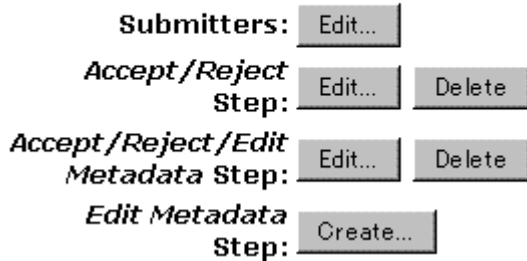
Name:

Short Description:

Introductory text (HTML):

編集画面の上半分は、名前、短い説明、紹介文、著作権表示、ライセンス、Provenance、ロゴ画像などの設定をする。下半分は、ワークフローに関する設定を行う。

## Submission Workflow



[Submitters] ... このコレクションに投稿できるユーザーを、グループ単位で編集できる。  
[Accept/Reject step] ... ひとつめのワークフローに所属するユーザーグループを編集できる。  
[Accept/Reject/Edit Metadata step] ... ふたつめのワークフローに所属するユーザーグループを編集できる。  
[Edit Metadata step] ... みつつめのワークフローに所属するユーザーグループを編集できる。

ここでリストに登録されたユーザーが、このコレクションの各ワークフロー担当となる。リストが空白の場合は、投稿プロセスでそのワークフローは省略される。

### [Item template]

これは、投稿されるアイテムの雛形を作成するためのボタンであり、このコレクションに投稿を開始するときには、あらかじめ必要な項目が埋まっていて記入を省力化できる。

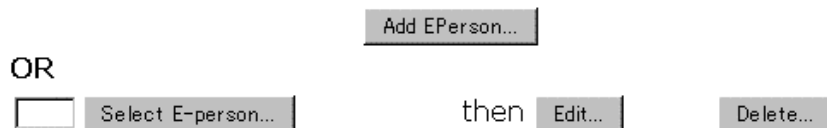
### 2.2.10 管理メニュー：ユーザーの編集

管理者権限でログインしているときは、左のメニューに **E-people** というリンクが現れるので、そこからユーザーの追加/編集/削除ができる。

## Administer EPeople

Choose an action:

[Help..](#)



[Add Eperson] からユーザーを追加するか、[Select E-person] ボタンから既存のユーザーを検索して、Edit または Delete を行う。

### 2.2.11 管理メニュー：アイテムの編集

左のメニューから **Items** というリンクをたどることで、そこからすでに登録されたアイテムの編集/削除ができる。

### 2.2.12 管理メニュー：アイテム項目の編集

左のメニューから **Dublin Core Registry** というリンクをたどることで、そこからアイテムの持つ項目をカスタマイズできる。

新たにエレメントを作成しても、投稿画面がそれにあわせて拡張されるわけではないので、必要に応じて、投稿画面そのものをカスタマイズする必要がある。(Java プログラミングの知識が必要)



### 2.2.13 管理メニュー：ファイル拡張子の編集

左のメニューから **Bitstream Format Registry** というリンクをたどることで、そこから新しい拡張子の登録や既存の拡張子の削除ができる。アップロードされたファイルの種類を **DSpace** が自動的に判断し、ブラウザに適切な **MIME** 情報を送信するのに役に立つ。

### 2.2.14 管理メニュー：グループの編集

左のメニューから **Groups** というリンクをたどると、そこからグループの追加/削除と、グループに属するユーザーの設定ができる。**Administrator** グループは管理者権限を持つユーザーで、通常編集の必要はない。**Anonymous** グループは、「誰でも」を意味するグループで、すべてのユーザーは暗黙にこのグループに属する。グループの中身が一見空でも編集の必要はない。

その他は、ワークフローの設定などに伴って自動的に作成されるものが多いので、その場合はここから編集する必要はない。

## Group Editor

[Help...](#)

Note that you do not need to manually add users to the "anonymous" group - all users are members implicitly.

Warning - if you try to delete a group that is referred to by an authorization policy or is a workflow group you will get an internal server error.

Create New Group

ID	Name		
1	Administrator	Edit...	Delete
0	Anonymous	Edit...	Delete
4	COLLECTION_1_ADMIN	Edit...	Delete

### 2.2.15 管理メニュー：ワークフロー

左のメニューから **Workflow** というリンクをたどることで、何らかの理由でワークフロー内に滞ったままになっているアイテムを取り除くことができる。ここからアイテムを探して **[Abort]** ボタンを押して処理する。ここで処理されたアイテムは、投稿者の **My DSpace** に戻される。

### 2.2.16 管理メニュー：権限の編集

左のメニューから **Authorization** というリンクをたどることで、そこから各種の詳細な権限を設定する。通常、これを利用しなくても運用に支障はない。

## Administer Authorization Policies

Choose a resource to manage policies for:

Manage a Community's Policies

Manage Collection's Policies

Manage An Item's Policies

Advanced/Item Wildcard Policy Admin Tool

### [Manage a Community's Policies]

コミュニティごとの権限を設定する。通常、どのコミュニティも、**Anonymous READ** (誰

でも一覧を見られる) に設定する。

#### [Manage Collection's Policies]

コレクションごとの権限を設定する。Anonymous DEFAULT\_ITEM\_READ は、ここに投稿されるアイテムは誰でも見ることができて、Anonymous DEFAULT\_BITSTREAM\_READ は、ここに投稿されるアイテムの内容までを誰でも見ることができることを意味する。

ADD は、ここに投稿できるグループを設定する。

#### [Manage an Item's Policies]

アイテムごとに、閲覧の権限を詳しく設定することができる。

#### [Advanced/Item wildcard Policy Admin Tool]

より細かい権限を設定する。

### 2.2.17 管理メニュー：ニュースの更新

DSpace のトップ画面に表示される文字列を編集する。[Top News] ボタンから中央に表示される文字を、[Sidebar News] ボタンから右部分に表示される文字をそれぞれ編集する。更新した内容は即時に反映される。

### 2.2.18 OAI-PMH プロバイダ機能

DSpace の起動しているマシンで、特別な URL にアクセスすることで、アイテムの情報を XML 形式で一括取得(ハーベスティング)することができる。この機能は主にコンピュータ間のメタデータの交換に利用されるが、一般的な WEB ブラウザで表示することも可能である。利用には、OAI-PMH2.0 プロトコルの知識が必要。ベース URL は、通常 `http://ホスト名:ポート/dspace-oai/request` であり、`http://ホスト名:ポート/dspace-oai/request?verb=Identify` 等の使い方をする。

## 2.3 DSpace 日本語対応

配布物からインストールした直後の DSpace の各画面表示はすべて英語であるが、内部的には UTF-8 でデータを持つので、日本語のメタデータを入力することは問題なく可能である。

ただし、いくつかの機能については、日本語に対応させるためのカスタマイズが必要である。

- 画面表示の日本語化
- ライセンス文の日本語化
- 電子メール雛形の日本語化
- 全文検索の日本語化

以下、これらの具体的な方法について記述する。

### 2.3.1 画面表示の日本語化

DSpace のソースコードを展開したものの中に、JSP という HTML の雛形ファイルのようなものが格納されているので、これを適宜書き換えることで画面表示の日本語化ができる。

ファイルは、ソースコード中の `jsp` というディレクトリに含まれている。

(例：`/usr/local/src/dspace-1.2-source/jsp/`)

ここにある、拡張子が `jsp` のファイルを書き換える。文字コードは UTF-8 で行うこと。

なお、既存のファイルを書き換えると以前のものに戻すことが難しくなるので、通常は `local` ディレクトリを作成し、この下に同じディレクトリ構造をコピーしてここを編集する方法をとる。DSpace は `local` ディレクトリにある JSP ファイルを優先的にインストールするので、初期状態のファイルを保ったまま、表示をカスタマイズしていくことができる。

例：

```
/usr/local/src/dspace-1.2-source/jsp/local/home.jsp を作ってこれを編集し、  
/usr/local/src/dspace-1.2-source/jsp/home.jsp の内容に代えることができる。
```

編集したファイルを実際の表示に反映させるには、DSpace を再度コンパイルして `war` ファイルを作成し、Jakarta Tomcat 等にインストールしなおす必要がある。

DSpace のソースを展開したディレクトリ上で

```
$ ant Δ build_wars
```

とし、ここで出来た `war` ファイルを Jakarta Tomcat の `webapp` ディレクトリにコピーし、Jakarta Tomcat を再起動する。この際、すでにインストールされたファイルは消去してから行う。

例：

```
$ cp Δ build/dspace.war Δ /usr/local/jakarta-tomcat-4.1.30/webapp/  
$ cd Δ /usr/local/jakarta-tomcat-4.1.30/webapp/  
$ rm Δ -R Δ dspace/  
$ cd Δ /usr/local/jakarta-tomcat-4.1.30/bin  
$ ./shutdown.sh  
$ ./startup.sh
```

ここでは DSpace のソースコードを編集する方法を紹介したが、何度も編集しながら結果を確認するときは、Jakarta Tomcat の下にインストールされた `jsp` ファイルを直接編集することでも、即時

に表示画面を変更することができる (例: /usr/local/jakarta-tomcat-4.1.30/webapps/dspace 下の jsp ファイル)。この方法をとるときは、他のカスタマイズのために DSpace をソースからインストールしなおすときに上書きされてしまわないように注意すること。

### 2.3.2 ライセンス文の日本語化

UTF-8 で書かれたファイルを正常に読み込めるようにカスタマイズする。修正するのは、src/org/dspace/core/ConfigurationManager.java。

```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.FileInputStream;
import java.io.InputStreamReader; <-----この行を追加
import java.io.InputStream;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.Properties;

(途中略)

// Load in default license
String licenseFile = getProperty("dspace.dir") + File.separator +
    "config" + File.separator + "default.license";
//BufferedReader br = new BufferedReader(new FileReader(licenseFile));
    この行をコメントアウトして、下のように修正
    BufferedReader br = new BufferedReader(new InputStreamReader(new
    FileInputStream(licenseFile), "UTF-8"));
```

ソースコードを修正したら、前段の説明と同じように、war ファイルを作成しなおし、Jakarta Tomcat にインストールして変更を反映する。

### 2.3.3 電子メール雛形の日本語化

電子メールの雛形ファイルは、config/emails/ 下にある各ファイルを書き換えて対応する (この場合は再コンパイル不要)。UTF-8 で保存すれば日本語化されたファイルも読み込み可能だが、問題が生じる場合は、ソースコードを一部修正して対応する。

修正するのは、src/org/dspace/core/ConfigurationManager.java。

```
public static Email getEmail(String template)
    throws IOException
{
    String subject = "";
    StringBuffer contentBuffer = new StringBuffer();
    BufferedReader reader = null;
    try
    {
        // Read in template (下のように修正)
        reader = new BufferedReader(new InputStreamReader(
            new FileInputStream(
                getProperty("dspace.dir") + File.separator +
                "config" + File.separator + "emails" + File.separator +
```

```
template), "UTF-8"));
```

ソースコードを修正したら、前段の説明と同じように、war ファイルを作成しなおし、Jakarta Tomcat にインストールして変更を反映する。

#### 2.3.4 検索の日本語化

メタデータ等の全文検索のために、DSpace は Lucene というインデックス作成・検索のためのライブラリを採用している。これは英単語を切り出してインデックスを作成するので、このままでは日本語データの検索ができない。下の URL にある方法を参考に、日本語のインデックスを切り出すような対応を行う。

参照 URL:

『DSpace で日本語メタデータの検索を実現する』  
<http://www12.ocn.ne.jp/~zuki/Japanization/dspace.html>

## 2.4 NII メタデータ記述要素への対応

DSpace の OAI プロバイダ機能に、NII メタデータ記述要素である `metadataPrefix=junii` 形式を追加するには、下のステップで作業をする。

### 2.4.1 ソースコードの改変とコンパイル

`dspace` のソースコードを展開したディレクトリの、`src/org/dspace/app/oai` に入る。下のファイルを新規に作成する。

`JUNIICrosswalk.java`

設定ファイル (`oaicat.properties`)

`config/oaicat.properties` の最終行に、下のような内容を追加する。

```
# List the supported metadataPrefixes along with the class that performs the
associated crosswalk
Crosswalks.oai_dc=org.dspace.app.oai.OAIDCCrosswalk
Crosswalks.junii=org.dspace.app.oai.JUNIICrosswalk <--追加
```

これで、OAICat フレームワークが `metadataPrefix=junii` という指定に対応したクラスを発見できるようになる。

設定ファイル (`dspace.cfg`)

`config/dspace.cfg` の最終行に、下のような内容を追加する。

各機関にあわせて値は書き換える。

```
##### junii oai configuration #####

junii.userid = A09999
junii.fano = FA009999
junii.institution = ¥u540d¥u53e4¥u5c4b¥u5927¥u5b66
```

上の例では、`junii.institution` は「名古屋大学」と書いてある。このように、日本語を設定するときは、適宜 `native2ascii` コマンドでエンコードする。

(`native2ascii` コマンドの例)

```
$> echo △ '名古屋大学' △ | △ native2ascii
¥u540d¥u53e4¥u5c4b¥u5927¥u5b66
```

上の手順を行ったら、

```
$> ant △ compile
$> ant △ build_wars
```

した後、`tomcat` を停止し、生成された `war` ファイルをインストール時にコピーした場所に上書きして `tomcat` を再起動する。このとき、すでに `tomcat` 自動生成したディレクトリのいくつかを削除しておく必要がある。

```
$> cd △ /usr/local/jakarta-tomcat-4.1.30/webapps/
$> rm △ -Rf △ dspace      (tomcat が作成したディレクトリを消す)
$> rm △ -Rf △ dspace-oai (同上)
```

### 2.4.2 ソースコード

【JUNIICrosswalk.java】 (太字は OAI Crosswalk.java と異なる部分)

```

package org.dspace.app.oai;

import java.util.Properties;

import
ORG.oclc.oai.server.crosswalk.Crosswalk;
import
ORG.oclc.oai.server.verb.CannotDisseminateFormatException;

import org.dspace.content.DCValue;
import org.dspace.content.DCDate;
import org.dspace.content.Item;
import
org.dspace.search.HarvestedItemInfo;

import
org.dspace.core.ConfigurationManager;

/**
 * An OAICat Crosswalk implementation
 * that extracts customized Dublin Core
 * from DSpace items into the junii
 * format.
 *
 * @author Yamamoto Tetsuya
 * @version $Revision: 1.3 $
 */
public class JUNIICrosswalk extends Crosswalk
{
    private String userid;
    private String fano;
    private String institution;
    private String baseURL;

    public JUNIICrosswalk(Properties properties)
    {
        super("http://ju.nii.ac.jp/oai
http://ju.nii.ac.jp/oai/junii.xsd");
        userid =
ConfigurationManager.getProperty("junii.userid");
        fano =
ConfigurationManager.getProperty("junii.fano");
        institution =
ConfigurationManager.getProperty("junii.institution");
        baseURL =
ConfigurationManager.getProperty("dspace.url");
    }

    public boolean isAvailableFor(Object nativeItem)
    {
        // [TODO] We have to return true/false
        according to metadata.
        return true;
    }

    private String taggedMetadata(String tag, String value) {

```

```

        // First do &'s - need to be careful
        not to replace the
        // & in "&" again!
        int c = -1;
        while ((c = value.indexOf("&", c +
1)) > -1)
        {
            value = value.substring(0, c) +
"&" +
            value.substring(c + 1);
        }

        while ((c = value.indexOf("<")) > -1)
        {
            value = value.substring(0, c) +
"&lt;" +
            value.substring(c + 1);
        }

        while ((c = value.indexOf(">")) > -1)
        {
            value = value.substring(0, c) +
"&gt;" +
            value.substring(c + 1);
        }
        return("<" + tag + ">" + value + "</"
+ tag + ">");
    }

    public String createMetadata(Object nativeItem)
        throws
        CannotDisseminateFormatException
    {
        Item item = ((HarvestedItemInfo)
nativeItem).item;

        // Get all the DC
        // DCValue[] allDC =
item.getDC(Item.ANY, Item.ANY,
Item.ANY);

        StringBuffer metadata = new
StringBuffer();

        metadata.append("<meta
xmlns=" + "http://ju.nii.ac.jp/oai" + " ")
        .append("<xmlns:xsi=" + "http://www.w3.org
/2001/XMLSchema-instance" + " ")
        .append("<xsi:schemaLocation=" + "http:
://ju.nii.ac.jp/oai " + " ")
        .append("<http://ju.nii.ac.jp/oai/
junii.xsd" + ">");

        metadata.append(taggedMetadata("code",
item.getHandle()));
        metadata.append(taggedMetadata("userid",
userid));
        metadata.append(taggedMetadata("fano",
fano));

        DCValue[] DCBuffer;
        String tempDate;

        DCBuffer = item.getDC("date",

```

```

"accessioned", Item.ANY);
    if (DCBuffer.length > 0) {
tempDate = DCBuffer[0].value;
        tempDate =
tempDate.substring(0, 4)
        + tempDate.substring(5, 7)
        + tempDate.substring(8, 10);
metadata.append(taggedMetadata("adate",
tempDate));
    }

tempDate = (new
DCDate(item.getLastModified())).toStri
ng();
tempDate = tempDate.substring(0, 4)
+ tempDate.substring(5, 7)
+ tempDate.substring(8, 10);

metadata.append(taggedMetadata("udate",
tempDate));
metadata.append(taggedMetadata("instit
ution", institution));

DCBuffer = item.getDC("title", null,
Item.ANY);
    if (DCBuffer.length > 0) {
metadata.append(taggedMetadata("title",
DCBuffer[0].value));
    }

DCBuffer = item.getDC("contributor",
"author", Item.ANY);
if (DCBuffer.length > 0) {
    for( int i=0; i<DCBuffer.length;
i++) {
metadata.append(taggedMetadata("creato
r", DCBuffer[i].value));
    }
}

DCBuffer = item.getDC("subject",
Item.ANY, Item.ANY);
if (DCBuffer.length > 0) {
    for( int i=0; i<DCBuffer.length;
i++) {
metadata.append(taggedMetadata("subjec
t", DCBuffer[i].value));
    }
}

DCBuffer = item.getDC("description",
"abstract", Item.ANY);
if (DCBuffer.length > 0) {
metadata.append(taggedMetadata("descri
ption", DCBuffer[0].value));
}

metadata.append( "<identifier
xsi:type='URL'" );
metadata.append( baseUrl + "/handle/" );

metadata.append( item.getHandle() );
metadata.append( "</identifier" );

metadata.append("</meta>");

return metadata.toString();
}

```



### 2.4.3 補足

metadataPrefix=oai\_dc 形式では、原則的にすべてのアイテムは OAI-PMH 上でデータを送信できるが、今回の metadataPrefix=junii 形式では、いくつかのメタデータは必須であったり形式が決まっており、この条件を満たしていないアイテムは OAI-PMH での送信対象からは外す必要がある場合がある。詳細は国立情報学研究所担当係に照会されたい。これを実現するためには、さきに挙げた JUNIICrosswalk.java に、さらにコードを加える必要がある。

具体的には、isAvailableFor メソッドが、常に true を返すのではなく、アイテムの内容によっては false も返すような改造を行う。

```
public boolean isAvailableFor (Object nativeItem)
{
// [TODO] We have to return true/false according to metadata.
return true;
}
```

このコードを、

```
public boolean isAvailableFor (Object nativeItem)
{
    Item item = ((HarvestedItemInfo) nativeItem).item;
    DCValue[] DCBuffer;
    DCBuffer = item.getDC(調べる element, 調べる qualifier, Item.ANY);
    .....
    if (...) {
        return true;
    } else {
        return false
    }
}
```

などといった形に書き換えていく。これの開発とデバッグについては、本稿の範囲を超える。

## 第3章 EPrints の導入と設定

本章では、東京大学情報基盤センターで実験的に行った EPrints のインストール、設定およびカスタマイズについて扱う。EPrints を導入して、デフォルトの状態でも運用することは可能であるが、その詳細については、別途「学術機関リポジトリ構築ソフトウェア実装実験プロジェクト」の Web サイトに掲載されている資料や公式ドキュメントをご覧ください。

### 3.1 導入手順

この節では、EPrints をインストールし、初期状態で動作させるまでの手順を解説する。なお、ここでは、以下の環境でインストールすることを前提としている。

- OS : FedoraCore 1
- httpd : Apache 2.0.47
- perl module : mod\_perl 1.99
- DB : MySQL 3.23
- EPrints : EPrints 2.3.3

これ以外の条件でインストールする場合には、下記の点に注意が必要である。

- OS が変わると、OS のインストール方法はもちろん、各種ソフトウェアのインストール先（ディレクトリ）が変わることがある。
- Apache 1.3 系を使用するときは、mod\_perl は 1.29 を使う。

#### 3.1.1 OS のインストール

##### (1) OS の入手

- ① <http://fedora.redhat.com/> から CD イメージをダウンロードする (disc3 までである)。
- ② 新品の CD-R を 3 枚用意し、CD-R 作成ソフトなどを用いてディスクイメージを CD-R に書き込む。

##### (2) OS のインストール

disc1 を、EPrints サーバにしたいマシンにセットし、マシンを再起動する。

- CD から起動しないときは、マシンの BIOS 設定で起動ドライブの優先度を変更する。
- インストールするパッケージは、EPrints のインストールマニュアルをよく読んで判断

すること。

- Apache, Perl, mod\_perl, MySQL のほか, これらに関係ありそうなものを選択すればよい。
- よくわからないときは, 「全部」としてもよい。
- mod\_perl と MySQL がうまくインストールされない場合があるが, 原因は不明。
  - そのときは, mod\_perl (3.1.2 を参照) と MySQL (3.1.3 を参照) を, あとで入れなおすことになる。
- マシンの IP アドレスやマスク値, デフォルトゲートウェイ, DNS サーバのアドレスなどはあらかじめ確認しておくこと。
- host 名も事前に考えておくこと。
- インストーラの出す質問に答えていけば, たいてい無事にインストールは完了する。

### (3) グループとユーザの作成

EPrints のインストールに必要なグループとユーザを作成する。

- ① eprints グループと eprints ユーザを作成する。

```
$> su
Password : password # ← root ユーザのパスワードを入力する
#> groupadd Δ eprints
#> adduser Δ -g Δ eprints Δ -d Δ /home/eprints Δ -m Δ eprints
#> passwd Δ eprints
Password : password # ← eprints ユーザのパスワードを入力する
```

- \* 以下, コマンドラインにて, "#>"で始まるコマンドは, root ユーザにて実行するものとする。

- ② mysql グループと mysql ユーザを作成する。

```
#> groupadd Δ mysql
#> adduser Δ -g Δ mysql Δ -d Δ /usr/local/var mysql
#> passwd Δ mysql
Password : password # ← mysql パスワードを入力する
```

### 3.1.2 mod\_perl のインストール (インストーラでうまくいかなかった場合のみ)

3.1.1 で指摘したように, インストールするように指定したつもりでも, パッケージがイ

インストール済みにならないことがある。その際は、手動で個別にインストールする。

#### (1) mod\_perl の入手

- <http://www.cpan.org/>から mod\_perl をダウンロードする。
  - ここでインストールした mod\_perl のバージョンは、mod\_perl-1.99\_14.tar.gz である。
  - ファイルの保存先は、/home/eprints/に設定している。各自、都合のよいところで構わない。
- インストールの前に、apxs がインストールされているかどうか (/usr/sbin/apxs が存在するか) を確認する。無ければインストールする。
  - disc3 の中に、httpd-devel-2.0.47-10.i386.rpm というファイルがあるので、これを使ってインストールする。
  - マウントされた CD のアイコンからたどっていき、rpm ファイルをダブルクリックする。
  - エラーで終了しても、/usr/sbin/apxs が作られていれば問題ない。ただし、CD が取り出せなくなっていることがあるので、umount -f /mnt/cdrom (または umount -l /mnt/cdrom) とするか、後でマシンを再起動して取り出すことになる。

#### (2) mod\_perl のインストール

下記の手順で、mod\_perl をインストールする。

```
#> cd Δ /home/eprints
#> gunzip Δ mod_perl-1.99_14.tar.gz
#> tar Δ xvf Δ mod_perl-1.99_14.tar
#> cd Δ mod_perl-1.99_14
#> perl Δ Makefile.PL Δ MP_APXS=/usr/sbin/apxs
#> make
#> make Δ test
#> make Δ install
```

#### 3.1.3 MySQL のインストール (インストーラでうまくいかなかった場合のみ)

3.1.2 と同様、インストールするように指定したつもりでも、パッケージがインストール済みにならないことがある。その際は、手動で個別にインストールする。

なお、3.1.5 以降で扱う、追加モジュールの中に、MySQL を必要とするものがあるので、事前にここでインストールを行う。

- ① MySQL のソースを解凍, 展開する。
- ② できたディレクトリに移動する。
- ③ 次のコマンドを実行する。

```
#> ./configure Δ --with-charset=ujis Δ --with-extra-charsets=all
# ← 何かエラーがでるようであれば, ./configure だけで実行
#> make
#> make Δ install
```

### 3.1.4 MySQL の初期化など

ここでは, MySQL を動作させるための準備作業を行う。なお, 以下のコマンドは, root ユーザで実行する場合以外は mysql ユーザにて実行する。

#### (1) データベースの初期化

```
#> /usr/local/mysql/scripts/mysql_install_db Δ --usr=mysql
#> chown Δ -R Δ mysql Δ /usr/local/var
#> chgrp Δ -R Δ mysql Δ /usr/local/var
```

#### (2) MySQL の起動

```
#> cd Δ /usr/local/mysql
#> ./bin/safe_mysqld &
```

#### (3) MySQL でのユーザの整備

- ① root で mysql にログインする。MySQL をインストール直後は, パスワードが設定されていないので, 次の手順で実行する。

```
#> /usr/local/mysql/bin/mysql Δ -u Δ root
mysql> SET PASSWORD FOR root@localhost=PASSWORD('new_password');
# ← 'new_password' に root のパスワードを入力する。
```

- ② mysql と eprints ユーザを追加する。

```
mysql> GRANT ALL PRIVILEGES ON *.* TO user@localhost
-> IDENTIFIED BY 'password' WITH GRANT OPTION;
# ← user と 'password' にそれぞれ mysql と eprints, およびそれぞれのパスワードを入力する
```

③ パスワードを設定した後のログイン方法は、次のとおりである。

```
/usr/local/mysql/bin/mysql -u user -p
# ← user には、設定したユーザ名を入力する。
```

### 3.1.5 必要なモジュールのインストール

EPrints では、mod\_perl に標準でついてくるモジュール以外に、下記のモジュールが必要である。

- Data::ShowTable
- DBI
- Mysql-Mysql Module
- MIME::Base64
- Unicode::String
- XML::Parser
- Apache
- Apache::Request a.k.a libapreq

#### (1) モジュールのインストール

上記モジュールをこの順番でインストールしていく。インストール方法は、おおむね次の要領である。

- ① <http://www.cpan.org/>から探してダウンロードしてくる。
- ② 解凍, 展開
- ③ できたディレクトリに移動
- ④ 次のコマンドを実行

```
#> perl △ Makefile.PL
#> make
#> make △ test
#> make △ install
```

#### (2) インストール時の注意点

インストールすべきモジュールのうち、いくつかのものについては、若干手間がかかったり、注意すべきものがあったので、ここで紹介しておく。

- Apache
  - Apache.pm のこと。mod\_perl をインストールしていればちゃんと入っているものであるが、EPrints を動かす段になってから EPrints のディレクトリへコピーする必要が出てくる場合もある。

- Apache::Request a.k.a libapreq
  - ファイル名は, libapreq2 (Apache の 2 系を使うときは, libapreq2 を使用のこと)。
  - まず, /usr/bin/apr-config があるか確認する。
    - ✧ 無ければ, disc3 の apr-devel-0.9.4-2.i386.rpm からインストールする。
    - ✧ エラーで終わっても, ファイルが作られていれば問題はない。
  - 次に, /usr/bin/apu-config があるか確認する。
    - ✧ 無ければ, disc3 の apr-util-devel-0.9.4-2.i386.rpm からインストールする。
    - ✧ エラーで終わっても, ファイルが作られていれば問題はない。
  - インストール時に, Apache::Test が古いという警告が出る場合がある。
    - ✧ Apache-Test-1.11.tar.gz からインストールする。
    - ✧ Apache-Test-1.11/t/のアクセス権を 777 にする。さらに上位のディレクトリのアクセス権も 777 にする必要がある場合もある。
 

```
#> cd Apache-Test-1.11
#> chmod 777 t
```
  - #> perl Makefile.PL 実行時に,
    - ✧ ExtUtils/XSBuilder.pm がないというメッセージが出た場合には,
      - ExtUtils-XSBuilder-0.25.tar.gz からインストールする。
    - ✧ Parse/RecDescent.pm がないというメッセージが出た場合には,
      - Parse-RecDescent-1.94.tar.gz からインストールする。
    - ✧ Tie/IxHash.pm がないというメッセージが出た場合には,
      - Tie-IxHash-1.21.tar.gz からインストールする。

なお, 下記のモジュールについては, 問題なくインストールできると思われる。念のため, 列挙しておく。

- Data::ShowTable
- DBI
- Mysql-MySQL Module
- MIME::Base64
- Unicode::String
- XML::Parser

### 3.1.6 GDOME のインストール

GDOME サポートは, EPrints の公式ドキュメントではオプションの扱いとなっているが, EPrints で日本語を扱うのであれば, これを利用することを推奨する。

### (1) ライブラリの追加

下記の 2 つのライブラリを `ftp://ftp.gnome.org/pub/GNOME/sources/libexm2/` からダウンロードしてインストールする。なお、本学でインストールしたライブラリのバージョンは 2.5.11 である。

- libxml2
- libxml2-devel

### (2) GDOME のインストール

`http://gdome2.cs.unibo.it/#downloads` から `gdome2` をダウンロードして、インストールする。なお、本学では、`gdome2-0.8.1` を利用した。

### (3) Perl モジュールの追加

下記の 3 つの Perl モジュールを `http://www.cpan.org/modules/by-module/XML/` からダウンロードして、インストールする。

- XML-LibXML-Common
- XML-Namespacesupport
- XML-GDOME

なお、本学ではそれぞれ、`XML-LibXML-Common-0.13`、`XML-Namespacesupport-1.08`、`XML-GDOME-0.86` をインストールした。

### (4) GDOME フラグの変更

ここですぐにできる作業ではないが、EPrints のインストール後、GDOME サポートを有効にするために、GDOME フラグをセットする必要がある。

具体的には、`/opt/eprints2/perl/lib/EPrints` にある `SystemSettings.pm` の 41 行目付近にある、`'enable_gdome' => 0,` を `'enable_gdome' => 1,` に変更する。

## 3.1.7 EPrints のインストール

EPrints プログラムをインストールする。

- ① MySQL が起動しているかどうか確認する。

```
#> ps Δ -ef Δ | Δ grep Δ mysql
root      12003      1  0  2004 ?        00:00:00 /bin/sh ./bin/safe_mysql
mysql     12017  12003   0  2004 ?        00:00:00 /usr/local/mysql/bin/mysqld
--defaults-extra-file=/usr/local/mysql/data/my.cnf --basedir=/usr/local/mysql
--datadir=/usr/local/mysql/data --user=mysql --pid-file=/usr/local/mysql/data
/eptest100.dl.itc.u-tokyo.ac.jp.pid --skip-locking
```

上記のようにプロセスが確認でき、mysql ユーザで `mysqld` が起動していれば OK である。



- ② `eprints-2.3.3.tar.gz` を解凍して展開する。ここでは、`/home/eprints` 以下に解凍し、インストールすることとする。

```
#> gzip Δ -d Δ eprints-2.3.3.tar.gz
#> tar Δ xvf Δ eprints-2.3.3.tar
#> cd Δ eprints-2.3.3
#> ./configure
```

- ③ `install.pl` の中身を確認し、Apache のバージョンを設定するところが空白になっていたら、「2」を補完する。

- ④ `install.pl` を実行する。

```
#> ./install.pl
```

- ⑤ 次に、アーカイブを作成する。この作業は、ユーザ `eprints` で行う。以下のコマンドを順番に実行する。なお、ID やホスト名は例として示してあるので、各自の状況に合わせて読み替えること。

```
#> su Δ eprints
$> cd Δ /opt/eprints2
# ← デフォルトでインストールした場合、このディレクトリにインストールされる。
$> bin/configure_archive
```

- `configure_archive` を実行すると、以下のような質問があるので、適切に答えていく。
  - ID `test01` # ← 作成するアーカイブの ID を指定する。
  - host 名 `eptest01.dl.itc.u-tokyo.ac.jp` # ← EPrint にアクセスするときのホスト名を指定する。
  - Webserver Port[80] # ← Web サーバを立ち上げる Port 番号を指定する。デフォルトは「80」である。
  - Alias # ← host 名のほかに名前をつけるときはここに設定する。
  - Admin Email `digilib@lib.u-tokyo.ac.jp` # ← EPrints サーバの管理者のメールアドレスを設定する。
  - Archive Name `test01` # ← アーカイブの名前を設定する。
  - (Mysql config) # ← ここからは MySQL 関係の設定。
  - Database Name `test01` # ← MySQL でのデータベースの名前。デフォルトではアーカイブ ID と同じ文字列が設定される。

- `Mysql host localhost #` ← MySQL がインストールされたホストを指定する。
- `Mysql port # #` ← MySQL のポート番号を指定する。デフォルトでよければ「#」とする。
- `Mysql socket # #` ← MySQL のソケット番号を指定する。デフォルトでよければ「#」とする。
- `Database User eprints #` ← MySQL データベースにアクセスするユーザを指定する。MySQL の初期設定時に作成したものを入力する。
- `Password #` ← MySQL データベースにアクセスするユーザのパスワードを指定する。MySQL の初期設定時に作成したものを入力する。
- `Create config files yes #` ← EPrints 用の設定ファイルを作成するかどうかを指定する。

\* 質問の後ろに [ ] でくくられた文字列が表示される場合は、何も入力せずに **Enter** を押すと、[ ] 内の文字列が設定される。

```
$> bin/create_tables Δ test01
$> bin/import_subjects Δ test01
$> bin/generate_static Δ test01
$> bin/generate_apacheconf
$> bin/create_user Δ test01 Δ username Δ email Δ admin Δ password
# ← アーカイブの最初のユーザ（管理者）を設定する。各引数は次のとおり。
username # ← アーカイブ ID が test01 のユーザ名
email # ← そのユーザのメールアドレス
admin # ← そのユーザの権限レベル（ここでは管理者権限）
password # ← そのユーザのパスワード
$> bin/generate_views Δ test01
```

- ⑥ アパッチの設定ファイルである `/etc/httpd/conf/httpd.conf` を編集する。オリジナルは別名でコピーしておく。なお、ホスト名やメールアドレスなどは例として示してあるので、各自の環境に合わせて変更すること。

```
$> cd Δ /etc/httpd/conf/
$> cp Δ httpd.conf Δ httpd.conf.org
$> gedit Δ httpd.conf
# ← エディタで httpd.conf を開き、下記の修正を行う。
LoadModule apreq_module modules/mod_apreq.so # ← 追加
User eprints # ← 設定
```

```

Group eprints # ← 設定
ServerAdmin digilib@lib.u-tokyo.ac.jp # ← web サーバの管理者のメールアドレスを
設定する
ServerName 133.11.199.100 # ← ホスト名か IP アドレス設定する
DocumentRoot "/opt/eprints2/archives/test01/html" # ← 適切に設定する
PerlModule ModPerl::Registry # ← 追加
Alias /perl "/opt/eprints2/cgi" # ← 追加
<Location /perl>
SetHandler perl-script
PerlResponseHandler ModPerl::Registry
PerlOptions +GlobalRequest
Options +ExecCGI
</Location> # ← 以上, 追加
(バーチャルホストを使用しない場合)
/opt/eprints2/archives/test01/cfg/auto-apache.conf から, 以下の部分をコピーしてく
る
<Location "">
ErrorDocument 401 /error401.html
(中略)
Redirect /change_user http://133.11.199.100/perl/users/home # ← この行は適切に
設定する
</Location>
(バーチャルホストを使用する場合)
Include /usr/local/eprints2/cfg/apache.conf # ← 追加
PerlRequire /opt/eprints2/bin/startup.pl # ← 追加

```

- ⑦ /opt/eprints2/perl\_lib/Eprints/SystemSettings.pm の中身を確認し, 'apache' => '1'  
であれば, 'apache' => '2' に修正する。

### 3.1.8 httpd の起動

- ・ httpd を起動する。

```
#> /etc/rc.d/init.d/httpd Δ start
```

なお, 終了する場合は,

```
#> /etc/rc.d/init.d/httpd Δ stop
```

- ・ 起動しようとする時、あれがない、これがないといった内容のエラーメッセージが出ることもある。その場合には、その都度、エラーメッセージで指定されたファイルをコピーする。ちなみに、テストサーバでは、

- /opt/eprints2/perl\_lib/にあるもの
  - ◇ Apache.pm
  - ◇ URI.pm
- /opt/eprints2/perl\_lib/Apache/にあるもの
  - ◇ AuthDBI.pm
  - ◇ Const.pm
  - ◇ Constants.pm
  - ◇ DBI.pm
  - ◇ Registry.pm

となっている。この辺は環境によって変わってくる可能性があるため、じっくりと丁寧に取り組むことが望ましい。

### 3.1.9 Indexer の起動

これから登録する EPrints のコンテンツに対して、インデックスを作成するためのスクリプト Indexer をあらかじめ起動しておく。

```
$> cd Δ /opt/eprints2/  
$> bin/indexer Δ start
```

### 3.1.10 EPrints の動作確認

ブラウザにて、EPrints の URL を入力し、動作確認を行う。図 3-1 のような画面が表示されれば、インストールは完了である。なお、以下の *localhost*, *port* には、それぞれ 3.1.6 の⑤で設定した値を入れること。

`http://localhost:port/`

## 3.2 EPrints の操作概要

ここでは、EPrints のインストールが完了した時点での、EPrints の概要および操作方法について簡単に触れる。より詳細な操作方法については、3.6 で扱う。

### 3.2.1 EPrints のホームページ

3.1.10 でも示したように、EPrints のホームページの URL は、`http://localhost:port/` によって指定する。`localhost` (ホスト名) や `port` (ポート番号) は、それぞれの環境によって異なる。

ホームページ (図 3-1) には、画面上部にメニューバー、画面中央上に簡易検索フィールド、その他のメニューが表示され、ここからさまざまな操作を開始することになる。

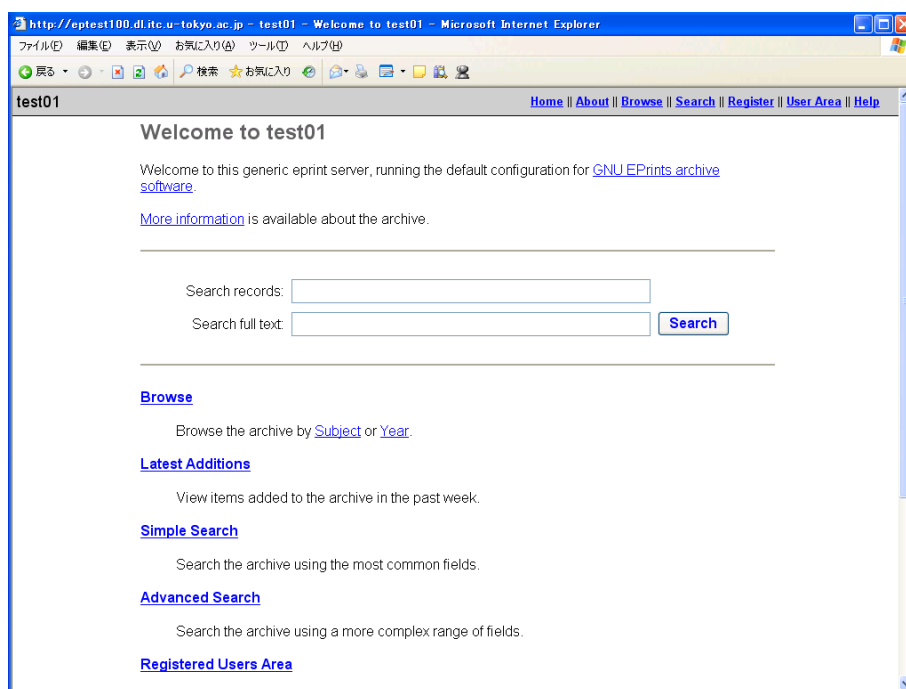


図 3-1 EPrints のホームページ

### 3.2.2 閲覧・検索

EPrints に登録されたコンテンツを検索するには、2つの方法がある。1つは、階層をたどって登録されているコンテンツを検索する方法、もう1つは、コンテンツのメタデータなどから検索する方法である。このほか、最近登録されたコンテンツを表示させる機能として、「Latest Additions」というメニューも用意されている。デフォルトでは、1週間以内に公開されたコンテンツが自動的に表示される仕組みとなっている。

## (1) 階層による検索 (Browse)

階層をたどって登録されているコンテンツを検索する場合には、「Browse」を利用する。デフォルトでは、Subject と Year による階層表示となっている (図 3-2)。

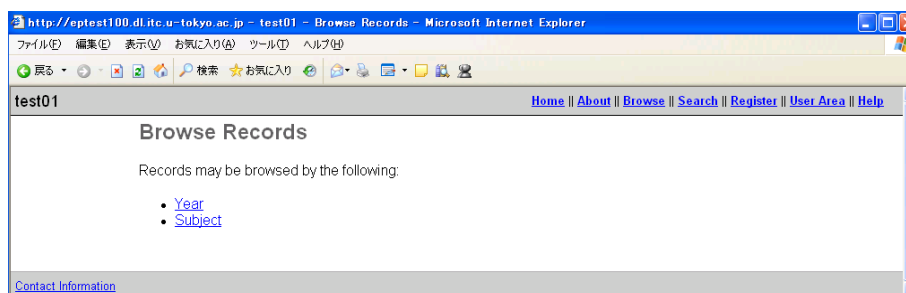


図 3-2 Browse の画面

なお、Browse メニューによって表示されるページは、静的なページなので、システム管理者が随時ページを再作成する必要がある。これについては、3.2.6 で扱う。

## (2) メタデータなどによる検索

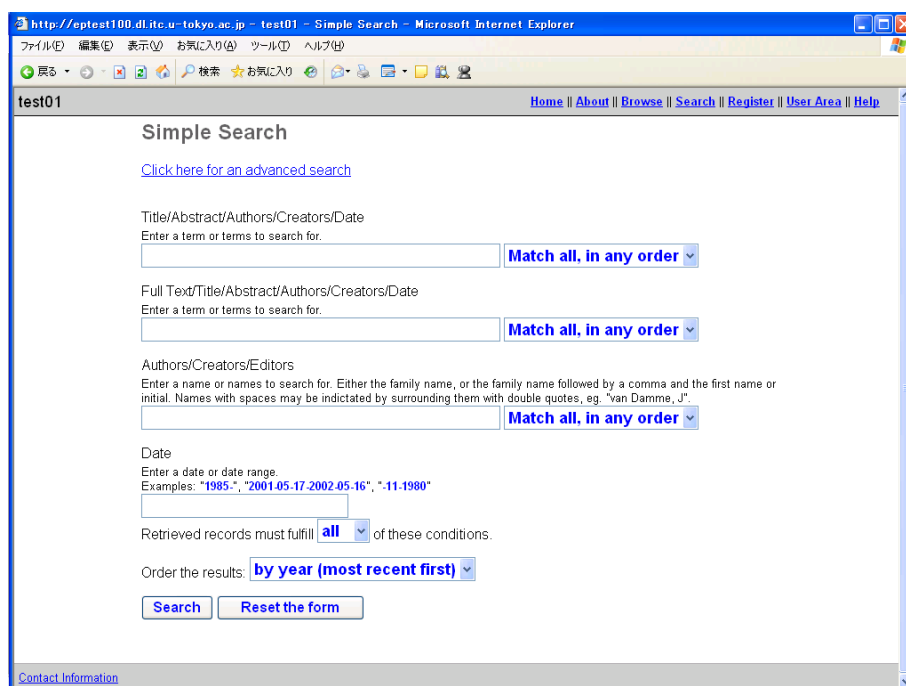


図 3-3 Simple Search 画面

登録されたコンテンツのメタデータなどから検索する場合、簡易検索 (Simple Search) と詳細検索 (Advanced Search) の 2 つの検索メニューが用意されている。

簡易検索では、タイトル、著者、抄録などから文献を検索できる。ホームページ中央上

に表示されている検索窓を利用するか、あるいは「Simple Search」というメニューをクリックして、図 3-3 から検索する場合の 2 通りの方法がある。一方、詳細検索であれば、さまざまなフィールドを指定して、文献を検索することができる。また、それぞれのフィールドにはプルダウン・メニューなども用意されている。

### 3.2.3 ユーザ登録

EPrints に登録されたコンテンツを閲覧することは誰でもできるが、EPrints にコンテンツを登録したり、いくつかのサービスを利用するためには、ユーザ登録が必要である。

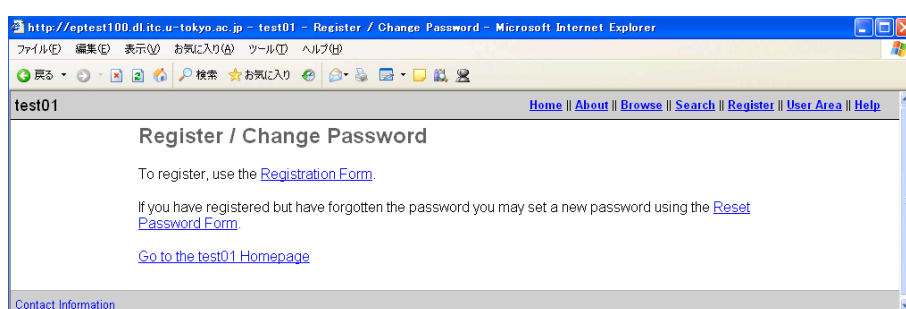


図 3-4 Register 画面

図 3-5 ユーザ登録画面

メニューの中から「Register」を選択し、Register 画面に進む（図 3-4）。その中から、「Registration Form」をクリックすると、ユーザ登録画面が表示される（図 3-5）。ここに、

必要な情報を入力し、「Register」ボタンをクリックする。ユーザ登録が無事済めば、登録完了画面が表示され、指定した E-mail アドレスに自動的にメールが送信される。Eメールの内容を確認の上、指定された URL にアクセスしてパスワードを有効にする。ユーザエリア (Registered Users Area) が表示されれば、ユーザ登録が完了となる。

### 3.2.4 ユーザエリア (Registered Users Area)

ユーザ登録が完了すると、EPrints に用意されたサービスをユーザエリア (Registered Users Area) で利用できるようになる。ホームページにある「Registered Users Area」メニューをクリックすると、ユーザ認証が求められる。ユーザ名とパスワードを入力してログインすると、図 3-6 のような画面が表示される。

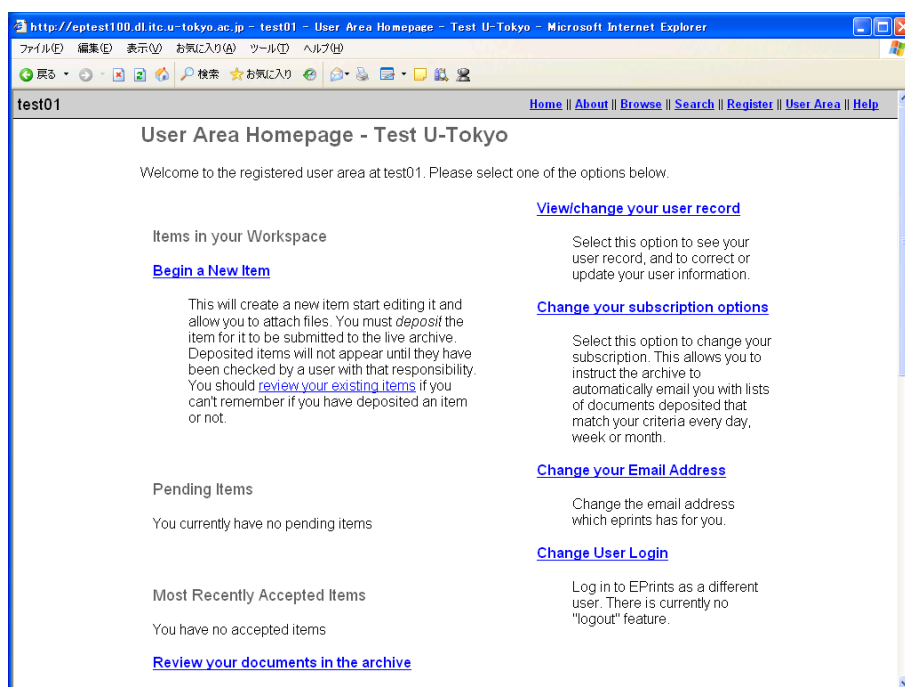


図 3-6 ユーザエリア

ユーザエリアでは、メニューが大きく左右に分かれて配置されている。左側はワークスペースと呼ばれ、コンテンツの新規登録 (Begin a New Item) や登録途中のコンテンツの編集など、コンテンツの登録に関連したメニューが用意されている。

一方、右側半分は、ユーザに関する情報を設定するメニューが用意されている。具体的には、ユーザ情報の編集 (View/change your user record), アラート機能の変更 (Change your subscription options), E-mail アドレスの変更 (Change your E-mail Address), 別ユーザでログイン (Change User Login) などがある。

また、EPrints では、一般ユーザのほか、編集・管理の権限を持つユーザが設定されてお



り、これらのユーザは、さらに別のメニューが用意されている。これについては、3.2.5以降で扱う。

#### (1) コンテンツの登録 (Begin a New Item)

新たにコンテンツを登録する場合は、メニューの「Begin a New Item」をクリックするところから始まる。登録までにはいくつかの画面 (ステップ) が用意されており、画面の内容に沿って必要な情報を入力していく。主なステップは、次のとおりである。

なお、登録手続きの詳細 (画面の遷移など) については、3.6を参照のこと。

##### ① Deposit Type

登録するコンテンツの種類 (EPrints Type) を選択する。

##### ② Core Bibliographic Information

登録するコンテンツのタイトル、著者名などを入力する。

##### ③ Publication Information

コンテンツの掲載された図書や雑誌のタイトル、巻・号、発行日など、出版に関する事項を入力する。

##### ④ Status of Item

コンテンツの現在の状況、すなわち Published (出版済)、In Press (出版中)、Submitted (投稿中)、Unpublished (未出版) といった情報を選択する。

##### ⑤ Abstract and References

コンテンツの抄録およびレファレンス (参考文献) を入力する。

##### ⑥ Subjects

コンテンツの主題を選択する。主題は、階層表示されるので、画面に従って選択すればよい。

##### ⑦ Additional Information

その他、必要に応じて、情報を入力する。

##### ⑧ Documents Attached to Item

コンテンツの登録 (アップロード) 作業を開始する。

##### ⑨ Document Information

登録するコンテンツのファイル形式などを指定する。

##### ⑩ Document File Upload

登録するコンテンツを選択して、アップロードする。ファイルは、ローカルディスクからのアップロードのほか、インターネット上に公開されている文書を指定して登録することもできる。

##### ⑪ Deposit Verification

ここまでのステップで登録した情報の内容を確認する。

すべてのステップが終わると、登録完了 (Completed Deposit) 画面が表示され、コンテ

ンツが登録される。なお、EPrints では、登録の途中で作業をいったん中断し、後から登録の手続きを再開できる仕組みが用意されている。その場合、ユーザエリアのワークスペースに登録途中のコンテンツが表示されるので、適切なメニューを選択すればよい。

## (2) ユーザ情報の編集 (View/change your user record)

ユーザ情報の編集やパスワードの変更は、このメニューをクリックして、図 3-7 の画面で行う。

Record for Test U-Tokyo

Please enter correct information about yourself for our records. This information will be useful to us and readers of your items. You don't have to supply all this information if you don't want to, you need only fill out those fields marked with a \* to start using the archive.

For instructions on how to change your email address, [click here](#).

**Password**  
Leaving this field blank will not affect the password. If you change your password you will be logged out.

**Hide Email**  
This option allows or prevents a users email being shown on their public summary page.  
 Make email visible to all.  
 Hide email to all except archive administrators.

**Name ★**

Family Name	Given Name / Initials	Lineage	Title
U-Tokyo	Test		

**Department**

**Organisation**

**Address**

図 3-7 ユーザ情報編集画面

## (3) アラート機能の変更 (Change your subscription options)

ユーザが関心のある subject を設定しておくこと、その主題にコンテンツが登録されたときに、メールで自動的に通知する機能である。

### 3.2.5 編集者・管理者機能

EPrints では、一般ユーザのほか、編集者 (Editor) および管理者 (EPrints Administrator) というユーザを設定することができる。これらのユーザが EPrints にログインすると、ユーザエリアに「Advanced Options」というスペースが追加表示される (図 3-8)。

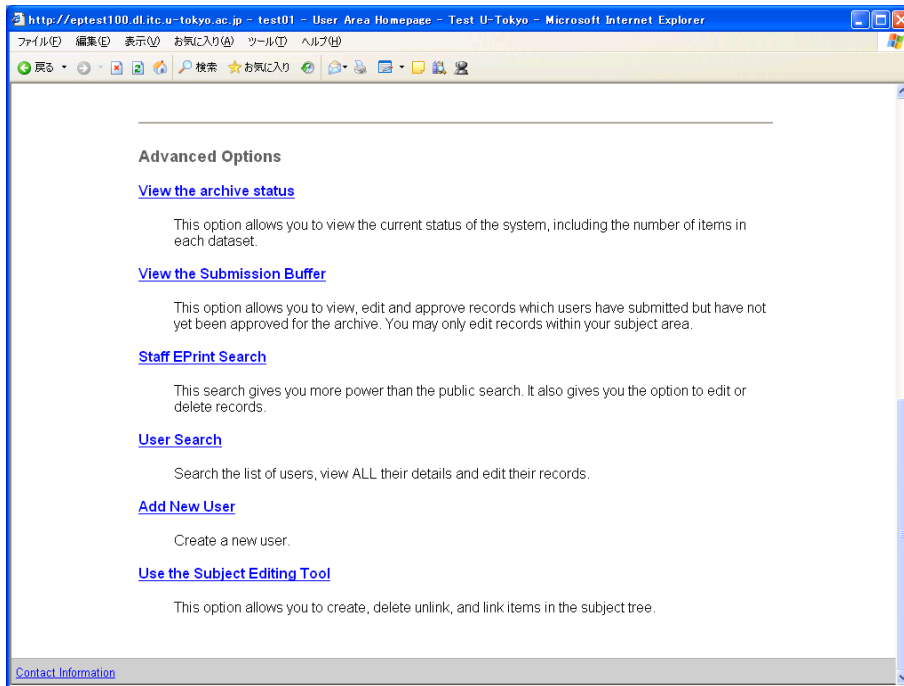


図 3-8 ユーザエリアの Advanced Options

(1) アーカイブ状態表示 (View the archive status)

EPrints アーカイブの現在の状況を表示することができる。EPrints ソフトウェアのバージョン、データベースの状態、ユーザ数、登録コンテンツ数、ディスク容量、インデックス生成日時などを確認することができる。

(2) バッファ表示 (View the Submission Buffer)

EPrints では、ユーザによって登録されたコンテンツは、いったんバッファに保存される (図 3-9)。



図 3-9 バッファ表示画面

これら公開許可待ちコンテンツは、編集者または管理者による確認を経てはじめて、閲

覧に供されることになる。具体的には、表示されているコンテンツのタイトルをクリックすると、図 3-10 のような画面が表示される。ユーザによって入力、選択された情報を確認し、必要があれば加筆・修正する。すべての作業を終え、「Move EPrint to Main Archive」をクリックすると、このコンテンツが一般公開される（検索の対象となる）。

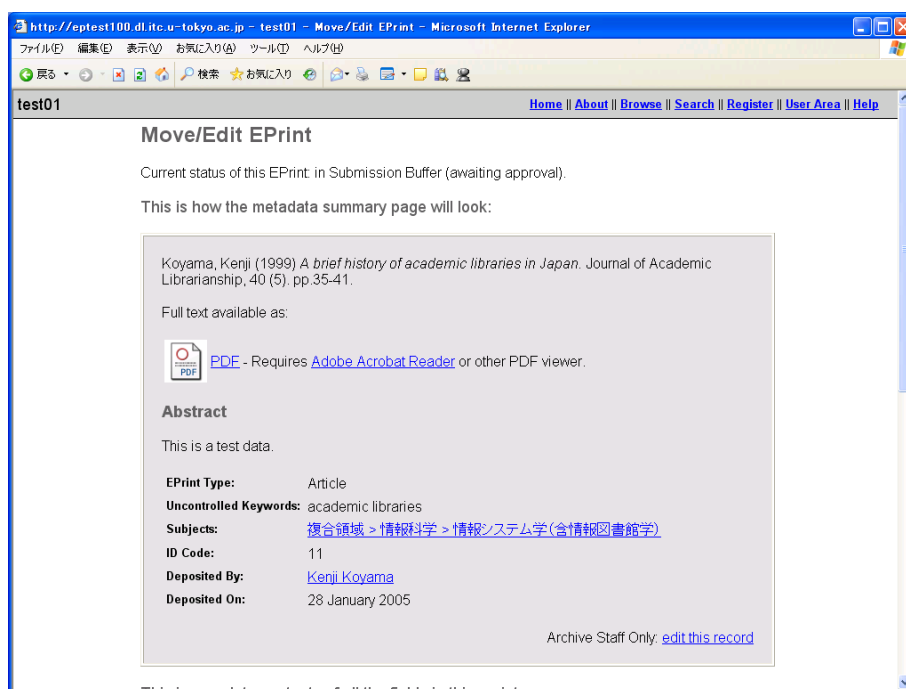


図 3-10 コンテンツの確認画面

### (3) 編集者・管理者用検索 (Staff EPrints Search)

一般ユーザよりも詳細に検索できる、編集者および管理者向けの検索メニューが用意されている。

### (4) ユーザ検索 (User Search)

編集者・管理者向けのユーザ検索機能である。

### (5) ユーザ追加 (Add New User)

管理者ユーザのみの機能。EPrints のユーザを個別に追加することができる。この機能を使うと、一般ユーザ、編集者、管理者の 3 つのタイプのユーザを指定することができる。

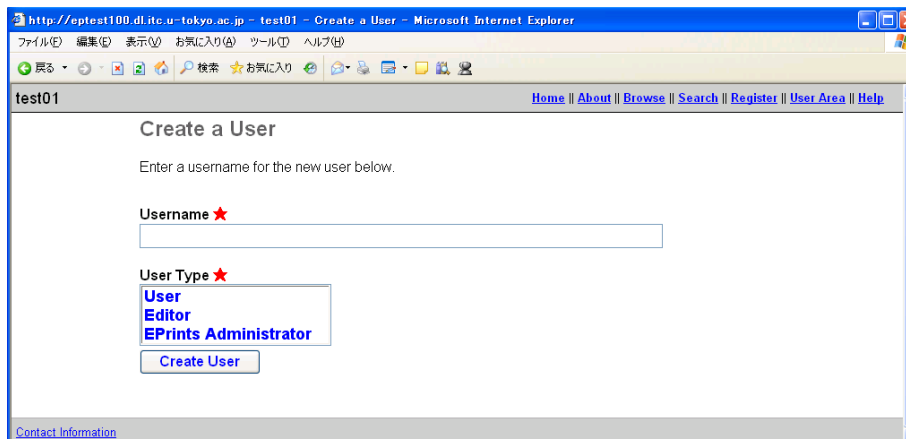


図 3-11 ユーザ追加画面

#### (6) サブジェクト編集 (Use the Subject Editing Tool)

管理者ユーザのみの機能。現在利用されている `subject list` に新たな項目を追加したり、削除することができる。メニューの選択後、画面の指示に従って作業をすると、簡単に `subject list` を編集することができる。なお、`subject list` そのものの入れ替え、変更については、3.5 を参照のこと。

### 3.2.6 その他

#### (1) 静的ページの作成

3.2.2(1)で取り上げた階層表示用の静的ページは、管理者がコマンドラインで別途作成する必要がある。

- ① EPrints をインストールしたディレクトリ (ここでは `/opt/eprints2`) に移動する。
- ② `generate_views` コマンドを実行する。

```
$> bin/generate_views △ test01 △ # ← 引数は各自で設定
```

#### (2) OAI-PMH リポジトリ

EPrints は、デフォルトで OAI-PMH のリポジトリとして動作する。ハーベストする際のベース URL を `http://localhost:port/per1/oai2` (`localhost` および `port` は各環境に合わせて設定のこと) と設定することで、ハーベスティングすることが可能である。

### 3.3 日本語対応

#### 3.3.1 EPrints の画面表示の日本語化

##### (1) 日本語用ファイルの準備 (その1)

/opt/eprints2/archives/test01/html/に/ja を作成し、/en 以下にあるファイルをコピーする。その後、contact.html, error401.html, index.html, information.html, vlit.html, help/index.html を必要に応じて日本語化 (翻訳など) する。

- 今回、EPrints がインストールされているディレクトリは、デフォルトである /opt/eprints2 になっている。この部分については、各自がインストールしたディレクトリを指定すること。
- 「test01」は、例として示したアーカイブ名であり、各自の状況に合わせて設定すること。

##### (2) 日本語用ファイルの準備 (その2)

① /opt/eprints2/archives/test01/cfg/の citations-en.xml と phrases-en.xml, template-en.xml をコピーして日本語化し、citations-ja.xml と phrases-ja.xml, template-ja.xml を作成する (ただし、citations-en.xml については、日本語化する場所はなさそうである)。

② phrases-ja.xml を下記のように修正する。

・ 修正前

```
<?xml version="1.0" encoding="iso-8859-1" standalone="no" ?>
<!DOCTYPE page SYSTEM "entities-en.dtd" >
```

・ 修正後

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE page SYSTEM "entities-ja.dtd" >
```

③ また、citations-ja.xml を下記のように修正する。

・ 修正前

```
<!DOCTYPE page SYSTEM "entities-en.dtd" >
```

・ 修正後

```
<!DOCTYPE page SYSTEM "entities-ja.dtd" >
```

(3) 日本語用ファイルの準備 (その3)

① /opt/eprints2/cfg/の system-phrases-en.xml をコピーして, system-phrases-ja.xml を作成し, 日本語化する。

② system-phrases-ja.xml の以下の部分を修正する。

・ 修正前

```
<?xml version="1.0" encoding="iso-8859-1" standalone="no" ?>
<!DOCTYPE page SYSTEM "entities-en.dtd" >
```

・ 修正後

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE page SYSTEM "entities-ja.dtd" >
```

(4) 日本語用ファイルの準備 (その4)

(3)と同じ/opt/eprints2/cfg/にある languages.xml のコメントアウト (#) をはずし, "yes"とする。

・ 修正前

```
# <lang Δ id="ja" Δ supported="no">Japanese</lang>
```

・ 修正後

```
<lang id="ja" supported="yes">Japanese</lang>
```

(5) 日本語用ファイルの準備 (その5)

/opt/eprints2/archivesにある test01.xml (アーカイブ名.xml) を次のように追加・修正する。

・ 修正前

```
<language >en</language>
<archivename language="en">eprints</archivename>
<defaultlanguage >en</defaultlanguage>
```

・ 修正後

```
<language >en</language>
<language >ja</language>
<archivename language="en">test01</archivename>
<archivename language="ja">test01</archivename> # ← 日本語用の名前を設定
```

```
<defaultlanguage >ja</defaultlanguage>
```

#### (6) httpd の再起動

以上の修正が終わったら、root ユーザで、httpd を再起動する。

```
$> su
Password : password # ← root ユーザのパスワードを入力する
#> /etc/rc.d/init.d/httpd Δ stop
#> /etc/rc.d/init.d/httpd Δ start
#> exit
```

#### (7) その他

- ① /etc/httpd/conf/の httpd.conf の 317 行目、PerlTransHandler EPrints::Rewrite の下をコピーして、/en/を/ja/に変更する。
- ② EPrints2.3.7 は、英語によるオリジナルのほか、日本語化キット (eprints-2\_3\_7-translation-ja.tgz) も用意されているので、そちらを利用するのもよい。下記の Web サイトからダウンロードできる。  
<http://wiki.eprints.org/w/EPrints2/ListOfTranslations>

### 3.3.2 検索の日本語化

EPrints では、メタデータ等の全文検索用インデックスを作成するためのプログラム (具体的には、ArchiveTextIndexingConfig.pm) が実装されている。しかし、これは英単語を切り出してインデックスを作成しているため、基本的には日本語データには対応していない。したがって、別途形態素解析器を実装する必要がある。

また、EPrints に登録されるコンテンツそのものを対象とした全文検索については、コンテンツごとに必要とされるフィルター・プログラムを適切に組み込めば、実現可能となっている。

これらの方法については、下記の Web サイトを参照していただきたい。

「学術機関リポジトリソフトウェアの日本語対応」

<http://www12.ocn.ne.jp/~zuki/Japanization/index.html> (参照 2005-02-18)



### 3.4 NII メタデータ記述要素への対応

この節では、NII のメタデータデータベース JUNII からのハーベスティングを可能とするための対応方法について述べるが、これには大きく 2 つの作業が必要となる。1 つ目は、junii スキーマによるハーベスティングを可能にするための修正・更新があげられる。もう 1 つは、JUNII に対応したフィールドの追加やそれを処理するためのプログラムの修正・更新である。具体的には、Title や Creator などのヨミ・フィールドの追加、Language フィールドの追加などがこれにあたる。もちろん、後者のような修正・更新をした場合には、改めて前者の修正・更新も必要となる。

以下では、上記の 2 つの作業について、本学で実施した例を参考に解説する。必要に応じて、各自の環境に合わせてカスタマイズしていただきたい。

#### 3.4.1 初期状態における JUNII 対応

ここでは、junii スキーマによるハーベスティングを可能にするための基本的な修正・更新を行う。修正するファイルは、

- /opt/eprints2/archives/test01/cfg/ArchiveOAIConfig.pm
- /opt/eprints2/perl\_lib/eprints/OpenArchives.pm

の 2 ファイルである。

##### (1) ArchiveOAIConfig.pm の修正

- ① # OAI 1.1 とコメントアウトされている部分の archive\_id を下記のとおり修正する。

```
# $oai->{archive_id} Δ = Δ "GenericEPrints"; # ← コメントアウトした  
$oai->{archive_id} = "UTtest01EP";
```

- ② # OAI-PMH 2.0 とコメントアウトされている部分の archive\_id を下記のとおり修正する。なお、こちらについては「.」を必ず入れなくてはならないことに注意すること（直上のコメントにもその旨の記述がある）。

```
# $oai->{v2}->{archive_id} Δ = Δ "UTtest01EP.OAI2"; # ← コメントアウトした  
$oai->{v2}->{archive_id} = "UTtest01EP.OAI2";
```

- ③ # OAI-PMH 2.0 とコメントアウトされている部分の以下の 3 箇所を、oai\_dc の例に倣って追加する。

```

# Exported metadata formats. The hash should map format ids to namespaces.
$oai->{v2}->{metadata_namespaces} =
{
    "oai_dc" => "http://www.openarchives.org/OAI/2.0/",
    "junii"  => http://ju.nii.ac.jp/oai/
};

# Exported metadata formats. The hash should map format ids to schemas.
$oai->{v2}->{metadata_schemas} =
{
    "oai_dc" => "http://www.openarchives.org/OAI/2.0/oai_dc.xsd",
    "junii"  => "http://ju.nii.ac.jp/oai/junii.xsd"
};

# Each supported metadata format will need a function to turn
# the eprint record into XML representing that format. The function(s)
# are defined later in this file.
$oai->{v2}->{metadata_functions} =
{
    "oai_dc"  => ¥&make_metadata_oai_dc_oai2,
    "junii"  => ¥&make_metadata_junii
};

```

- ④ # GENERAL OAI CONFIGURATION とコメントアウトされている部分の「latin1」となっている 5 箇所を「utf8」に修正する。そのまま修正してもいいが、ここではオリジナル部分を#を使ってコメントアウトし、コピーしたものを修正することとした。

```

# $oai->{content}->{"text"} Δ = Δ latin1( <<END );
$oai->{content}->{"text"} = utf8( <<END );
OAI Site description has not been configured.
END
$oai->{content}->{"url"} = undef;

# $oai->{metadata_policy}->{"text"} Δ = Δ latin1( <<END );
$oai->{metadata_policy}->{"text"} = utf8( <<END );
No metadata policy defined.

```

This server has not yet been fully configured.  
Please contact the admin for more information, but if in doubt assume that  
NO rights at all are granted to this data.

END

```
$oai->{metadata_policy}->{"url"} = undef;
```

```
# $oai->{data_policy}->{"text"} Δ = Δ latin1( <<END );
```

```
$oai->{data_policy}->{"text"} = utf8( <<END );
```

No data policy defined.

This server has not yet been fully configured.

Please contact the admin for more information, but if in doubt assume that  
NO rights at all are granted to this data.

END

```
$oai->{data_policy}->{"url"} = undef;
```

```
# $oai->{submission_policy}->{"text"} Δ = Δ latin1( <<END );
```

```
$oai->{submission_policy}->{"text"} = utf8( <<END );
```

No submission-data policy defined.

This server has not yet been fully configured.

END

```
$oai->{submission_policy}->{"url"} = undef;
```

```
# $oai->{comments} Δ = Δ [
```

```
#     latin1( "This system is running eprints server software ("
```

```
#         EPrints::Config::get( "version" )." ) developed at the "
```

```
#         "University of Southampton. For more information see "
```

```
#         "http://www.eprints.org/" )
```

```
# ];
```

```
$oai->{comments} = [
```

```
    utf8("This system is running eprints server software ("
```

```
        EPrints::Config::get( "version" )." ) developed at the "
```

```
        "University of Southampton. For more information see "
```

```
        "http://www.eprints.org/" )
```

```
];
```

- ⑤ サブルーチン `make_metadata_oai_dc_oai2` を流用して、サブルーチン `make_metadata_junii` を追加する。修正箇所は、下線で示した。

```
sub make_metadata_junii
{
    my( $eprint, $session ) = @_;

    my @dcdata = &eprint_to_unqualified_junii( $eprint, $session );

    my $archive = $session->get_archive();

    # Get the namespace & schema.
    # We could hard code them here, but getting the values from our
    # own configuration should avoid getting our knickers in a twist.

    my $oai_conf = $archive->get_conf( "oai", "v2" );
    my $namespace = $oai_conf->{metadata_namespaces}->{junii};
    my $schema = $oai_conf->{metadata_schemas}->{junii};

    my $junii = $session->make_element(
        "junii:dc",
        "xmlns:junii" => $namespace,
        "xmlns:dc" => "http://purl.org/dc/elements/1.1/",
        "xmlns:xsi" => "http://www.w3.org/2001/XMLSchema-instance",
        "xsi:schemaLocation" => $namespace." ".$schema );

    # turn the list of pairs into XML blocks (indented by 8) and add them
    # them to the DC element.
    foreach( @dcdata )
    {
        if ( $_->[0] eq "identifier" ) {
            $junii->appendChild( $session->render_data_element( 8,
                $_->[0], $_->[1], "xsi:type"=>"URL" ) );
        } else {
            if ( $_->[0] eq "type.nii" ) {
                $junii->appendChild( $session->render_data_element( 8,
```

```

        "type", $_->[1], "xsi:type"=>"Nil" ) );
    } else {
        $junii->appendChild( $session->render_data_element( 8,
$_->[0], $_->[1] ) );
    }
}

# produces <key>value</key>
}

return $junii;
}

```

⑥ サブルーチン `eprint_to_unqualified_dc` を流用して、サブルーチン `eprint_to_unqualified_junii` を追加する。初期状態での修正はほとんどなく、下記のとおりである。3.4.2以降の修正に伴い、修正が必要となる。

- ・ 参加組織 ID を出力させるため、「`my @dcdata = ();`」の直後に、「`push @dcdata, [ "fano", "FB00003" ];`」を一行追加する。
- ・ 査読を受けているかどうかの項目は必要ないと判断し、「`my $ref = "NonPeerReviewed";`」以下、「`push @dcdata, [ "type", $ref ];`」までコメントアウトする。

## (2) OpenArchives.pm

サブルーチン `sub make_header` の中で、下記の 2 箇所（下線部）を追加する。

```

sub make_header
{
    (中略)
    $header->appendChild( $session->render_data_element(
        6,
        "identifier",
        EPrints::OpenArchives::to_oai_identifier(
            $oai_id,
            $eprint->get_value( "eprintid" ) , "xsi:type"=>"URL" ) );
    (中略)
    $header->appendChild( $session->render_data_element(
        6,

```

```

        "datestamp",
        $datestamp ) );

# setSpec として, setnii=setA を出力させる
$header->appendChild( $session->render_data_element(
    6,
    "setSpec",
    "setA" ) );
(中略)
}

```

### 3.4.2 入力項目の修正・変更事例

次に、JUNII に対応したフィールドの追加やそれを処理するためのプログラムの修正・更新を行う。以下で紹介する事例は、EPrints の既定のフィールドを改造した一例である。修正するファイルは、最低限

- ArchiveMetadataFieldConfig.pm
- metadata-types.xml
- phrases-ja.xml
- ArchiveOAIConfig.pm

の4ファイルである。いずれも、/opt/eprints2/archives/test01/cfg ディレクトリに存在する。

場合によってはその他のモジュールの修正も必要になるが、いずれにせよ EPrints の公式ドキュメントにはあまり詳しい手引きは載っていない。

入力項目の変更等、MySQL のテーブル構造の変化を生じさせる修正を反映させるためには、アーカイブを一旦削除して、一から生成しなおす必要がある（ドキュメントには SQL に知識があれば、アーカイブを消去せずに変更できる旨記述してあるが、一方 SQL コマンドによる具体的手順の記述は見当たらない）。よって、運用前に入力項目等について充分検討、テストしておく必要があるものと思われる。

なお、アーカイブを削除した後、3.1.6 の⑤で示した、bin/configure\_archive 以下の一連のコマンドの実行および httpd の再起動を行う必要があるが、bin/configure\_archive においては「Create config files」の問いに対して「no」とする。さもないと、せっかく修正したファイルがデフォルトの内容に置き換えられてしまうことになる。

以下の記述では、ファイル修正後のアーカイブ削除、再生成の手順については記述を省略している。また、新規のフィールドを追加する場合、フィールド名は任意なので、この手順書と同じ名称にする必要はないことを申し添えておく。

(1) Title ヨミフィールドの追加

Title (タイトル) が日本語の場合, ヨミが必要なため, 別途フィールドを追加する。

① ArchiveMetadataFieldConfig.pm の修正

以下の記述を追加 (eprint section 内, title の次に追加) する。

```
{ name => "titletrans", type => "longtext", multilang=>0 },
```

② metadata-types.xml の

各 eprint type の中 (title の次) に, 以下の記述を追加する。

```
<field name="titletrans" />
```

③ phrases-ja.xml に以下の例のように補記する。

```
<ep:phrase ref="eprint_fieldname_titletrans">Title ヨミ</ep:phrase>
<ep:phrase ref="eprint_fieldhelp_titletrans">Title のヨミを分かち書きにより記述する。Title が日本語を含む場合のみ記述対象となる。
<br />Example: <span class="example">ジカン ノ レキシ</span>
<br />Example: <span class="example">エンジニア ノ タメノ スウガク</span>
<br />Example: <span class="example">エンジニア ノ タメノ スウガク ダイ 5 ハン
~</span>
<br />Example: <span class="example">セカイ ノ エコシステム ダイ 26 カン セカイ ノ
イリエ</span>
</ep:phrase>
```

④ ArchiveOAIConfig.pm の修正

サブルーチン sub eprint\_to\_unqualified\_junii 中に下線部を追加する。

```
push @dcdata, [ "title", $eprint->get_value( "title" ) ];
push @dcdata, [ "title.transcription", $eprint->get_value( "titletrans" ) ];
```

(2) 既存フィールドを利用した Creators ヨミの追加

Creator (作成者) が日本人名の場合, そのヨミが必要となる。EPrints における creators と editors は 4 つの要素で構成され, これらが内部的に処理されて creator (あるいは contributor) としてのメタデータが生成される。

すなわち, ArchiveMetadataFieldsConfig.pm 項目定義で「name」タイプの場合,

```
[ title ][ given name ][ family name ][ lineage ]
```

の4つである。このうち、title (称号, 肩書き) と lineage (家系) は、非表示にすることができる。また、family name (姓) と given name (名) の順序も変更することができる。

我が国では個人名に対し、title と lineage を使用することはないと思われるので、lineage に姓のヨミ、title(honourific)に名のヨミを当てることも可能である。その場合、

[ 姓 ] [ 名 ] [ 姓のヨミ ] [ 名のヨミ ]

とし、ヨミ部分は creator.transcription としてメタデータが生成されるようにする。なお、editors に関しても同様の方法を適用できる。

#### ① ArchiveMetadataFieldsConfig.pm の変更

- ・ 修正前

```
{ name => "creators", type => "name", multiple => 1, input_boxes => 4, hasid => 1,
input_id_cols=>20, family_first=>1, hide_honourific=>1, hide_lineage=>1 }
```

- ・ 修正後 : title と lineage を表示させるようにした。

```
{ name => "creators", type => "name", multiple => 1, input_boxes => 4, hasid => 1,
input_id_cols=>20, family_first=>1, hide_honourific=>0, hide_lineage=>0 },
```

#### ② ArchiveConfig.pm の変更

- ・ 修正前

```
$c->{field_defaults}->{input_name_cols} = {
    honourific=>8,
    given=>20,
    family=>20,
    lineage=>8
};
```

- ・ 修正後

```
$c->{field_defaults}->{input_name_cols} = {
    given=>10,
    family=>10,
    honourific=>20,
    lineage=>20
};
```



```
};
```

- \* 「=>」に続く値を変更することにより、テキストボックスの長さ（バイト）を指定することができる。適当なバイト長に設定のこと。

### ③ /opt/eprints2/perl\_lib/EPrints/MetaField/Name.pm の変更

- ・ 修正前

```
sub get_input_bits
{
    my( $self, $session ) = @_;

    my @namebits;
    unless( $self->get_property( "hide_honourific" ) )
    {
        push @namebits, "honourific";
    }
    if( $self->get_property( "family_first" ) )
    {
        push @namebits, "family", "given";
    }
    else
    {
        push @namebits, "given", "family";
    }
    unless( $self->get_property( "hide_lineage" ) )
    {
        push @namebits, "lineage";
    }

    return @namebits;
}
```

- ・ 修正後

```
sub get_input_bits
{
    my( $self, $session ) = @_;
```

```

    my @namebits;
    if( $self->get_property( "family_first" ) )
    {
        push @namebits, "family", "given";
    }
    else
    {
        push @namebits, "given", "family";
    }
    unless( $self->get_property( "hide_lineage" ) )
    {
        push @namebits, "lineage";
    }
    unless( $self->get_property( "hide_honourific" ) )
    {
        push @namebits, "honourific";
    }

    return @namebits;
}

```

次に creator のヨミを creator.transcription として出力するための修正を行う。

#### ④ /opt/eprints2/perl\_lib/EPrints/Utils.pm の修正

このファイル中の サブルーチン sub make\_name\_string において [family], [given], [lineage], [honourific]を組み合わせて creator エレメントを生成している。

よって、sub make\_name\_string では{family} {given}のみを組み合わせて creator を生成させ、また sub make\_name\_string を流用して sub make\_name\_string\_trans を作り、これにより {lineage} {honourific}のみを組み合わせ、creator.transcription を生成させることとする。

lineage に姓のヨミ， title(honourific)に名のヨミを当てると，流用して sub make\_name\_string\_trans を記述する際，多少書き換えが少なくて済む。

以下に，変更例を掲載する。

```
sub make_name_string
```

```

{
    my( $name, $familylast ) = @_;

    my $firstbit = "";
    if( defined $name->{given} )
    {
        $firstbit= $name->{given};
    }

    my $secondbit = "";
    if( defined $name->{family} )
    {
        $secondbit = $name->{family};
    }

    if( defined $familylast && $familylast )
    {
        return $firstbit.", ".$secondbit;
    }

    return $secondbit.", ".$firstbit;
}

sub make_name_string_trans
{
    my( $name, $familylast ) = @_;

    my $firstbit = "";
    if( defined $name->{honourific} && $name->{honourific} ne "" )
    {
        $firstbit = $name->{honourific}." ";
    }

    my $secondbit = "";

```

```
    if( defined $name->{lineage} && $name->{lineage} ne "" )
    {
        $secondbit .= ", ".$name->{lineage};
    }

    if( defined $familylast && $familylast )
    {
        return $firstbit." ".$secondbit;
    }

    return $secondbit.", ".$firstbit;
}
```

⑤ Archive0A1Config.pm の修正

サブルーチン sub eprint\_to\_unqualified\_junii 内を次のように修正する。

・ 修正前

```
my $creators = $eprint->get_value( "creators", 1 );
if( defined $creators )
{
    foreach my $creator ( @{$creators} )
    {
        push @dcdata, [ "creator",
EPrints::Utils::make_name_string( $creator ) ];
    }
}
```

・ 修正後

```
my $creators = $eprint->get_value( "creators", 1 );
if( defined $creators )
{
    foreach my $creator ( @{$creators} )
    {
        push @dcdata, [ "creator",
EPrints::Utils::make_name_string( $creator ) ];
    }
}
```

```

        push @dcdata, [ "creator.transcription",
            EPrints::Utils::make_name_string_trans( $creator ) ] ];
    }
}

```

### (3) フリーキーワード入力欄の変更

EPrints のフリーキーワード入力欄は、1 個のテキストボックスに複数のキーワードを入力できるようになっている。英語等の入力ではあまり問題ないのかもしれないが、ここでは「1 個のテキストボックスに 1 個のキーワード」を入力するよう修正してみることにする。また、それらを 1 つずつ、junii に subject として引き渡せるよう、カスタマイズする。

#### ① ArchiveMetadataFieldsConfig.pm の修正

- 修正前

```
{ name => "keywords", type => "longtext", input_rows => 2 },
```

- 修正後

```
{ name => "keywords", type => "text", multiple => 1, input_boxes => 4, input_add_boxes=>1,
input_cols=>10},
```

#### ② phrases-ja.xml において、入力 BOX1 つにつき、1 個ずつキーワードを入力する旨記述しておく。その方法は、(1) の③と同様である。

#### ③ ArchiveOAIConfig.pm の修正

サブルーチン sub eprint\_to\_unqualified\_junii 中に下線部を追加する。ここでは、「my \$subjectid;」に始まるプログラム群の後に追加した。

```

my $subjectid;
foreach $subjectid ( @{$eprint->get_value( "subjects" )} )
{
    (中略)
}

my $kw = $eprint->get_value( "keywords" );
if( defined $kw )
{

```

```
foreach my $keywords ( @{$kw} )
{
  push @dcdata, [ "subject", $keywords ];
}
}
```

(4) Language フィールドの追加

junii では必須項目となっている言語を指定する局面がないため、新たに追加する。

- ① ArchiveMetadataFieldConfig.pm の eprint section 内に以下を追加する。このセクション内なら場所はどこでも良いと思われる。

```
{name=>"lang",type=>"text", multiple => 1, input_boxes => 1, hasid=>1, input_add_boxes=>1, input_cols=>3}
```

\* 「language」という名称が EPrint ですでに規定されているようで(その役割は未確認)、且つ、この language の働きがよくわからないので、バッティングしないよう名称を「lang」とした。

- ② metadata-types.xml の各 type の中に、下記を追加する (required="yes"により必須項目扱い)。

```
<field name="lang" required="yes"/>
```

このとき、各 eprint type の<page name="hoge" />で、どの画面においてどの項目を表示するか決めているらしいので、ここでの記述位置は画面を意識しておくことが大切である。

\* 本学では<page name="pubinfo" />内の最後に記述した(つまり Publication Information 画面の最後の項目として表示される)。

- ③ phrases-ja.xml に以下のように記述する(あくまで参考例)。

```
<ep:phrase ref="eprint_fieldname_lang">言語</ep:phrase>
<ep:phrase ref="eprint_fieldhelp_lang">ドキュメント中で用いられている言語をコードで記述する。日英併記など複数言語を使用している場合は、[More Spaces]で記入欄を追加して記述すること。
<br />Example: <span class="example">日本語= jpn</span>
<br />Example: <span class="example">英語= eng</span>
```

</ep:phrase>

\* ファイル中の記述位置は多分上述の `metadata-types.xml` に依存すると思われるのでどこでも良いのではないかとと思われるが、本学の場合は ISSN の次に記述してみた。

#### ④ ArchiveOAIConfig.pm の修正

サブルーチン `sub eprint_to_unqualified_junii` 中の最後のあたりに、次のものを追加した。

```
my $lang = $eprint->get_value( "lang", 1 );
if( defined $lang )
{
    foreach my $language ( @{$lang} )
    {
        push @dcdata, [ "language..ISO639-2", $language ];
    }
}
```

#### (5) Type.nii の自動設定

EPrints のデータ登録の流れからして、どの局面で `junii` の `type.nii` に該当するデータを登録させるか、じっくりくる画面がなかった。データ登録時冒頭の `eprint type` の選択を `type.nii` の選択に全面変更することも考えられるが、

- Eprints のデータ登録画面は、どの `eprint type` を選択したかによってその後の画面が異なること
- `eprint type` のタイプ設定自体も生かしたかったこと

から、選択した `eprint type` によって、自動的に `type.nii` が設定されるように修正してみた。

以下では一例として

- 論文、図書の一部、モノグラフ、学位論文 => 研究成果-論文
  - 図書、特許、その他 => 研究成果-論文以外
  - 会議やワークショップのアイテム => 研究成果リスト-講演会等
- となるようにしてみる。

その方法としては、`ArchiveOAIConfig.pm` のサブルーチン `sub eprint_to_unqualified_junii` 中のこれまでの `type` 関係記述の次に以下を追加する。

```
my $type_nii ='研究成果-論文以外';
my $type_ep="";
```

```
$type_ep=$ds->get_type_name( $session, $eprint->get_value("type"));

if((~$type_ep=~/論文/)or($type_ep=~/一部/)or($type_ep=~/モノ/)) {$type_nii=' 研究成果-論文';}
if($type_ep=~/アイテム/) {$type_nii=' 研究成果リスト-講演会等';}

push @dcdata, [ "type.nii", $type_nii ];
```

(6) その他

① Publisher のヨミフィールドの追加

(1) で扱った、Title ヨミの設定にならって行った (type は text)。

② Creators, Editors の団体名用フィールドの追加

EPrints 所定の creators と editors (双方とも個人名用) とは別に、団体名記述用のフィールド (およびそのヨミ用フィールド) の追加も Title ヨミの設定と同様の手順で可能だが、複数の団体名が記述できるようフィールドタイプを multiple とした。

\* creators と editors について「個人名用」と「団体名用」を本来分ける必要はなく、個人名と団体名いずれであっても記述できるフィールドがあれば良い。しかし現時点で EPrints デフォルトの creators, editors を無効にしようとするエラーが発生してしまうため、暫定的にこのような対応をとっている。



### 3.5 その他カスタマイズ

EPrints は、3.3 の日本語対応および 3.4 の JUNII 対応といったカスタマイズのほか、EPrints の公式ドキュメントや関連資料を利用するなどして、さまざまなカスタマイズが可能である。もちろん、デフォルトでの運用も可能であるので、各大学の事情にあった導入および運用を考えていただきたい。しかし、いったん EPrints を利用した学術機関リポジトリを開始すると、その後のカスタマイズが難しくなる場面もあるので、その仕様はできるかぎり試行段階までで固める必要があることを申し添えておく。

さて、この節では、これまでに扱った内容以外のカスタマイズの例として、デフォルトで用意されている Subject list を変更した事例について紹介する。

もともと EPrints にデフォルトで用意されている Subject list は、アメリカ議会図書館分類表 (LCC) である。本学では、これを Subject list の日本語化とリスト変更の実験のため、科学研究費の分野体系に変更した。

Subject list で日本語を使用するには XML ファイルとして、list ファイルを作成しなければならない。そこで、EXCEL で Subjects リストを作成し、簡単な Perl スクリプトにかけて XML 形式のファイルを生成させる例を述べる。

#### ① EXCEL 表

subjectid, name, parents, depositable の 4 項目 (4 列) の表。記述については eprints-docs.txt を参照のこと。これを TAB 区切り text ファイルとして保存する。

#### ② perl スクリプト

基本的には以下のものである (ActivePerl による、パソコンで処理した)。

```
#####
# EXCEL 表=>TAB 区切り text データから
# Eprints の subjects の XML ファイルを生成する(言語は ja 固定)
# 生成されたファイルを UTF-8N, LF に変換してサーバに。
#####

if(@ARGV<2)
{
    die "入力ファイル名と出力ファイル名を指定してください";
}
```

```

# 【入力ファイル, 出力ファイルの OPEN】 *****
open(IN, $ARGV[0]) || die "$ARGV[0] :$!";
open(OUT, ">$ARGV[1]") || die "$ARGV[1] :$!";
#*****

print OUT "<eprintsdata>\n";

while(<IN>)
{
  chomp($_);
  print OUT "\t\t<record>\n";

  @buf=split(/\t/, $_);
  print OUT "\t\t<field name=\"$subjectid\">$buf[0]</field>\n";
  print OUT "\t\t<field name=\"$name\"><lang id=\"$ja\">$buf[1]</lang></field>\n";
print OUT "\t\t<field name=\"$parents\">$buf[2]</field>\n";
  print OUT "\t\t<field name=\"$deposable\">$buf[3]</field>\n";

  print OUT "\t</record>\n";
}

print OUT "</eprintsdata>\n";

# 【ファイルクローズ】 *****
close(IN);
close(OUT);
#*****

```

- ③ 出力ファイルは以下ようになる（その他の方法でも，以下の様な形式でファイルを作成すれば良い。なお，本来，階層構造のインデントによる表現は必須ではない）。

```

<eprintsdata>
  <record>
    <field name="subjectid">subjects</field>
    <field name="name"><lang id="ja">testja</lang></field>
    <field name="parents">ROOT</field>

```

```

        <field name="deposable">FALSE</field>
    </record>
    <record>
        <field name="subjectid">02</field>
        <field name="name"><lang id="ja">文学</lang></field>
        <field name="parents">subjects</field>
        <field name="deposable">FALSE</field>
    </record>
    (中略)
    <record>
        <field name="subjectid">0851</field>
        <field name="name"><lang id="ja">医用生体工学・生体材料学
</lang></field>
        <field name="parents">07</field>
        <field name="deposable">TRUE</field>
    </record>
</eprintsdata>

```

- ④ 出力結果をエディタを使って UTF8 に変換し, `subjects.xml` としてサーバの当該アーカイブの `cfg` 内 (ここでは `/opt/eprints2/archives/Eprints/cfg/`) に置く (ファイル名は任意)。

- ⑤ `import_subjects` の実行

```
$> bin/import_subjects Δ test01 Δ --xml
```

\* なお, 本学で上記を実行した際, デフォルトで設定されているパスで, `subjects.xml` を探すことができなかった (`…/perl/lib/…` に設定されているようであった) ため, `xml` ファイルの場所を下記のとおり, フルパスで指定した。

```
$> bin/import_subjects test01 Δ -xml /opt/eprints2/archives/Eprints/cfg/subjects.xml
```

### 3.6 カスタマイズ後の操作の流れ

EPrints の基本的な操作については、3.2 で解説したが、コンテンツの登録手順については、JUNII 対応させたこと (3.4 参照)、subject list を入れ替えたこと (3.5 参照) などから、初期状態とは異なった画面構成となっている。これらを踏まえ、この節では、コンテンツの登録手順を中心に、操作の流れを解説する。

#### (1) EPrints のホームページ

日本語環境のパソコン等から EPrints にアクセスすると、図 3-12 のように、日本語のホームページが表示される。



図 3-12 EPrints ホームページ

#### (2) コンテンツの登録

EPrints のホームページからユーザ・ページ (図 3-13) に入り、メニューの中から「新しい文献の登録」をクリックする。なお、本学の日本語環境下においては、「ユーザエリア」を「ユーザ・ページ」、「コンテンツ」を「文献」としている。

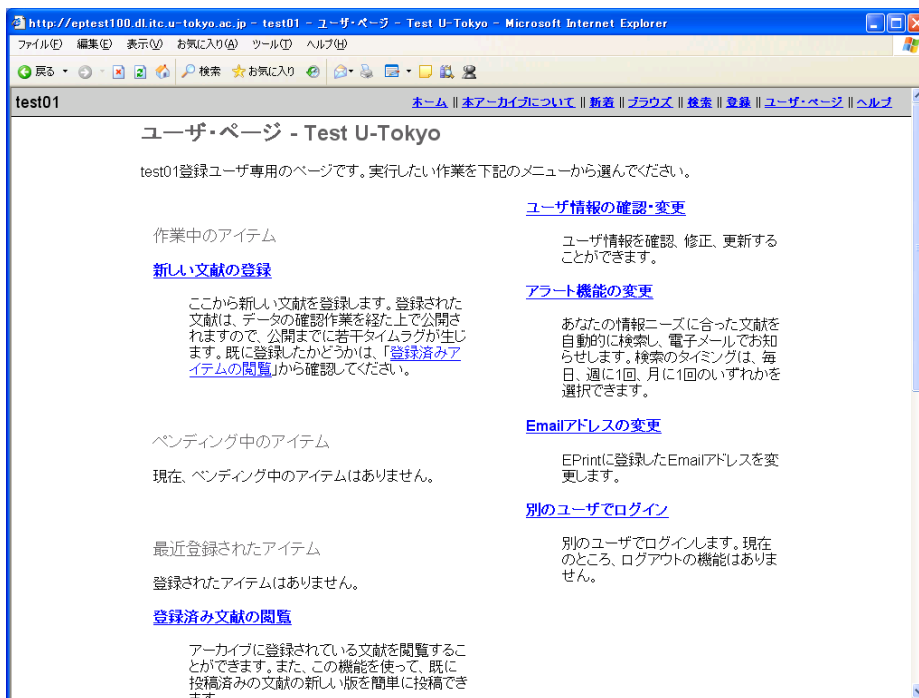


図 3-13 ユーザ・ページ

① 登録するファイルの種類

EPrints に登録する文献の種類 (EPrints Type) を選択する (図 3-14)。

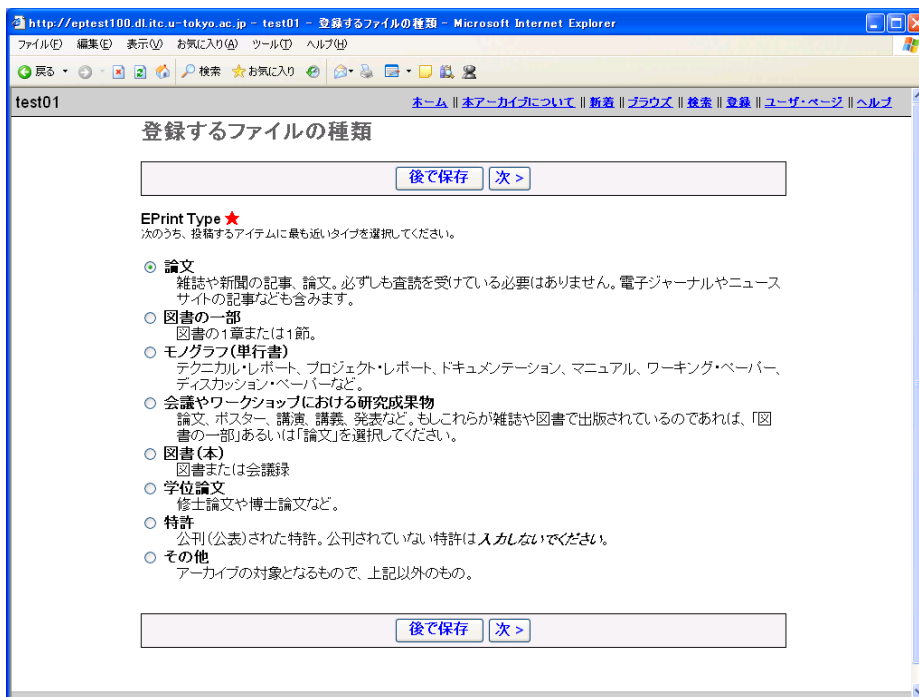
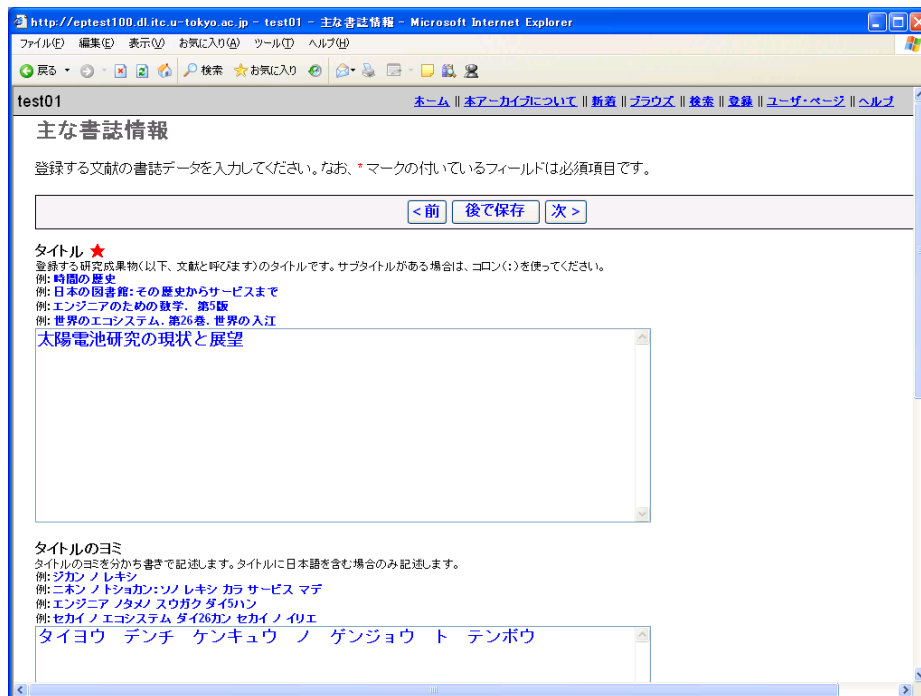


図 3-14 登録するファイルの種類

## ② 主な書誌情報

次に、登録する文献のタイトルや著者名などを入力する（図 3-15）。入力する内容は、登録するファイルの種類によって異なる。



The screenshot shows a web browser window with the URL <http://eptest100.dl.itc.u-tokyo.ac.jp/~test01>. The page title is "主な書誌情報" (Main Bibliographic Information). Below the title, there is a message: "登録する文献の書誌データを入力してください。なお、\*マークの付いているフィールドは必須項目です。" (Please enter bibliographic data for the document to be registered. Note that fields with an asterisk are required). There are navigation buttons: "< 前" (Previous), "後で保存" (Save later), and "次 >" (Next). The "タイトル" (Title) field is marked with a red asterisk and contains the text "太陽電池研究の現状と展望" (Current status and prospects of solar cell research). Below it, there is a "タイトルのコミ" (Title comment) field containing "タイヨウ デンチ ケンキュウ ノ ゲンジョウ ト テンボウ" (Taiyō Denchi Kenkyū no Genjō to Tenbō).

図 3-15 主な書誌情報

## ③ 出版情報

投稿する文献が掲載された図書・雑誌のタイトル、巻・号、発行日などの出版情報を入力する（図 3-16）。なお、JUNII 対応の 1 つである言語コードの選択フィールドも、本画面の下部に用意している。

## ④ 文献の現在の状況

この画面では、「セット」（NII の JUNII 対応の 1 つ）、文献の「現在の状況」、「査読」の有無について尋ねている（図 3-17）。

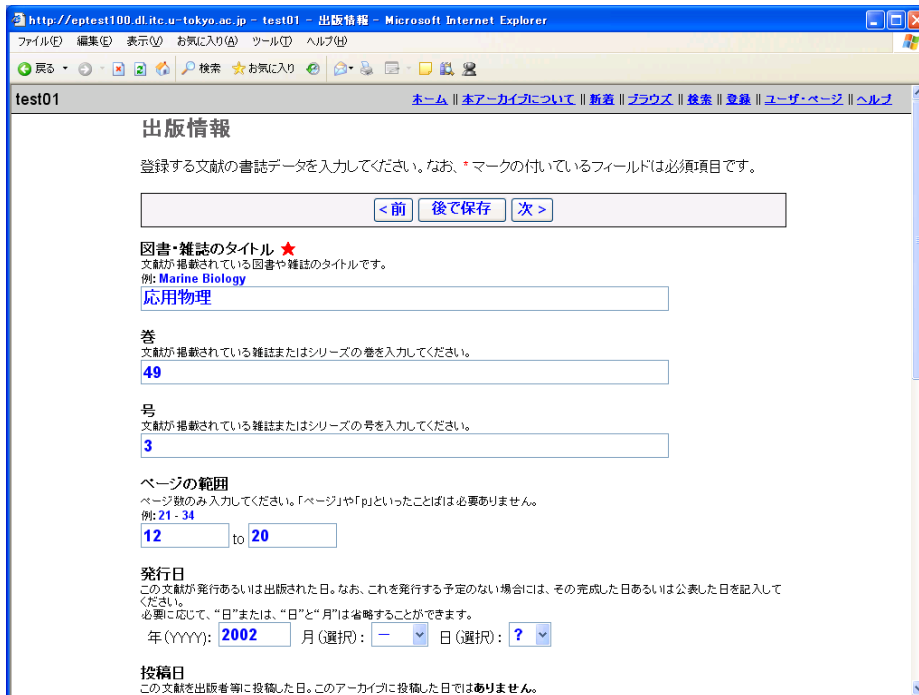


図 3-16 出版情報

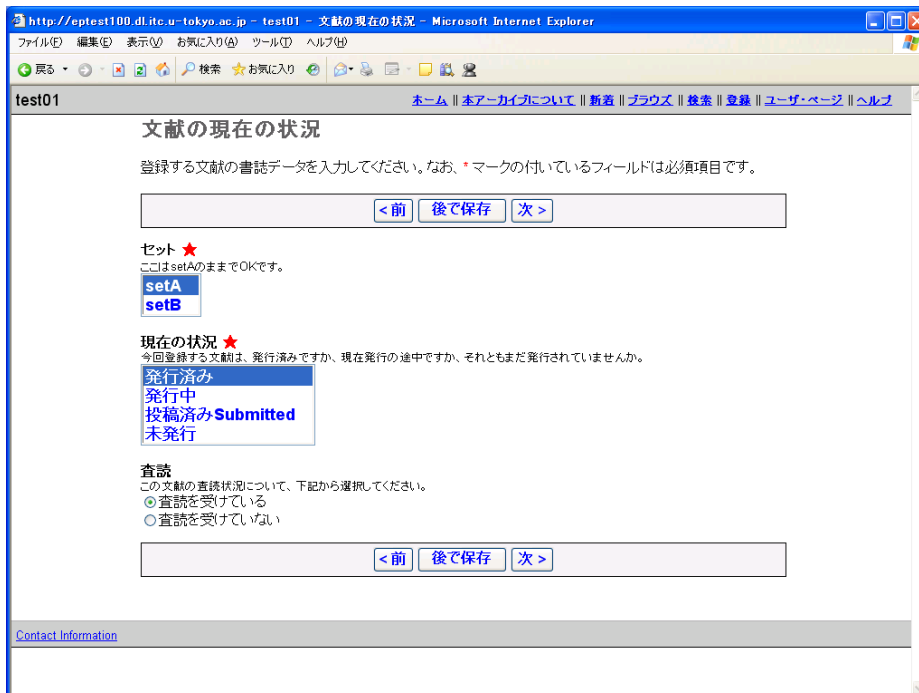


図 3-17 文献の現在の状況

⑤ 抄録・レファレンス (参考文献)

登録する文献の「抄録」, 「レファレンス (参考文献)」, 「キーワード」 (新たに追加した項目) を入力する (図 3-18)。

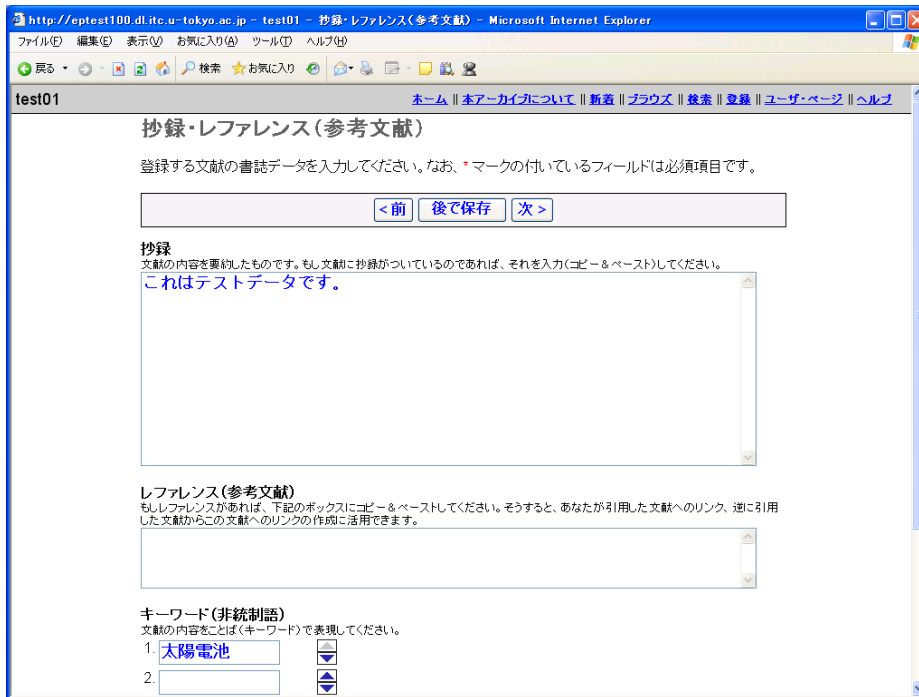


図 3-18 抄録・レファレンス (参考文献)

⑥ 主題

文献の内容に合った主題をリスト（ツリー構造になっている）の中から選択するか、クイック検索を使って主題を検索する（図 3-19）。主題は複数選択することもできる。

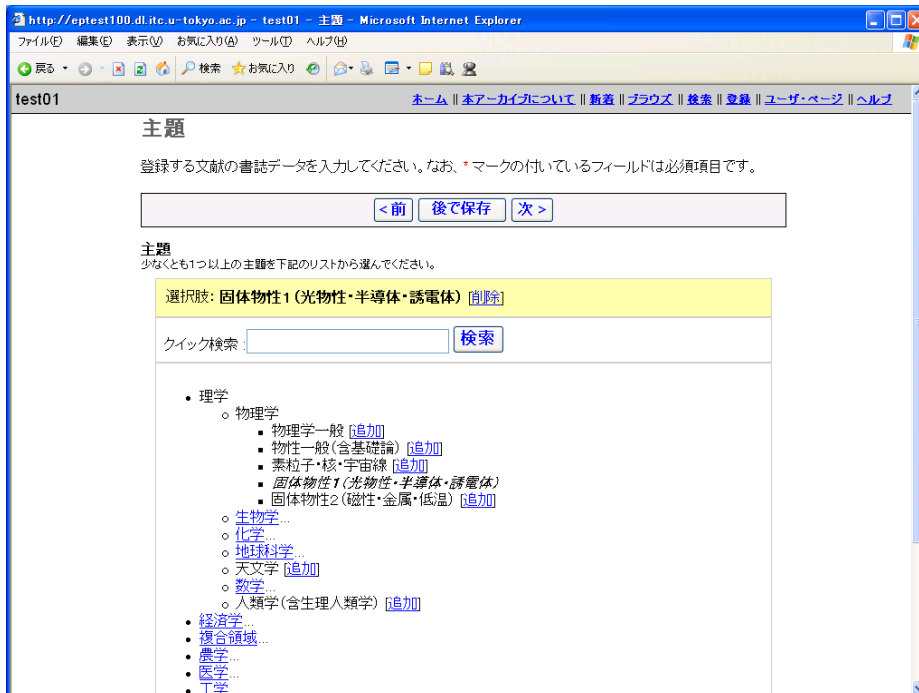


図 3-19 主題



⑦ その他の情報

その他，必要に応じて，情報を入力する（図 3-20）。

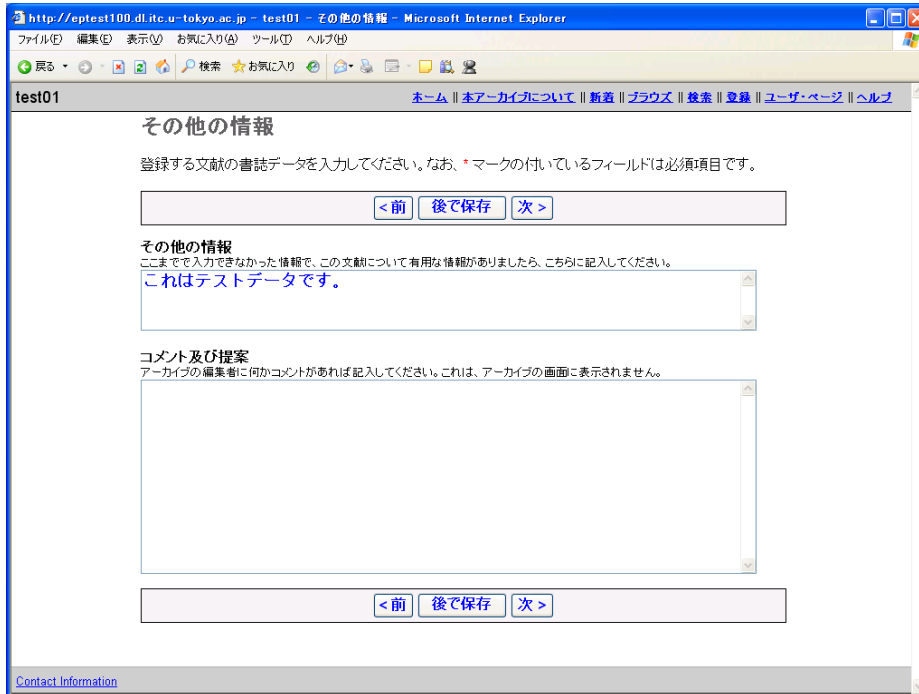


図 3-20 その他の情報

⑧ ドキュメント・ファイルの登録

文献を実際に登録（アップロード）するには，大きく4つの手続きが必要となる。まずは，「新たなドキュメント・ファイルを追加する」ボタンをクリックして，次の画面に進む（図 3-21）。

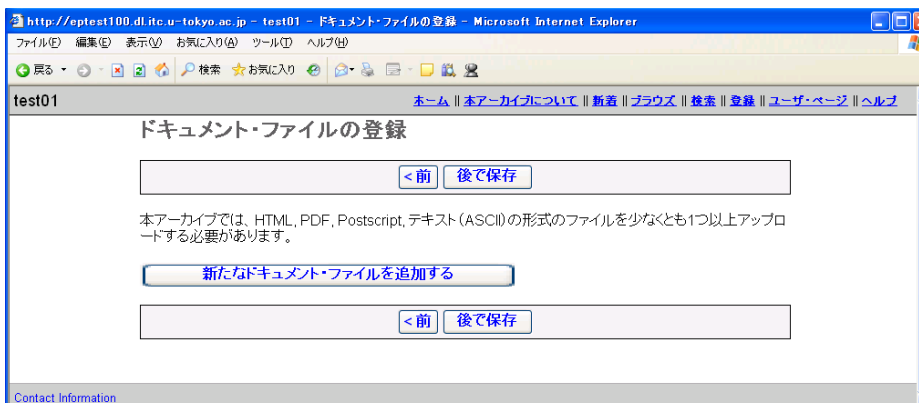


図 3-21 ドキュメント・ファイルの登録

⑨ ファイル形式の指定

ドキュメント・ファイルの保存形式を指定する。また、文献の閲覧を許可する範囲を 3 レベルから指定する (図 3-22)。

なお、EPrints では、HTML, PDF, Postscript, テキスト (ASCII) の形式のファイルを少なくとも 1 つ以上アップロードする必要がある。

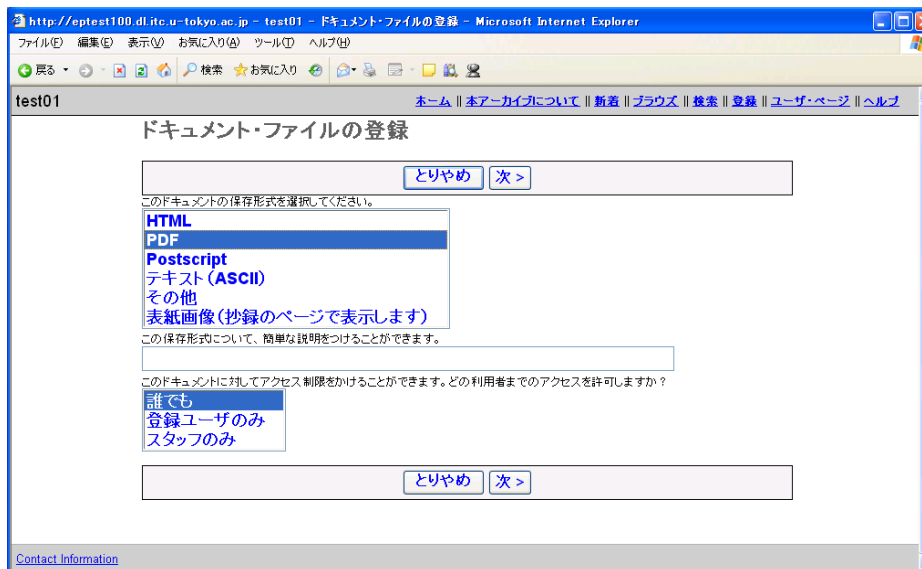


図 3-22 ファイル形式の指定

⑩ ファイルの登録

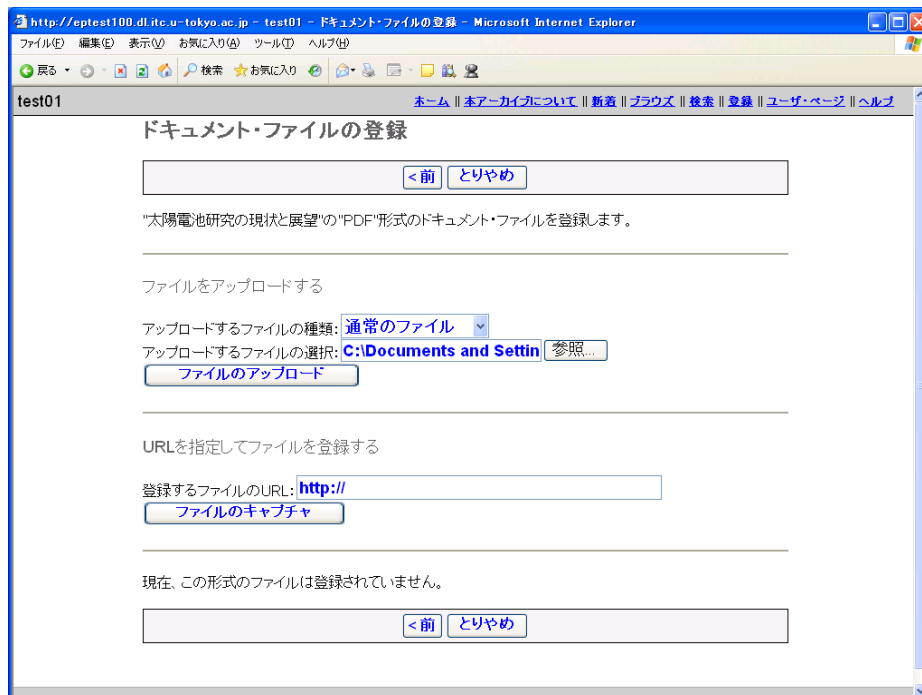


図 3-23 ファイルの登録

図 3-23 が実際に文献を選択する画面である。登録する文献のファイルが手元のパソコンなどにある場合には、「ファイルをアップロードする」を使う。「参照」ボタンをクリックして、そのファイルが保存されているフォルダ（ディレクトリ）に移動し、該当ファイルを選択する。ファイルを選択したら、「ファイルのアップロード」ボタンをクリックする。

ファイルが html 文書で、すでにインターネット上に公開されている場合には、その URL を指定して、投稿することもできる。その場合には、「URL を指定してファイルを登録する」を使う。URL を入力して、「ファイルのキャプチャ」ボタンをクリックする。

また、ファイルは複数投稿することができる。すべてのファイルを投稿したら、「完了」ボタンをクリックして、先に進む。

#### ⑪ 登録ファイルの確認

すべての手続きが完了すると、⑧の画面に戻る（図 3-24）。ここで登録されている内容を確認の上、先に進む。他に、追加するファイルがある場合は、「新たなドキュメント・ファイルを追加する」をクリックし、ここまでの手順を繰り返す。

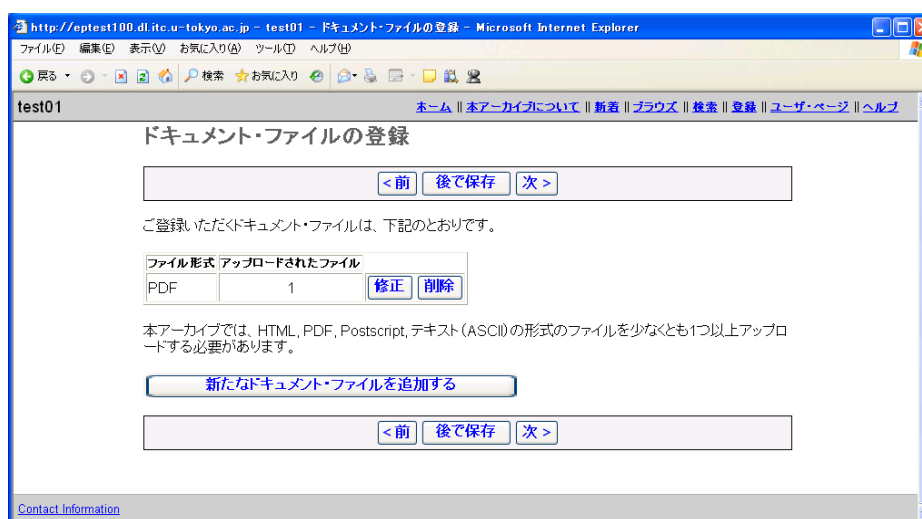


図 3-24 登録ファイルの確認

#### ⑫ 登録内容の確認

ここまでのステップで登録した情報の内容を確認する（図 3-25）。特に問題がなければ、ページ下部にある「登録承諾文」を確認し、「今すぐ EPrints に登録」ボタンをクリックする。

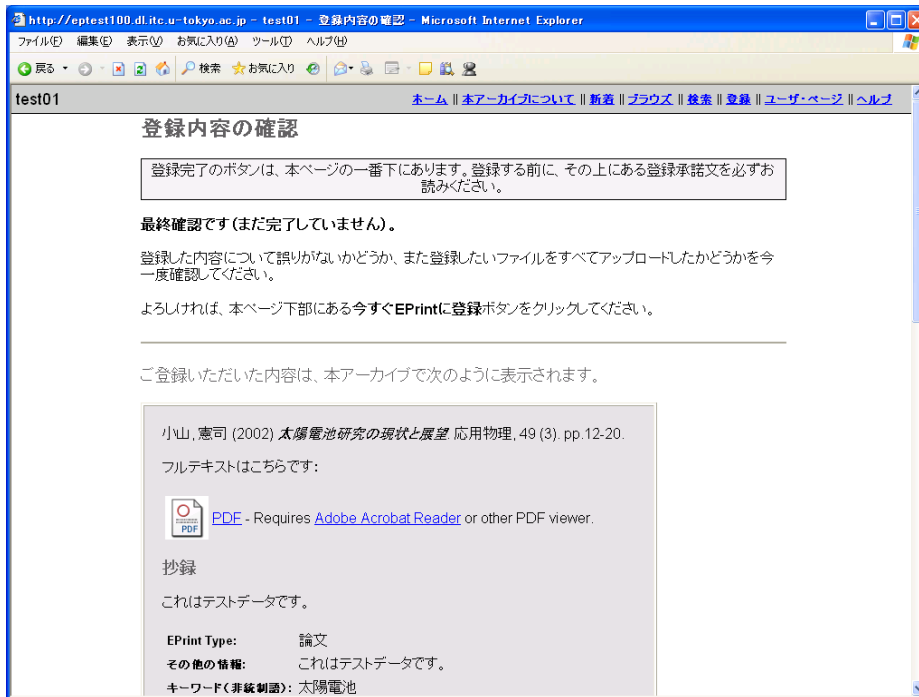


図 3-25 登録内容の確認

⑬ 登録完了

登録完了画面が表示され、すべての手続きが完了する (図 3-26)。

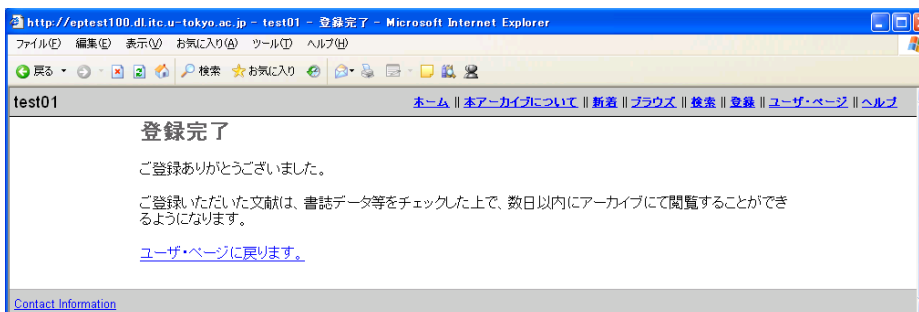


図 3-26 登録完了

## 第2部 運用上の諸課題

## 第 1 章 学内合意形成と運用体制

一般的に、試行運用期においては、ある程度までは担当部門の裁量で、システム構築や登録試行、ハーベスト等の実験を様々に身軽に実行することが出来る。図書館の自主事業の一環として、学内調整や制度・組織の整備を保留したままプロジェクトを進めても大きな問題はないだろう。

しかし、「〇〇大学機関リポジトリ」という名のもと、正式に学術機関リポジトリを立ち上げ、学内外に周知し、維持するには、機関そのもの（例えば大学当局）により承認された事業であるという保証が不可欠である。

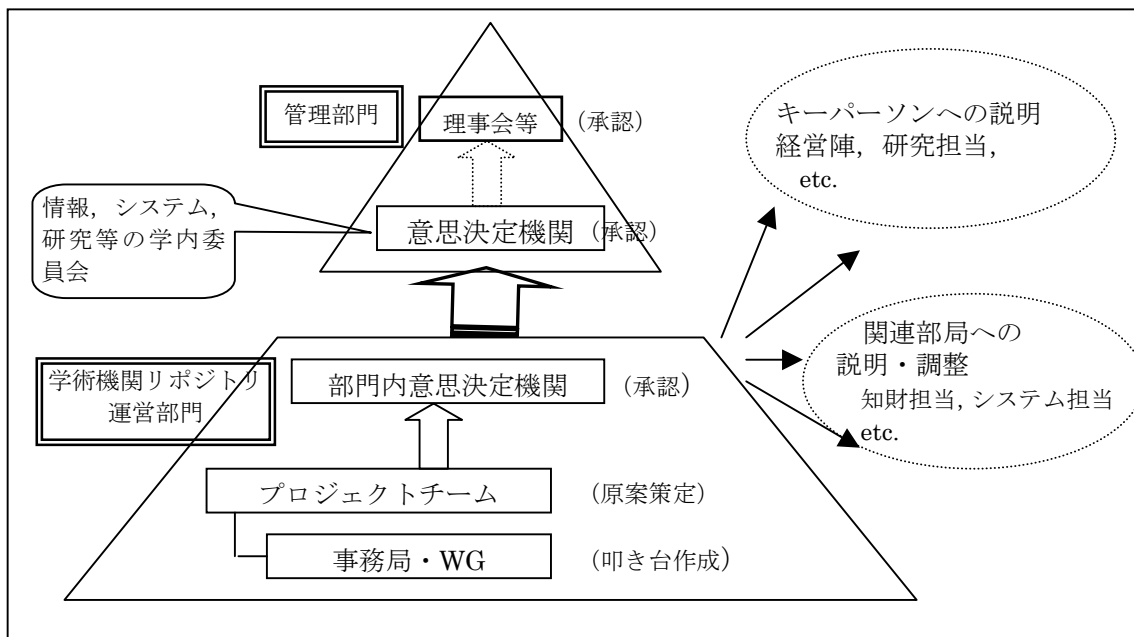
いいかえると、本番運用開始を目指すのであれば、「図書館の自主事業」を脱し、機関公認の 1 事業としての地位を獲得せねばならない。そのためには、少なくとも、大学自体への働きかけにより事業実施の合意を得、ゴーサインを取り付けること、運用のためのルールを整備すること、運用のための組織を整備すること、の 3 点をクリアすることにより、事業としての体を為す必要があるだろう。

### 1.1 学内合意形成

学内合意形成とは、機関自体がこの事業を認知し、賛同し、後押しするような状況を作り出すことである。学内要所への説明やプレゼンテーション、しかるべき委員会での審議を通じ、運営当事者以外に学術機関リポジトリを理解してもらうための努力、と言える。

#### 1.1.1 合意形成のプロセス

機関の風土や文化、または規模にも応じて合意形成のプロセスは多様であり、それぞれに最適な方法で合意形成を得ることが望ましい。



例えば、一般的なボトムアップ式のモデルによれば、まず、自部門内（千葉大学の場合は附属図書館運営委員会）で事業実施をオーソライズし、関係他部局（知財本部等）との調整を行い、上位部門（情報企画委員会、理事会）への提案を行うという手順を踏むことになる。

### 1.1.2 合意を得る内容 — 「なぜ学術機関リポジトリなのか」 —

プロセスや掛ける時間は各機関の事情次第であるにしても、合意を得るべき内容は共通している。「なぜ学術機関リポジトリを立ち上げるか」ということであり、その了解がとれれば「事業実施へのゴーサイン」が出たことになる。

図書館等を除けば、大学や研究者間における学術機関リポジトリの認知度はまだまだ低いのが実情である。このため、学術機関リポジトリ導入を目指すにあたっては、学術機関リポジトリとは何か、何の役に立つのか、なぜ自機関に導入するのか、といった「学術機関リポジトリ構築の意義・背景」、つまり「なぜほかでもなく学術機関リポジトリなのか」を適切に説明し、意識を共有しておく必要がある。

学術機関リポジトリの持つ意義や歴史的背景については、一般的に図書館の分野では、学術誌の高騰による雑誌購読中止（いわゆる「シリアルズクライシス」）が研究成果の入手・発信を妨げてきたことに対して、研究者側（機関）主体の研究成果流通基盤を新たに構築することで一石を投じてみよう、というオープンアクセス（OA）運動の試みであると理解される。しかしながら、この歴史を説明するだけでは、おそらく「図書館寄り」という印象を払拭することはできないため、OA以外の文脈、例えば、自機関で生産されたデジタル情報資源の蓄積・保存を行う、といった意義も同時に見出だしておく必要があるだろう。

さらに例えば、各出版社におけるセルフアーカイブ許諾率の上昇や英米議会におけるセルフアーカイブをめぐる近年の動向も、プラスの要素として盛り込んでおくべきである。

### 1.1.3 合意形成の補強材料 —メリットの強調，試行運用の実績提示—

理念的には賛同できるとしても、例えば、登録者の負担増，著作権に対する不安，先行例不足，効果測定のにくさ，といった不安が合意の足かせとなる可能性がある。

そこで、学術機関リポジトリを通じ、大学はその研究教育活動を社会に説明することができ、研究者は、自身の研究業績の可視性を高め研究インパクトの向上につながる、といった大学や研究者がいかに恩恵をこうむるか、という文脈によるメリットの強調こそが、合意を得る際の補強材料として必要不可欠である。

また、プロトタイプシステムなど学術機関リポジトリのシステムを実感できるものがあれば、説得材料として積極的にプレゼンテーションすることが望まれる。試行運用システムの提示によって検索や登録，ブラウジングの具体的なイメージをつかむことができるし、初期構築用のコンテンツが多少なりとも入っていれば、「既にここまで蓄積した」ことを実績として示すことができる。さらに、先行事例調査，教官向けアンケート，著作権調査等の実績があれば、適宜説明に加えることにより、補強材料となりうる。

	内容	補強材料
学術機関リポジトリ構築の意義・背景	<p>【共通する内容】</p> <ul style="list-style-type: none"> <li>学術機関リポジトリとは何か</li> <li>何の役に立つのか</li> <li>なぜ学術機関リポジトリである必要があるか</li> <li>どんなメリットがあるのか，</li> </ul> <p>【各機関固有の内容】</p> <ul style="list-style-type: none"> <li>なぜ本学に導入するのか</li> <li>本学にとってのメリットは何か</li> <li>本学研究者にとってのメリットは何か</li> <li>学外（社会，企業等）にとってのメリットは何か，</li> </ul>	<p>【メリットの強調】</p> <p>（大学）</p> <ul style="list-style-type: none"> <li>社会に対する研究教育活動内容の説明責任の履行</li> <li>中期計画・中期目標の具体的方策研究業績集約の事務合理化</li> </ul> <p>（研究者）</p> <ul style="list-style-type: none"> <li>論文の可視性向上</li> <li>研究インパクト向上</li> </ul> <p>（学外）</p> <ul style="list-style-type: none"> <li>産学連携のヒントとなり得る</li> <li>表面的な PR ではなく，具体的な研究教育活動を知ることができる</li> </ul> <p>【試行運用等の実績提示】</p> <ul style="list-style-type: none"> <li>登録/検索インターフェースのイメージ</li> <li>雑誌掲載論文，紀要論文の先行登録</li> <li>教官向けのアンケート実施結果</li> </ul>

表1 学内合意形成を得るべき内容

## 1.2 運用ルールの整備 —運用指針—

1.1 に述べた学内合意形成においては事業実施の了解を得ることに主眼を置いたが、運用の原則となるルール—運用指針—についても、事業開始前に策定し、承認を得るなり、機



関の公的な規定や裁定として位置づけるなり、各機関の必要に応じてオーソライズしておく必要がある。

下記は運用方針を Web で公開している先行事例の一部である。運用主体の部局はどこか、検討組織はどこに設けるのか、誰が何をどのように登録できるのか、一旦登録したデータの削除は可能なのか、著作権の扱いはどうするのか等、事業を切り盛りする上で根幹となる原則はおおよそ共通して含まれている。さらに、「意義や目的」、参加申請、著作権方針及び利用許諾書、などを含むケースもある。

なお、国内外とも、学術機関リポジトリへの登録は登録者側の希望に基づくケースが一般的であるが、クイーンズランド工科大学は、研究成果の可視性及びインパクト向上のため、研究成果の学術機関リポジトリ登録を原則として義務化するという、非常に先進的な制度を持っている。

表2 国内外の学術機関リポジトリの運用指針例

学術機関リポジトリ	運用部門	登録者	コンテンツ登録対象	コンテンツ削除方針	著作権規定
DSpace@MIT	図書館は全体の統括。加盟した研究組織は指針策定などの裁量が与えられる	DSpace に加盟した研究組織所属の研究者（要参加申請）。	研究教育利用目的で作成したもの	申請により可能だが、メタデータは残す	含む（原則として著者が全権利を保有し、学術機関リポジトリでの非排他的利用を認める）
	「Policies and Guidelines」に Community & Collection Policies(運用体制), Content Guideline (登録対象), Deposit License(著作権), 等の6方針が示される URL <a href="http://libraries.mit.edu/dspace-mit/mit/policies/">http://libraries.mit.edu/dspace-mit/mit/policies/</a>				
eScholarship (UCLA)	図書館は全体の統括。Unit単位に加盟する	加盟組織が登録可と判断すれば学内外の者を問わず可（要参加申請）	特に規定なし	Unit に削除申請すれば可能だがメタデータは残る。改版可	含む（原則として著者が全権利を保有し、学術機関リポジトリでの非排他的利用を認める）
	「Repository Policies」に登録対象、削除方針、登録申請書へのリンク、著作権方針へのリンク、等の基本方針が簡潔に1枚のWeb ページに収まっている。 URL <a href="http://repositories.cdlib.org/escholarship/policies.html">http://repositories.cdlib.org/escholarship/policies.html</a>				
CODA (カリフォルニア工科大)  LibrarySystem digital Collections の例	図書館	図書館関係者（アーカイブ毎に異なる）	特に規定なし	改版以外の削除は認めない	含む（原則として著者が全権利を保有し、学術機関リポジトリでの非排他的利用を認める）
	Caltech の学術機関リポジトリ登録コンテンツの一つとして公開している。 URL <a href="http://caltechlib.library.caltech.edu/19/">http://caltechlib.library.caltech.edu/19/</a> URL <a href="http://caltechlib.library.caltech.edu/6/">http://caltechlib.library.caltech.edu/6/</a>				

QUT ePrints (クイーンズ ランド工科大)	図書館	所属する者。 研究成果の学 術機関リポジ トリ登録は原 則として義務 である。	プレプリン ト、ポストプ リント、博士 論文、その他 の研究成果	規定なし?	QUT COPYRIGHT GUIDE に従う こと
	<a href="http://www.qut.edu.au/admin/mopp/F/F_01_03.html">http://www.qut.edu.au/admin/mopp/F/F_01_03.html</a>				
筑波大学 電子図書館 システム	図書館	所属する者	学内の研究成 果/所蔵資料/ 出版物等	申請により可能	含む（原則と して著者が全 権利を保有 し、学術機 関リポジトリ での非排他的 利用を認める）
	「電子図書館システムへの登録に関する実施要綱」に目的、定義、登録対象、提供されたコンテンツの利用方法、登録申請方法、削除・修正方針、運用組織等が盛り込まれている。 URL <a href="http://www.tulips.tsukuba.ac.jp/pub/riyou_annai/DL/youkou.html">http://www.tulips.tsukuba.ac.jp/pub/riyou_annai/DL/youkou.html</a>				
千葉大学 学術成果 リポジトリ	図書館	在籍経験のあ る教職員及び 院生	学内で主要部 分が作成され た学術研究成 果	申請により可能	含む（原則と して著者が全 権利を保有 し、学術機 関リポジトリ での非排他的 利用を認める）
	「千葉大学学術成果リポジトリ運用指針」 目的、定義、登録対象、提供されたコンテンツの利用方法、登録申請方法、削除・修正方針、運用組織等が盛り込まれた運用方針と登録申請書とがセットになっている。 <a href="http://mitizane.ll.chiba-u.jp/curator/about/">http://mitizane.ll.chiba-u.jp/curator/about/</a>				

### 1.3 運用体制の整備

学術機関リポジトリを運用開始するまでは、制度やシステムといった枠組み作りが運用部門の主要業務となるが、運用開始後は、コンテンツ登録の支援、著作権調査、登録申請の受付、学部・教官への説明といった日常業務が降りかかってくる。また、同時に、制度やシステムの維持管理、学内外への広報活動もその業務の範疇となる。

研究者自身によりセルフアーカイブされるようになれば、あまり運用部門の手はかからなくなると予想されるが、学内に浸透するまでの間は、各先行機関が経験しているように、コンテンツの拡充のために運用部門自身による手取り足取りの支援が必要となるであろうし、参加登録申請依頼や雑誌掲載論文の登録代行をキャンペーン的に行う場合などは一時的に業務量が集中することが想定される。

また、全学の研究者を1つの学術機関リポジトリの対象ユーザとするのではなく、学部、キャンパス、分野等を基準に学術機関リポジトリを「小分け」した単位を設け、それぞれの管理を各組織に任せる、という方式も想定される。例えば、MITのDSpaceやeScholarship

の場合は、学部や研究科をコンテンツ登録の単位（コミュニティ、ユニット）として各単位内の運営を任せており、学術機関リポジトリ運用主体である図書館は、統括的な役割に徹しているようである。

体制の組み方は各機関の事情に合わせる必要があるが、学内合意形成を行う時期に、あるいは、試行運用期に組んだ人材と体制をベースに実運用体制にスライドさせるのが効率的であろうと思われる。いずれにしても、コンテンツ登録の支援、著作権調査、登録申請の受付、学部・教官への説明、システム維持管理といった定期/不定期の実務（未経験の実務）をこなすグループが、大きな迷いをもたずに動けるような体制が望ましい。

## 第2章 コンテンツ収集・投稿促進

### 2.1 必要性

実際に動くシステムを構築し、諸制度を一通り整え、当座の事務処理体制を決めてしまえば、とりあえず学術機関リポジトリを運用開始することは可能である。

しかし、その学術機関リポジトリが充実したものになるかどうかは、別の問題である。

コンテンツが集まらない、あるいは、集まったにしても意図とずれたコンテンツで満たされているのであれば、せっかく立派なシステムや運用指針を持っていたとしても、学術機関リポジトリとしての価値は非常に低いものになる。学術研究活動のアウトプットの中核が何処に表れるかは、学問分野によって異なると考えられる。理工学のある分野では最良の論文は欧文商業誌に投稿されるのが通例であるかもしれないし、また、文系のある分野にとっては主たる研究成果発表の場は大学の刊行する研究紀要であるかもしれない。学術機関リポジトリはこうした学術研究成果の中心を見極め、それを的確にすくいとることによって、量的にもまた質的にも充実を目指していく必要がある。逆に非常に少ないコンテンツしか集まらない、あるいは、学術的価値という観点で焦点のずれた収集に留まれば、学術機関リポジトリの価値は大きく損なわれてしまう。

スティーブン・ハーナットはこのような「学術機関リポジトリもどき」を ESpace (DSpace をもじった E(empty) Space) と揶揄したが、いかにして ESpace を避けつつ、価値あるコンテンツで満たす道をたどるかが、学術機関リポジトリの成否を分ける最も重要かつ困難な課題である。

本章では、この観点に立ち、グラスゴー大学と千葉大学の事例を主に参考にしつつ、コンテンツ収集に際して取るべき2つの戦略を紹介する。

### 2.2 登録コンテンツ数の現状

英国のレポート (Pathfinder Research on Web-based Repositories 2004 .1) によると、45 機関の学術機関リポジトリにおけるコンテンツ登録数の中央値は 290 と報告されている。MIT などの一部の突出した例外を除けば、この程度の登録数で伸び悩んでいる (満足している?) のが世界の現状であり、量的な面に限っていえば、日本と彼我の差は実は殆ど無いも同然ともいえる。というのも、学術機関リポジトリを構築しようとする機関は、少なくとも、以下に示す方法で初期データを構築するだけで、おそらく数量的には数百点のコンテンツをそれ程の困難なく登録できるからである (もっとも問題はそれ以降であるが)。

## 2.3 2つの構築戦略

グラスゴー大学の報告 (<http://www.nii.ac.jp/metadata/irp/mackie/>) によれば、学術機関リポジトリをコンテンツで満たす戦略には大きく分けて2つある。

ひとつは、「リポジトリを根付かせるためにかせるために既存のコンテンツをいかに収集するか」という戦略であり、主として学術機関リポジトリ初期段階の量的充実に力点をおいたものである。

もうひとつは、ともかく上記の戦略によって学術機関リポジトリが機関内に認知された後に、「教員が体系的にセルフアーカイブ」(同) するよう仕向けるという、学術機関リポジトリの理想とする姿に近づけるための戦略である。

つまり、前者は、第1フェーズとしての初期データ構築であり、後者は第2フェーズとしてのセルフアーカイブ推進である、と位置づけることができる。

### 2.3.1 第1フェーズ 初期データ構築

同報告によれば、「リポジトリ構築の初期段階にあつては、学術機関リポジトリがどのように役立つのかを関係者に立証できることが重要であり、それはコンテンツがあつてはじめて成り立つものである」とされている。コンテンツがゼロの状態から学術機関リポジトリをスタートすることは無謀であるし、従来型の電子図書館とは違い、初めからコンテンツが充満している必要はない。しかし、学術機関リポジトリの存在自体を知ってもらい、将来のセルフアーカイブにつなげるには「ある程度の見せコンテンツ」を初期データとしてあらかじめ仕込んでおき、構成員の目に触れさせ、インパクトを与える必要がある。また、学術機関リポジトリに何かがある程度入っていれば、研究者への説得、学内外広報、当局への説明、等の場面で具体的な話ができるので、相手の理解を得られやすいというメリットがある。

では、具体的にはどのような既存コンテンツが学術機関リポジトリの初期データとしてふさわしいか。国内の大学で構築する場合を例に列挙してみたい。

対象	具体的に	千葉大学の例
学内外の Web サイトで既に公開されているコンテンツ	学部・研究室・個人の Web サイトで公開しているもの その他	学内の各 Web サイトを丹念に調査し、学術機関リポジトリ登録候補となる論文等を選定。本人の承諾を得て登録 (約 100 件)
電子化済み学内出版物 (紀要等)	学部等で電子化したものの。NII の学術誌公開支援事業で電子化したもの。	NII で電子化した紀要の複製を学術機関リポジトリに登録 (約 500 件)

電子化済み学位論文	学部等で電子化したもの	自然科学研究科の博士論文の学術機関リポジトリ登録を推進（平成16年度は約30件登録予定）
学術機関リポジトリ登録が可能な雑誌に掲載された論文等	ProjectROMEO で Green（セルフアーカイブ可）とされる雑誌の掲載記事を学術機関リポジトリ登録。グラスゴー大によれば「最難関」。	Green 誌の掲載論文について、期間・出版社を限定した形で学術機関リポジトリ登録を推進（平成16年度は約70件登録予定）

これらのコンテンツ登録支援（登録代行）は、基本的に図書館、学部、その他学術機関リポジトリ運営主体側が行うことになる。

リポジトリへの登録だけを行うのであれば、上記の作業に伴う労力はさほどのものでもない。しかし、Green 誌収録論文の登録を手堅く行うのであれば、登録候補の選定、著作権調査、著作者への許諾依頼・コンテンツ提供依頼を合わせて行う必要があり、相応の体制で行わねば、まともな結果は得られないだろう。

グラスゴー大学の報告においても、この Green 誌掲載論文の登録は「教員にコンテンツを提供するよう説得するという点で、さらに、出版社の著作権契約との関連で最も難関」とされる。しかし、研究者との接点を作り第2フェーズへと駒を進めやすくするため、また、雑誌掲載論文（特に査読済のもの）を主要コンテンツとすることにより、質的な充実を図るため、この「難関」は経験しておくことは決して無駄ではない。

### 2.3.2 第2フェーズ セルフアーカイブの促進

機関内の研究者にもしオープンアクセスの理念から始まって学術機関リポジトリの意義や目的を説明すれば、明らかに異を唱える向きは少数派であり、大方からは賛同を得ることができるであろうと予測される（千葉大、北大、グラスゴー大）。しかしながら、残念ながら、このことと、自らの手間を削ってセルフアーカイブすることが受容されることとは、別であり、学術機関リポジトリの意義や目的は、賛同は得られたとしても、面倒なコンテンツ登録作業のインセンティブとはなりえない。

## 2.4 セルフアーカイブの障壁

学術機関リポジトリの理念や意義が、それだけでは、研究者のセルフアーカイブのインセンティブとなりえない背景には、例えば次のような意識が障壁になっていると予測される。

- 1) 登録するメリットが感じられない。
- 2) 登録作業は時間と手間がかかる。やっつけられない。
- 3) 著作権に関する懸念。自分が登録することで何の問題も発生しないか。

これらの障壁について、学術機関リポジトリ運営主体は、以下のような論理と方法で払拭するよう努力せねばならない。

	消極的態度	対応	具体的に
1	登録するメリットが感じられない	メリットを強調する。あるいは、強制的に登録させる。	【メリット】可視性向上，OA 誌の被引用率の高さ，恒久保存，研究成果リストの出力（データを1回入力するだけで，複数の出力で使用）などを紹介 【強制】研究者の義務と大学が定める（例 クイーンズランド大）。公的資金補助を得た研究成果は必ず登録，など。
2	登録作業は時間と手間がかかる。	登録手順の簡易化・省力化に努める。学術機関リポジトリ運営主体による登録支援作業の実施。	登録インターフェースの入力項目や画面展開の改善。 （例）千葉大では，研究者向けの登録インターフェースを大幅に簡素化。図書館がデータ補完。 学術機関リポジトリ運営主体による登録支援作業の実施。 （例）セントアンドリュース大では，メール添付された論文を図書館が登録代行。
3	著作権に関する懸念。自分が登録することで何の問題も発生しないか。	各学会・出版社のセルフアーカイブ許諾方針の紹介。	ProjectROMEO による欧米学会・出版社のセルフアーカイブ許諾方針を紹介。国内学会の動向の紹介。

## 2.5 協力者の獲得

学術機関リポジトリに興味を持ってくれた研究者，批判的な意見を浴びせる研究者，いずれも関心を持ってくれているという点で非常に重要な協力者でありカスタマーである。

例えば，この人たちは，「学術機関リポジトリの伝道者として」，「システム評価等の対象者として」，「各種委員会のメンバー候補として」，「その他有益な助言者・忠告者として」，学術機関リポジトリ運営主体には無くてはならない人々である。

すでに目ぼしい研究者がいれば，味方になってくれるように直接頼めばよいが，一体どの研究者が協力者たりうるのか分からない場合，例えば，次のような機会を通じて協力者候補を発見することが出来る。

- 学内の Web サイトから多数の論文を公開している研究者

- コンテンツ提供依頼を行った際、積極的な問い合わせのあった研究者候補
- 学術機関リポジトリに関するアンケートを行った際、積極的な意見を貰った研究者

つまり、研究者からのメールや Web サイトの様子から「協力者の匂い」を感じ取ること、協力者の「匂い」がした研究者には、候補として必要なときに何かを頼めるように日頃からコンタクトしておくことが求められる。



(参考)

## 千葉大学における初期データ構築の試み

平成 17 年 1 月から 2 月にかけて、雑誌掲載論文及び自然科学研究科博士論文について、下記のとおりリポジトリ登録依頼を行った。

### 1. 雑誌掲載論文

#### (1) 概要

査読後著者最終稿のリポジトリ登録を許諾する出版社の刊行誌に千葉大学の教員が執筆した論文をピックアップし、各教員宛に論文本体の電子ファイルを提供するよう依頼した。

対象出版社：AMS (アメリカ数学会), APS (アメリカ物理学会), Blackwell, Elsevier, Nature Publishing Group

対象期間： 2002 年~2004 年刊行号に掲載されたもの

#### (2) 結果

依頼数 (教員数)	447 論文 (256 名)
論文電子ファイル提供数	141 論文 (31.5%)
登録可能論文電子ファイル数	65 論文 (14.5%)

※登録不可の論文電子ファイル (76 論文) は、そのほとんどが「出版社版 (Publisher's Version) であった。

(回答の詳細)

学会・出版者	依頼数	論文ファイル受領		ファイル送付無し (未回答あるいはファイル無し)
		登録可能	登録不可能	
AMS	91	17	17	57
APS	21	7	4	10
Blackwell	32	1	2	29
Elsevier	262	30	53	179
Nature online	41	10	0	31
計	447(256 名分)	65(39 名) 14.5%	76(45 名) 17.0%	306(177 名) 68.5%

平成 17 年 3 月 7 日現在

## 2. 自然科学研究科博士論文

### (1) 概要

自然科学研究科の平成 16 年度秋期修了者を対象に、学位論文本体の電子ファイルによる提供を依頼した。

### (2) 結果

論文提出者数	82名（課程 61名+論文 21名）
電子ファイル提供者数	30名
電子ファイル提供率	36.7%

## 第3章 著作権処理

### 3.1 はじめに

学術誌に掲載された論文の著作権（財産権的な意味での著作権）は、その全部あるいは大半は、学会や出版社に帰属するのが通例であり、著者は、自著論文のコピーや送信を自分の一存で行うことは出来ず、これら出版者や学会の許諾を得てはじめて可能となる。

言語著作物（書き物）全般の傾向かもしれないが、学術情報においては、このように「創り手」や「読み手」よりも「売り手」優位という構図が確立しており、情報流通の硬直化やその高コストな体質を生み出す要因と言えるかもしれない。

学術機関リポジトリという取り組みの意味付けの仕方にはいくつかあるが、著者の所属機関がこのような学術研究の根幹に関わる事態を憂慮して立ち上がり、自機関の研究成果を積極的に無料公開することにより、コミュニケーションの活性化を図り、「創り手」であり「読み手」である研究者の側に流通の主導権を取り戻そうという試み、すなわちオープン・アクセス運動の1形態として理解されることが多い。

著者側が主導権を握るには、より多く・より強く自著論文の著作権を保持せねばならない。そのためには、これまでに失ってきた（出版者・学会に明け渡してきた）著作権を少しでも多く自分のそばに置かねばならない。学術機関リポジトリとは、つまり、研究インパクトの向上という目標と同時に、著者の権利を取り戻す活動という側面があり、著者には、現在自著論文の著作権について、何を（どの支分権を）、誰が、どのように、どういう条件で保持しているかに関心を払うことが求められる。それは、実際にコンテンツを預かり、発信する者（学術機関リポジトリを運営する図書館等）にとっても無視できないことである。

本章は、主に著作権処理について、学術機関リポジトリを運営する側の実務的な側面を中心に記述しているが、学術機関リポジトリにおける著作権は上述のように「著者に権利を取り戻す運動」である、という文脈を前提として取り扱う必要がある。

### 3.2 著作権処理の必要性

学術機関リポジトリの登録対象の殆どは学術論文等であり、「思想又は感情を創作的に表現したものであって、文芸、**学術**、美術又は音楽の範囲に属するもの」という著作権法でいう「著作物」に該当する。

学術機関リポジトリにおいては、「著作物」をサーバ上に複製したり、送信したり、不特定多数のエンドユーザがサーバからダウンロードしたり、といった場面が起こる。すなわち様々な方法による「著作物の利用」が行われる。

それが「適正な利用」であるためには、一部の例外（法 30-50 条に規定される権利制限

規定に該当する利用方法，著作権が消滅した著作物，著作権の無い著作物，等の場合）を除き，利用しようとする者が，当該著作物の著作権者から許諾を得る，あるいは，当該著作権の全部あるいは一部を著作権者から譲渡させ，目的遂行上必要な権利を得る，といった「著作権処理」と言われる手続きを踏む必要がある。

このことは，全文コンテンツを電子的に蓄積・提供するいわゆる「電子図書館」におけるのと同様である。

著作権処理をいかに合理的に済ませるかという点が学術機関リポジトリ運用者にとっての大きな課題のひとつである。

### 3.3 著作権処理の諸側面

#### 3.3.1 著作権処理の当事者

学術機関リポジトリにおいて著作権と関わりを持って登場する当事者は次の4者に単純化される。

（実際は，権利共有の場合などがあり，一筋縄にはいかない）

当事者	立場	内容
研究者	著作権者	原著作者である
出版社・学会等	著作権者	原著作者から権利の一部/全部の譲渡/許諾を受けている場合が多い
学術機関リポジトリ運営主体（図書館等）	利用者	複製，公衆送信を行う
エンドユーザ	利用者	閲覧，ダウンロードする

#### 3.3.2 当事者間の関係

図 3-1 は4者の関係を示したものである。

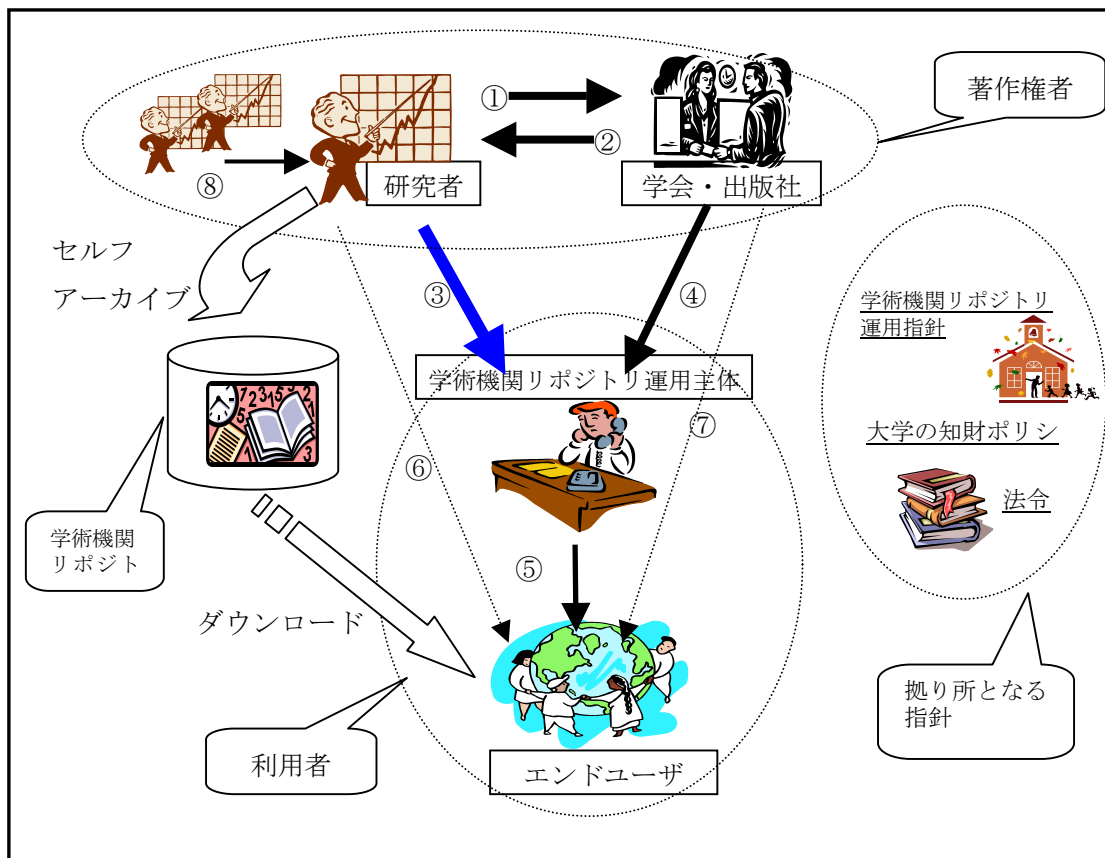


図 3-1 学術機関リポジトリにおける著作権処理の相関図

①～⑧に示す矢印は、利用権限の流れ（許諾／譲渡）を示す。学術機関リポジトリ運営主体が最も意識すべきは著作物利用可能性と直接関わる③と④である。

③は許諾書面を用意し、大学内で学術機関リポジトリ運営主体が中心となって処理を進めることとなり、次項でその書面の構成や事務処理方法について検討してみたい。

④については、少なくとも欧米の主要学会・出版社については英国の調査機関（ProjectROMEO）によりセルフアーカイブ許諾方針について調査が行われ、結果が公表されているため、それを参照すれば済む。しかし、国内の学会・出版社についての同様の本格的な調査は行われていない。各大学が個別に学会・出版社にアプローチすることは非常に非合理的なので、我が国の図書館界においても、英国同様の調査を実施し、出来れば著作権者同士の関係（①と②）もあわせて情報共有していくことが望ましい。

また、③と④の処理を行う上では、利用権限の流れとは直接関わらないが、著作物利用の仲介者として学術機関リポジトリ運営主体がエンドユーザに対して「説明」や「警告」を行う⑤も意識する必要がある。

さらに、エンドユーザが権利制限規定の範囲を越える利用（例えば翻訳、改変、放送、

有償提供等)を希望するのであれば、著作権者とエンドユーザとの間で関係(⑥⑦)が生じることとなる。この利用様態は、一般に学術機関リポジトリの範疇を越えるため、本稿では取り上げない。

### 3.4 研究者から学術機関リポジトリ運営主体への許諾

#### 3.4.1 許諾内容

図 3-1 の③で行う許諾内容に盛り込むべき要素とそこに記載される一般的な内容を列挙してみたい。国内外の学術機関リポジトリや電子図書館における著作権処理の先行例によれば、書面に盛り込まれる要素はおおむね共通しているが、当然、各大学の実状と学術機関リポジトリの目的等に合わせ適宜アレンジする必要がある。

- ・ 当事者  
誰が誰に対して許諾するのかを明示する。  
個々の研究者が許諾者であり、学術機関リポジトリ運営主体の責任者が被許諾者となる。運営形態次第では、学部長や学長が被許諾者となることもありえよう。
- ・ 利用対象著作物  
どの著作物を利用したいのかを明示する。  
個々の著作物単位に許諾書を作るのは煩雑なため、可能であれば「今後登録対象とするコンテンツ全て」などと包括的に示すのが望ましい。
- ・ 利用目的  
一般的には学術機関リポジトリで公開する意義(視認性の向上、情報発信の強化、情報開示の促進 等々)を述べることになる。
- ・ 利用内容  
一般的な学術機関リポジトリであれば電子ファイルを電子的に複製し、ネットワークを通じて不特定多数に送信可能化することは最低限盛り込む必要がある。  
=>保存のための複製(バックアップ)や将来の利用保証のために行う媒体変換等も含める等が想定される。
- ・ 利用期間  
更新制にするか無期限とするか。  
=>著作権処理の回数を出来るだけ少なくするのであれば、期限を設けず、システムが存続する限り恒久的に利用可能とした方がよいだろう。
- ・ 著作物使用料  
著作権者に対して著作物使用料は払わないこと、エンドユーザに対して著作物使用料を科さないことを明記する(著作権料を払う、という考え方もあるかもしれない)。
- ・ 利用者としての遵守事項  
目的外利用しない。  
2次利用者(エンドユーザ)に対する注意喚起(図 3-1 の⑤)。著作権者に許諾を要する利用(図 3-1 の⑥⑦)と要しない利用(権利制限規定内の利用)の周知徹底。
- ・ コンテンツ削除時の取扱  
学術機関リポジトリからコンテンツを削除した時点で、利用許諾は無効とする。

- ・ その他  
その他の事項は当事者間で別途定める。

### 3.4.2 実施方法

著作権処理を行うタイミングとしては、大きく分けて次の2パターンがある。

1. 初回のコンテンツ登録前に、1回だけ実施する。
2. コンテンツ登録する度に、一つ一つの著作物について毎回実施する。

特に紙の書面を用いるのであれば、1によるやり方が妥当である。例えば千葉大学の場合、学術機関リポジトリのアカウント発行申請書と著作物利用許諾書をかねた書面を初回に提出させることとしている。

ソフトウェアの利用許諾でよく採用されているPCの画面上同意書の文面を表示させ「同意する／しない」を選択させる、いわゆるクリックオンライセンス式であれば、2の方式も可能であろう。ただし、効力をもつ契約書として著作権者の署名（サインや押印）を重視するのであれば、1の方式が望ましいと考えられる。DSpaceでは、コンテンツ登録の最終段階で著作権方針を画面上に表示させ、Agreeさせる仕様となっている。

初回の登録までに書面での許諾を得ておき、毎回の登録時に画面上で著作権方針を表示して注意喚起する、という方式が磐石かもしれない。

### 3.4.3 発生する事務作業

書面による著作権処理を例にとれば事務作業としては、おおよそ以下の要素が想定される。

- ・ 文案策定
- ・ 発送準備（紙ないしメール）
- ・ 後処理（回答の整理、保管、アカウント発行、その他）
- ・ 質問対応

学術機関リポジトリの場合は、原則として著作権者が自機関内に限定されるため、「電子図書館」の著作権処理業務で最も難儀する「著作権者探し」は行う必要がない。

短期間のうちに大量の著作権処理をこなして結果を出さねばならない、という状況であれば、場合によっては外注等も視野に入れ人海戦術で乗り切る必要があるかもしれない。

## 3.5 課題

極論すれば、著作物1件それぞれに異質な、大小の様々な次元の課題が含まれており、それらの全てを手堅く解決しようとするれば、学術機関リポジトリ事業そのものが頓挫してしまうであろう。

しかしながら、これまでの先行事例の経験から、今後学術機関リポジトリ構築を本格化させるにあたり看過できない共通の課題がいくつか浮上している。

### 3.5.1 共著者の著作権処理

1 論文が共著の場合、特段の取り決めがなければ、共著者全員が著作権を保有している。全員が学内に在籍中であればまだ良いが、学外者を含む場合にまともに全員を相手にしようとするれば、著作権処理作業が停滞してしまう。

便法として、代表著者を許諾者とみなし、共著者の同意を取り付けた上で許諾するよう依頼することで、許諾者を1人にすることができる。

### 3.5.2 国内学会・出版社の著作権方針の把握

現時点では、国内誌投稿論文をセルフアーカイブする際に参考となる、ProjectROMEOに該当する国内学会・出版社に対する本格的な著作権方針調査は行われていない。このため、国内誌掲載論文のセルフアーカイブは当面は本格的実施に踏み切るのが困難な状況にある（個々の学術機関リポジトリ導入機関で、学会・出版社の著作権方針を確認する必要がある）。

### 3.5.3 著者最終版 Author Final version の入手

図 3-1 の④において90%を超える緑化誌の大半がセルフアーカイブ可と認めているのは”Author final version” と呼ばれる著者側にある最終原稿であり、Publisher version と呼ばれる、電子ジャーナルとして提供される「完成形」電子ファイルのセルフアーカイブを認めているのは少数の出版社に限られている。

研究者は、査読前後を問わず、Publisher version 以外の形で（つまり自作したファイルとして）、自分の論文を完成形の1電子ファイルで保有していることは少ないようである。例えば、著者側に存在するのは、本文及び参考文献等の複数のtxtファイル、10数本からなる図表等の画像ファイル、といったバラバラの状態である。場合によってはそれらさえ揃わず、「紙」が混ざる場合もある。

この問題に対処するには、セルフアーカイブのために、著者側でも、Publisher version とは別に論文の完成形を作成するよう依頼すること、あるいは、学術機関リポジトリ運用主体がその作成を代行すること、が想定されるが、そのような手間が積極的に受け入れられるとは考えにくい。また、割り切って、統合された1電子ファイルでなく、複数のファイルからなる場合には、そのまま全ファイルを登録する、という選択肢もあるかもしれない。



(参考)

平成 12 年度図書館電子化システム特別委員会第 3 年次報告: II-4.資料電子化に伴う具体的な著作権処理について

<http://wwwsoc.nii.ac.jp/janul/j/publications/reports/71/s4.html>

## 第4章 学内同種事業との連携

先に図書館の役割として、現代の学術情報流通において、学術機関リポジトリの構築と運営に図書館は主体的に取り組むべきである、と述べたが、学内全体の理解を得て、安定的、効果的な運営を行っていくためには、学術情報を創成・保持する研究者はもちろんのこと、以下のような学内の他のシステム、関係機関との協調と連携が必要となる。

- 教育研究成果情報を提供するシステムとの連携

著書、学術論文などの研究成果情報を登録したデータベースである研究業績データベース、また、講義計画や講義内容、課題や資料などの教育情報を登録したデータベースであるシラバスデータベースに代表されるように、全学的な教育研究成果の検証が可能なシステムが、全国の多くの大学で整備されインターネット上で公開されている。これらのシステムは、学内の研究者情報や教育研究活動の情報等を統合してデータベース化し、教育研究を支援するという目的で構築されている。そして、その情報をインターネットを介して提供することによって、研究者ばかりでなく、一般市民へも大学等の教育研究活動に関する信頼性の高い情報を幅広く提供し、開かれた大学としての役割を果たしている。また、国内だけではなく国際社会への理解の推進を図り、我が国の大学の国際的通用性の向上に寄与している。

こうしたシステムの多くは、事務局を始めとする大学の事務部門、及び、ネットワーク基盤部門において構築、運用されているが、それらの目的、運用方法と学術機関リポジトリのそれとを比較すると、いかに似通っている部分が多いかに気付く。このことから、安定的な学術機関リポジトリの運用のためには、これら先行するシステムで培われた経験と技術に学ぶべきであり、同一情報を二重に登録するといった手間を教員にかけさせない等、大学全体の情報発信に係る作業の効率化のために、密接的な相互連携を行うべきである。

- 教育サービスとの連携

ネットワーク基盤の整備・普及によってインターネットによる教育は現実的なものとなり、各大学ではインターネットを利用した大学講座の開講や、バーチャルユニバーシティの構築等様々な情報通信技術を活用した **e-learning** と呼ばれる教育サービスを提供し始めている。その良い例が無料で一般社会に自校の講義資料等を提供している MIT の「オープンコースウェア (OCW)」であるが、MIT ではこれらのデジタル資料を収録、公開する基盤のシステムとして学術機関リポジトリ構築ソフトである **DSpace** を使用している。

日常の教育から生まれる講義資料、実習資料、セミナー資料、各種マニュアル、学生レポート等の教育成果は、学術機関リポジトリにとって、学内、一般社会へのアピール度が高く、かつ大量に収録することが可能である魅力的なコンテンツである。MIT の例からも

明らかのように、学術機関リポジトリが e-learning に代表されるこれらのコンテンツを扱う既存の教育サービスと連携できれば、あるいは、その関係機関の協力が得られるならば、これらの教育成果を効率的に学術機関リポジトリへ収録することができ、その機能を利用して、大学はより効果的に教育成果を社会に還元する体制が整う。こうなれば、学術機関リポジトリは研究的側面だけではなく、教育的側面からの存在意義も増すことになる。

- 学内における小規模同種活動との連携

機関全体の資料を収集・保存・公開するという意味での学術機関リポジトリではないが、学内の構成員及び部分的な組織が主体となって、計画的に研究成果をインターネット上等で公開している大学の例は少なくない。

各大学の学部講座等研究組織のホームページでは、各組織内の学位論文、紀要、研究成果の解説などを一般社会に向けて公開している場合が多い。ただこの場合、学術機関リポジトリという概念は取っておらず、多くは電子化したファイルをインターネット上のコンテンツとして体裁を整え並べてあるだけであり、より広範囲の人に利用してもらうためのデータ交換、データ検索などの機能は備えていない。また、そのコンテンツの永続的な保存・アクセスを保証しているわけでもない。

一方、限られた学問の分野ではあるが、その分野の組織が主体となって、教員等が資料をセルフアーカイブ方式で登録・公開している例もみられる。東京工業大学の「Information Theory Archive of Japan 論文サーバー」は、同大学大学院理工学研究科集積システム専攻情報通信システム講座ネットワーク構成分野が構築・運営しているもので、「情報理論とその応用」分野の論文を収録対象とし、教員等により EPrints を用いて資料をセルフアーカイブ方式で登録している。また、北海道大学大学院理学研究科数学専攻の「数学の海」も数学分野の論文について同じような方式をとっている。このような例は、教員自身が資料を学術機関リポジトリに登録して保存・公開する必要性を自覚していることを示しており、学術機関リポジトリの理念と一致する。

これらの同種の活動において公開される資料は、正式に著作権処理を行っている、あるいは著者に同意を得ている場合が多く、学術機関リポジトリの構築・運用の初期段階において行われる初期データ整備の登録対象として注目に値する。

また、これらの活動組織は、他と比較すると学術機関リポジトリの構築・運用に対する教員の理解が得やすいと推測され、学内合意に向けての布石とするため、協力を要請する最初の候補としてリストアップすべきである。

- 知的財産管理部門との連携

近年、大学の使命として、教育研究を通じての長期的な社会貢献は勿論のこと、産業界を中心とした社会との日常的及び組織的連携により、大学の研究成果を最大限に活用し、直接的に社会に還元することが求められてきている。このニーズに応じるため、大学にお

いて新たに設けられたのが知的財産管理部門であるが、知的財産は特許的観点により捉えられ、どうしても利益を重視したものとなりがちである。その結果、財産として価値のあるものはすべて大学に囲い込まれ、学術情報の自由な流通が阻害されることになる。これは、学術機関リポジトリの **OpenAccess** 的な精神とは相反するものといえる。

しかし、その活動の目的を学内における知的財産の発掘・権利化・活用を迅速かつ組織的に行った上で社会に還元し、地域経済の活性化、我が国の国際競争力の向上に寄与することと定義するならば、学術機関リポジトリの理念とかなりの部分で一致する。

こうしたことから、学術機関リポジトリを構築・運用するに当たっては、早い段階で知的財産管理部門と協議を行い、運用方針等で互いの立場を明確にした上で、相互に牽制しあうことなく互いの本来の目的を達成しなければならない。

## 第5章 メタデータ

メタデータは、「データに関するデータ (data about data)」と定義される。情報資源 (コンテンツ, リソース) 本体に関する記述を行うためのデータであり, 情報資源本体を一次情報と捉えればそれに対する二次情報とも言える。学術機関リポジトリにおいては, 例えば, 収録された論文等の研究成果物についての「タイトル」, 「著者に関する情報」, 「キーワード」, 「作成日付」等がメタデータに相当することとなる。

学術機関リポジトリ構築に際し, メタデータ設計に関しては, 何を収録対象とするかを踏まえた上で, 各機関における学術機関リポジトリシステム中でのデータ管理, 公開・検索等の観点と同時に, メタデータ交換 (メタデータ・ハーベスティング) を通じた他機関との相互運用性 (Interoperability) にも配慮した必要項目の設定や入力規則の検討が望まれる。

### 5.1 各リポジトリにおけるメタデータ項目設定

#### 5.1.1 メタデータスキーマ

メタデータ項目設定にあたっては, 導入する学術機関リポジトリ用ソフトウェアがサポートするメタデータスキーマ (記述規則) が何であるかを, まず確認しておく必要がある。それをベースとして, 必要に応じカスタマイズを施すこととなる。

本報告書第1部で取り上げたオープンソースの DSpace, EPrints は, いずれもダブリン・コア (Dublin Core Metadata Element Set (DCMES)) をメタデータスキーマとして採用している。

ダブリン・コアは, Dublin Core Metadata Initiative (DCMI) により, ネットワーク情報資源の発見と相互運用性を主眼にメタデータの標準化を目的として開発されたメタデータ記述規則である。「Title」, 「Creator」, 「Subject」, 「Identifier」, 「Description」など15の基本項目 (Core Element) を定めており, これらに限定子 (qualifier) を付加することによる各項目の詳細化も認められている。現状では, ダブリン・コアが, ネットワーク情報資源用メタデータ記述規則の国際的な標準として各国で利用されており, これを共通規則としたメタデータの収集・統合検索のシステムも稼動している。

#### 5.1.2 項目仕様検討・カスタマイズ

次に, 各学術機関リポジトリソフトウェアが採用するメタデータスキーマをベースとして, 各機関の学術機関リポジトリ収録対象等に応じたカスタマイズをしていくこととなる。なお, 項目仕様検討においては, 詳細は次節に述べるが, メタデータ・ハーベスティング時のメタデータ出力を念頭においておくべきである。

学術機関リポジトリのデータ収録対象は, 学内で生産された研究成果物であるが, それ

には、論文を始めとして、各種報告書、会議・講演録、教材等様々なタイプが想定される。それらについて、何をメタデータとして入力し、どの項目に格納するか、また各項目をどのように記述するかは十分な検討をすべきであろう。

そのタイプ毎に例えば、各情報資源独自の管理番号、対応する図書館所蔵資料の請求記号・資料 ID 等や、あるいは、各学術機関リポジトリで詳細な検索を提供した場合に用意しておくべき種別等の項目など、各機関の目的に応じたメタデータスキーマの拡張・詳細化を行うことも想定されよう。このような点を考慮すると、メタデータ項目のカスタマイズの容易さや拡張性という点も、学術機関リポジトリソフトウェア選択時の一つのポイントとなるであろう。

なお、一方、メタデータがハーベスティングされ統合的データベースに収録された場合等での相互利用性の観点からすると、使用項目・記述方法等について何らかの共通ルールを決める枠組みも望まれる。具体例として、ダブリン・コアを採用している先行リポジトリを見てみても、論文の掲載誌情報（雑誌名・巻号・出版社等）の格納項目・記述方法等に、機関毎のばらつきが見られる状況である。

## 5.2 メタデータ・ハーベスティング

ここでは、メタデータを介した他機関とのシステム間リンクに関わる部分を述べる。

### 5.2.1 メタデータ・ハーベスティングと OAI-PMH について

学術機関リポジトリの狙いの 1 つには、研究成果物の可視性向上があるが、その効果を高める技術的手法としてメタデータ・ハーベスティングがある。その仕組みは、自機関リポジトリサーバに登録されたメタデータを機関外に流布させることで情報発見・アクセスを促進させるという考え方に基づいている。

OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting) は、メタデータ・ハーベスティングのための通信規約 (プロトコル) である。学術機関リポジトリソフトウェアといえば、本プロトコルに準拠していることが事実上の必須要件となっており、本報告書で取り上げた各学術機関リポジトリソフトウェアもすべて対応している。

OAI-PMH は、コンテンツを保有しそのメタデータを発信する機関 (データ・プロバイダ) と、メタデータを収集し情報サービスを運営する機関 (サービス・プロバイダ) との間のデータ交換の標準化を目指すものである。サービス・プロバイダは、ハーベスタと呼ばれるメタデータ収集ソフトウェアを用い、データ・プロバイダが運営する OAI-PMH 準拠のリポジトリ群からメタデータをハーベスト (harvest=刈り取り, 収集) し、自らの情報サービスを構築する。

なお、メタデータ・ハーベスティングは、統合・横断検索を実現させる技術の 1 つと位置付けられるが、例えば類似する、Z39.50 などを利用した横断検索システムと比較した場

合、検索実行時に分散する複数のデータベースを直接検索するのではなく、中央集約的な1つのデータベース中に予め収集・蓄積されたメタデータを検索に用いるという点で特徴がある。また、機関外のシステムとのデータ交換・収集を行うものであるが、これは通常メタデータのみを送受信するものであって、PDF等の研究成果物本体については、各学術機関リポジトリに格納されているデータが参照される。

### 5.2.2 メタデータ・ハーベスティングのための設定と注意点

OAI-PMHを利用したメタデータ交換では、限定子(qualifier)無しのダブリン・コア形式(Simple Dublin Core)の出力応答が最低要件となっている。Simple Dublin Coreに変換することをダム・ダウン(Dumb-Down)と呼ぶ。

学術機関リポジトリは、ハーベスタに応答し、要求された形式でメタデータ出力できるように(少なくともSimple Dublin Coreでデータを返せるように)設定しておくとともに、Dumb-Downして出力した時にも項目内容の整合性が保たれるよう配慮しておく必要がある。

また、学術機関リポジトリに収録しているメタデータをすべてハーベスティングさせるか否かについても考えておくべきである。研究成果物の種別や、論文の査読の有無、論文の掲載雑誌等で区別したいというニーズがあれば、ハーベスティングの可否を判断できる項目をメタデータ中に用意しておくことなども想定される。なお、OAI-PMHでは、ハーベスティングの可否判断はサポートされていないため、学術機関リポジトリ側の各ソフトウェアレベルでの対応が必要となる。

### 5.2.3 国立情報学研究所メタデータ・データベース(JuNii)との連携

国内の各大学・研究機関等のメタデータを蓄積し、統合検索できる環境を提供するサービスとして、国立情報学研究所の「JuNii 大学 Web 資源検索-大学メタデータポータル」がある。JuNiiは、OAI-PMHにおける、サービス・プロバイダ機能及びデータ・プロバイダ機能を備えている。

国内において構築される学術機関リポジトリでは、ハーベスティングによるメタデータ登録対象としては、まずはJuNiiが第一目標となると考えられる。JuNiiは、国内学術機関からのメタデータを集約し検索を提供することで、日本における研究成果物のポータルの役割を果たす。

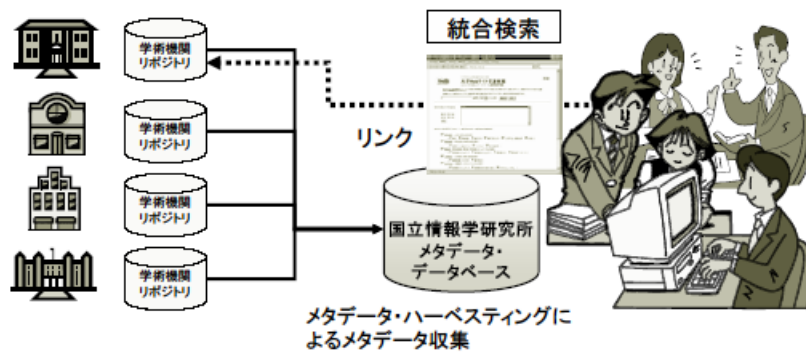


図 5-1 学術機関リポジトリと JuNii との関係

なお, JuNii は, ダブリン・コアに準拠した標準的なデータ形式を基本としているが, NII 独自の限定子を採用するなど拡張したものである。詳細は, 「NII メタデータ・データベース入力マニュアル 1.2 版」および「XML Schema for NII メタデータ・データベース」等で参照できるが, 例えば, Subject には NDC 付与, Type (資源タイプ) には NII 所定のタイプ名付与などを必須としている。国内での学術機関リポジトリソフト導入に当たっては, これらに対応したシステムが用意されることが望まれる。



No.	要素	入力レベル	備考
1	Title (タイトル)	必須及び選択	
2	Creator (作成者)	あれば必須	
3	Subject (主題)	必須及び選択	
4	Description (内容記述)	選択	
5	Publisher (公開者)	選択	
6	Contributor (寄与者)	選択	
7	Date (日付)	あれば必須	
8	Type (資源タイプ)	必須	
9	Format (フォーマット)	-	
10	Identifier (資源識別子)	必須及び選択	
11	Source (情報源)	-	当面使用しない
12	Language (言語)	必須	
13	Relation (関係)	選択	
14	Coverage (範囲)	-	当面使用しない
15	Rights (権利関係)	-	当面使用しない

次の項目は、メタデータの記述要素ではないが、記述を必須とする。

1	Institution (機関名)	あれば必須	
---	-------------------	-------	--

次の項目は、メタデータの記述要素ではないが、記述することが可能である。

1	Comment (コメント)	選択	
---	----------------	----	--

図 5-2 JuNii メタデータ項目

また、「NII メタデータ・データベース入力マニュアル」は、各項目の記述規則等を示しているが、学術機関リポジトリが収録対象とする個々の研究成果物レベルでの記述に関しては、それらのコンテンツの特性に即した、より詳細な応用プロファイルを考えていく余地がある。これについては、今後、学術機関リポジトリを構築する大学と国立情報学研究所でさらに議論を深める必要がある。

#### 5.2.4 その他のサービス・プロバイダへの対応

OAI-PMH に対応したリポジトリであれば、JuNii に限らず他のハーベスタからのメタデータ収集を受け付けることができる。メタデータの広汎な流布には、様々なサービス・プロバイダからハーベストしてもらうことが1つの方法とも言える。

OAI (Open Archive Initiative) では、データ・プロバイダおよびサービス・プロバイダの公式リストを維持しており、学術機関リポジトリをこのリストに登録する際は、OAI 準拠であるかのテストを行う機能を持っている。

大規模なサービス・プロバイダの例としては、ミシガン大学によるプロジェクトである OAIster がある。OAIster は、世界中のリポジトリから収集したメタデータをデータベースに蓄積し、検索インターフェースを提供している。2005 年 1 月現在、396 のリポジトリから収集された 480 万件以上のメタデータが登録されている。日本国内のリポジトリとしては、JuNii (57801 件)、千葉大学 (638 件) のメタデータが収録済みである。

OAI 等のデータ・プロバイダリストに、自学術機関リポジトリの情報を登録すれば、そのメタデータの流通の可能性は一層高まることとなる。

なお、ここで注意したいのは、一度プロバイダリストに登録されると、不特定のハーベスタからのハーベスティングを受け付けることになる点である。JuNii や千葉大学では、リスト登録後数日中に、ハーベスティングが開始された。

### 5.3 メタデータの入力・品質管理

千葉大学附属図書館をはじめとした先行事例を見ても、研究成果を学術機関リポジトリに登録する研究者に対して、詳細で正確なメタデータ入力を求めるのは困難と言える。このため、研究者がデータ登録を行った後に、リポジトリ管理者がメタデータの品質チェック・修正を行うステップを用意しておくことが現実的であろう。このことはデータ登録から公開までのフローを決定する際も、考慮すべき点となる。

また、データ登録・修正作業の効率化を考えれば、入力画面の使い勝手の良さや一括登録機能も当然要求される。

### 5.4 学術機関リポジトリ用ソフトウェア機能要件

以上を踏まえ、メタデータに関わる観点から学術機関リポジトリソフトウェアに求めたい要件を列挙すれば以下ようになる。

- メタデータ項目の拡張性、カスタマイズの容易さ（入力画面、検索画面に対応）
- OAI-PMH 対応（ダブリン・コア形式、JuNii 形式）
- 管理者用メタデータ修正画面の使い勝手（特に、JuNii に対応した必須項目表示・入力支援機能等）
- 一括登録機能

### 5.5 その他の課題・注意点

前節まで、学術機関リポジトリ稼動に当たっての基本要件というべきものを述べたが、そのほかにもメタデータに関連し、以下のような課題・留意点があるので列記する。

### 5.5.1 メタデータやコンテンツ本体の表層 Web 化

研究成果の流通・顕在性を高めるためには、OAI-PMH を介したメタデータ流通だけでなく、Google 等の検索エンジンからのクローリングにも対応できることが望まれる。そのためには、学術機関リポジトリのデータベース内に格納されている研究成果物個々のメタデータやコンテンツ本体を、検索エンジンのクローラーが収集対象とする Web 上の階層(表層 Web) に晒すような技術的解決が期待される。

### 5.5.2 コンテンツ本体のハーベスティング

コンテンツの保存(バックアップ)や全文検索システムなどへの要求から、OAI-PMH を利用してのコンテンツ本体のハーベスティングを実現しようとする動きも出てきている。

### 5.5.3 保存メタデータ

デジタル形式のコンテンツの長期保存をサポート・促進することを目的とし「保存メタデータ」を整備すべきとの指摘もある。保存メタデータとは、ファイルのフォーマットのバージョン、ファイルサイズ、コンテンツ内容表示に必要なソフトウェア、OS 環境等の情報を記述するものであり、長期間経過後にもコンテンツ本体へのアクセスを保証するためのものである。

<情報源と参考文献>

[メタデータ, DublinCore について]

- 『図書館目録とメタデータ：情報の組織化における新たな可能性』, 日本図書館情報学会研究委員会編 -- 勉誠出版, 2004, ix, 193p. -- (シリーズ・図書館情報学のフロンティア ; No.4)
- Dublin Core Metadata Initiative  
<http://dublincore.org/>
- 日本語による Dublin Core Element Set (杉本先生訳)  
[http://www.dl.u-lis.ac.jp/DC/index\\_J.html](http://www.dl.u-lis.ac.jp/DC/index_J.html)
- 杉本重雄, Dublin Core について (第1回), 情報管理, vol.45, no.4, 2002-7, pp.241-254
- 杉本重雄, Dublin Core について (第2回), 情報管理, vol.45, no.5, 2002-8, pp.321-335

[メタデータ・ハーベスティング, OAI-PMH について]

- Open Archive Initiative  
<http://www.openarchives.org/>
- Open Archives Initiative Protocol for Metadata Harvesting  
<http://www.openarchives.org/OAI/openarchivesprotocol.html>
- OAI-PMH2.0 日本語訳  
<http://www.nii.ac.jp/metadata/oai-pmh2.0/>
- 国立大学図書館協議会図書館高度情報化特別委員会ワーキンググループ「電子図書館の新たな潮流～情報発信者と利用者を結ぶ付加価値インターフェイス」参考資料 1  
『Open Archives Initiative メタデータ収集プロトコルについて』  
[http://www.ll.chiba-u.ac.jp/~joho/project/DLnext/rep\\_plan/dlnext\\_ref1.pdf](http://www.ll.chiba-u.ac.jp/~joho/project/DLnext/rep_plan/dlnext_ref1.pdf)
- OAI-PMH の NII メタデータ・データベースへの適用について  
<http://www.nii.ac.jp/metadata/oai-pmh/>  
実装仕様書  
[http://www.nii.ac.jp/metadata/oai-pmh/implementation\\_as\\_repository.html](http://www.nii.ac.jp/metadata/oai-pmh/implementation_as_repository.html)
- プロバイダリスト <http://www.nii.ac.jp/metadata/oai-pmh/links.html>
- OAIster <http://oaister.umdl.umich.edu/o/oaister/>  
データプロバイダリスト <http://oaister.umdl.umich.edu/o/oaister/viewcolls.html>

[JuNii について]

- NII メタデータ・データベース入力マニュアル 1.2 版  
<http://www.nii.ac.jp/metadata/manual/>
- junii のための XML Schema  
<http://ju.nii.ac.jp/oai/junii.xsd>

[全般]

- SPARC 学術機関リポジトリ チェックリストおよびリソースガイド  
(日本語訳: 千葉大学附属図書館学術機関リポジトリワーキンググループ)  
[http://mitizane.ll.chiba-u.jp/curator/about/SPARC\\_IR\\_Checklist.pdf](http://mitizane.ll.chiba-u.jp/curator/about/SPARC_IR_Checklist.pdf)

### 第 3 部 各大学の状況

## 第1章 北海道大学

以下に北海道大学における学術機関リポジトリ運用に関する取り組みの状況を述べる。

平成16年6月、北海道大学附属図書館において学術情報ポータル検討ワーキンググループを設置した。学術機関リポジトリシステムのソフトウェアとしては、フリーソフトであり、既に海外の各機関で実績のあるDSpaceを、ハードウェアとしては、既存のUnixベースのノートPCを使用しシステムの構築作業を開始した。

DSpaceは本報告執筆時点（平成17年1月末）でバージョン1.2.1 bata4を採用しており、画面、メタデータ検索、コンテンツ全文検索において日本語化の対応を行った。また、OAI-PMHに関しては、oai\_dc形式で対応可能であったが、国立情報学研究所とのハーベスティングテストを経てjunii形式でも対応可能となった。

平成16年7月、国内外の各機関が運用する学術機関リポジトリについての事例調査、学内教員、組織によるインターネットを中心とした情報発信状況の調査、及び、本学における学術機関リポジトリの必要性、学術機関リポジトリの実現方策案の検討を行い、10月に報告書「北海道大学における学術機関リポジトリの在り方について」を作成した。その後、この報告書を元に館内事務部の合意を得て、図書館委員会に報告した。

平成16年11月ー12月、情報発信に関する教員の意識を調査するため、かつ、学術機関リポジトリがどのようなものかを知ってもらうため、学内の教員約2,000名を対象に情報発信に関するアンケート調査を実施した。

平成17年1月、アンケート調査の結果を調査報告書にまとめ、ホームページ上で公開した。更に、各関係者、及び、アンケート調査において「再度連絡しても良い」と回答された方に対して調査報告書、及び、アンケート調査回答に寄せられた「ご意見・ご要望」に関連する簡単なQ&Aを送付した。

平成16年11ー12月、緑化出版社の現状調査を行った。また、主にWeb of Scienceを使用して、過去2年間に北海道大学の教員が各緑化出版社の雑誌にどのような論文を掲載したかという緑化雑誌掲載論文調査を行った。

平成17年1月、緑化雑誌掲載論文調査で得られた北海道大学教員の論文リストに対し、北海道大学研究業績データベース等を使用して教員の日本語名、所属、連絡先を確認し、それらをリストに付加して、データ数約3,000件の北海道大学緑化雑誌掲載論文リストを作成した。

平成16年12月、実験運用・試行運用・本運用のために必要な規程類（登録要領、登

録規程，利用規程，ライセンス，ID 発行申請書等）の調査及び検討を行い，各規程類の草案の作成を開始した。

平成17年1-2月，著者ローマ字ヨミフィールドの追加，日本語化等 DSpace のカスタマイズを行った。

今後の計画だが，試行運用前の準備として，3月から5月にかけて附属図書館による学術機関リポジトリ実験運用を行う予定である。現在，実験運用版の運用方法，各規程，マニュアル類の準備を行っており，準備が整い次第宣伝を行うとともに，11月に行ったアンケート結果等を基に抽出した教員に対し協力を要請していく。実験段階における登録者の範囲は，本学研究業績データベースの収録対象者（本学教員），技能職員，教務職員，図書系事務職員，大学院博士後期課程在籍の学生とし，掲載コンテンツとしては，登録者が単独あるいは共同で，本学在籍中に教育もしくは研究の結果またはその過程で得られた学術的，教育的価値のある公表済み資料，教育資料を対象とする。実験運用において学術機関リポジトリに登録されたデータの公開は学内のみに限定する。なお，学内合意の形成に関しては，この実験と並行に行っていく予定である。

実験運用後は，この実験の過程で得られた意見，結果等を基に運用方法，規程類に適宜変更を加える。その後，学内合意が得られれば，必要なハードウェア，ソフトウェアを再検討し，初期コンテンツを登録，学内教員等への広報を経た後，試行運用に向かう予定となっている。

事項	状況
使用ハードウェア	Apple iBOOK (ハードディスク容量 30GB )
使用ソフトウェア	DSpace1.2.1 beta4
URL	<a href="http://chiri.lib.hokudai.ac.jp:8000/dspace2/index.jsp">http://chiri.lib.hokudai.ac.jp:8000/dspace2/index.jsp</a>
試行運用開始時期	2005年4月以降の予定。運用実験終了後開始
日本語対応	日本語化対応済み(画面，メタデータ検索，コンテンツ全文検索)
OAI-PMH 対応	oai_dc 形式対応。junii は対応確認済み。
MH 可能時期 (Base URL, metadataPrefix)	試行運用開始後。 <a href="http://chiri.lib.hokudai.ac.jp:8000/dspace2-oai/request?">http://chiri.lib.hokudai.ac.jp:8000/dspace2-oai/request?</a>
掲載コンテンツ	(○)商業誌掲載論文 (ポストプリント)
◎ 主として注力	(○)商業誌掲載論文 (プレプリント)
○ 搭載可能とする	(○)国内学会誌掲載論文 (ポストプリント)
△ 検討中	(○)国内学会誌掲載論文 (プレプリント)
× 対象外	(○)学会発表資料 (○)科研報告書



	<p>(○)紀要論文</p> <p>(○)学位論文</p> <p>(○)その他（上記以外の公表済学術的資料，及び教育資料）</p> <p>( )制限せず</p>
登録者の範囲	<p>本学「研究業績データベース」の収録対象者</p> <p>技能職員，教務職員，図書系事務職員，大学院博士後期課程在籍の学生</p>
数値目標等（あれば）	
権利処理について （IR／研究者）	館内検討中。
権利処理について （研究者／学術雑誌）	館内検討中。
館内合意，学内合意等の状況	館内WGにて報告書を作成。館内合意済み。
図書館は何を行うか	館内検討中。
コンテンツ収集（増加）の方策	館内検討中。
本プロジェクト終了後の（本格運用など）見通し	2005.01.31 現在 IR 運用実験の準備中。実験は学内のみに公開し，その結果を基に試行運用を行う予定。

## 第2章 千葉大学

### 2.1 概要

千葉大学では、現在、附属図書館を中心として、「千葉大学学術成果リポジトリ計画」を推進しており、平成16年度中の正式運用開始を予定している。

これは、千葉大学内で生産された多様な研究成果（論文、教材、画像・音声データ等）を対象とした、千葉大学版の学術機関リポジトリの構築をめざしたプロジェクトである。

### 2.2 検討経緯と現状

#### 2.2.1 ワーキング・グループの活動

まず、平成14年度に館内にワーキング・グループを設置し、学術機関リポジトリの運用に向けて、以下のような予備的な活動を開始した。

##### (1) 学内教官向け調査

平成14年10月、本学の助手以上の全教官（1,243名）を対象に書面を送り、「学術情報の発信に関するアンケート調査」を実施した。回収率は約3割と必ずしも高い数字ではなかったが、本調査を通じて、論文の全文情報の公開を予定している教官が少なからず存在していること、組織的な情報発信支援体制の確立を希望していることなど、学術機関リポジトリに対する潜在的なニーズの存在が明らかになった。

##### (2) 海外の事例調査

計画を開始した当時、国内には学術機関リポジトリの先行例はほとんど皆無であった。そのため、海外の先行事例の調査を行い、それと平行して、SPARCの”Institutional Repository Checklist & Resource Guide”の翻訳を試みた。

##### (3) プロトタイプ・システムの開発

学術機関リポジトリ・システムに求められる機能要件について検討を行い、以下のような基本機能を備えたプロトタイプの開発を行った。

- ① 投稿受理機能
- ② 管理機能
- ③ 検索機能
- ④ メタデータ連携機能

#### 2.2.2 協力者会議及び学術情報発信専門委員会における検討（平成15～16年度）

平成15年7月に附属図書館長の下に、教官と事務官から成る「学術情報発信のための

協力者会議」を設置し、運用開始に向け、以下のような準備作業に着手した。引き続き、平成16年度は附属図書館運営委員会の下に学術情報発信専門委員会を設けて検討を重ねた。

#### (1) 初期データ整備

既存のデータについては、初期の段階で可能な限り予め学術機関リポジトリに登録しておくという方針が採用され、初期データ整備を進めている。平成17年1月現在で、約650のコンテンツが登録されている。

- ・ 紀要類の掲載論文 約500
- ・ プレプリント/テクニカルレポート 約50
- ・ 学内外のウェブサイト（学部、学科、個人のサイト）上で既に公開されているコンテンツ等 約100

上記について引き続き構築を図るとともに、以下のコンテンツについても整備を進めているところである。

- ・ 論文の学術機関リポジトリ掲載を許可した出版社の刊行誌掲載論文
- ・ 自然科学研究科の学位論文 数十 を予定
- ・ その他

#### (2) 運用をめぐる検討事項

「誰が、何を、どのように学術機関リポジトリに登録できるのか?」、「一旦登録されたデータの削除は可能か?」、「学術機関リポジトリの管理責任者は誰か?」といった点、また、著作物利用許諾のあり方について方針を検討し、原案を策定した。

#### (3) 学術雑誌掲載論文の著作権について

出版社や学協会の著作権譲渡に関する方針について調査を行った RoMEO プロジェクトの成果を翻訳し、取りまとめた。また、国内学会の動向についてもサンプル的な調査を行い、今後の参考とした。

#### (4) システム改良

平成14年度に開発したプロトタイプの問題点を解消すべく、より簡便な新インターフェイスの開発等を行った。

また、学術機関リポジトリに関する各種 Web ページを一元化した新しいホームページを立ち上げた。 <http://mitizane.ll.chiba-u./curator/>

## 2.3 今後の予定

年度内に全学に向けて、正式発足を通知し、学内研究者によるコンテンツの登録（セルフアーカイブ）の案内を行う予定である。

運用開始後は、引き続き情報発信専門委員会を中心とし、初期データ構築の推進、運営方針や著作物利用許諾方針等の見直し、学内向け広報戦略の策定、学内外の関連組織との連携等の課題に取り組んでいきたいと考えている。

事項	状況
使用ハードウェア	Dell PowerEdge 600SC (ハードディスク容量 80GB )
使用ソフトウェア	独自開発システム (CMS 社) <ul style="list-style-type: none"> <li>・ 採録誌情報の拡張定義を行う (Qualifier に bibliographicCitation を採用予定)</li> <li>・ 登録インタフェース (簡易・一括), 検索インタフェース開発</li> </ul> DSpace1.1.1 <ul style="list-style-type: none"> <li>・ 実験機として運用し, 現行システム改善に役立てる。</li> <li>・ 学内 LAN 接続済み。固定 IP, ドメイン取得予定。</li> <li>・ 1.1.2 にバージョンアップ予定</li> </ul>
URL	ホームページ: <a href="http://mitizane.ll.chiba-u.jp/curator/">http://mitizane.ll.chiba-u.jp/curator/</a> BaseURL: <a href="http://mitizane.ll.chiba-u.jp/cgi-bin/oai/oai2.0">http://mitizane.ll.chiba-u.jp/cgi-bin/oai/oai2.0</a>
試行運用開始時期	プロトタイプ版試行入力 平成 15 年度～
日本語対応	対応している 文字セットは UTF-8
OAI-PMH 対応	対応している (oai_dc, ju_nii に対応)
MH 可能時期 (Base URL, metadataPrefix)	すでに可能
収載コンテンツ ◎ 主として注力 ○ 搭載可能とする △ 検討中 × 対象外	(◎)商業誌掲載論文 (ポストプリント) (◎)商業誌掲載論文 (プレプリント) (○)国内学会誌掲載論文 (ポストプリント) (○)国内学会誌掲載論文 (プレプリント) (○)学会発表資料 (○)科研報告書 (○)紀要論文 (○)学位論文 (○)その他 ( ) ( )制限せず

登録者の範囲	(1) 本学に在籍する, または在籍したことのある教職員及び大学院生 (2) その他館長が特に認めた者
数値目標等 (あれば)	(1) NIIによる電子化紀要等の初期登録データ 500件程度登録済 (千葉大学経済研究, 千葉大学看護学部紀要, 言語文化論叢) (2) セルフアーカイブ可能な大手 EJ に近年掲載された論文 500件程度 (許諾依頼中) Elsevier, Blackwell, Kluwer, Nature, AMS, APS
権利処理について (IR/研究者)	初回コンテンツ登録前に, 登録者から図書館長宛に書面で著作物利用許諾文を提出
権利処理について (研究者/学術雑誌)	研究者-出版社・学会間交渉の仲立ち・代理: 学術雑誌の出版社の著作権ポリシー (特に, 掲載論文のセルフ・アーカイビングを許すか否か) に関する情報を図書館が研究者に積極的に提供する。
館内合意, 学内合意等の状況	館内合意: 運営委員会において事業実施の了解済 (平成 15 年度) 学内合意: 情報企画委員会で了解済。
図書館は何を行うか	1. 学内プロモーション (コンテンツ登録促進) ・ 図書館員による登録支援 ・ 学術雑誌のセルフ・アーカイビングについての情報提供 2. 維持管理 ・ データ管理 (メタデータ充実, 一次データのバッチ取込等) ・ システムメンテナンス ・ 一次データ (コンテンツ) の長期保存 ・ 運営指針の策定
コンテンツ収集 (増加) の方策	1. 簡便な登録インターフェースの提供 2. 図書館員による登録支援登録代行 ・ 登録の代行 (柔軟に対応) ・ セルフ・アーカイブ可能な論文を図書館が登録代行 ・ セルフ・アーカイブ可能な論文の提示 3. 学術機関リポジトリ登録のメリット強調 ・ 例えば, 投稿論文へのアクセス数通知による登録喚起 4. 学内で構築計画が進んでいる「教員業績等データベース」との連携
本プロジェクト終了後の (本格運用など) 見通し	年度内に正式スタートすることで決定 運用指針原案, ユーザインタフェース等は準備済み。

学術機関リポジトリ設置の効果の評価方法	研究成果の視認性向上を計ることのできる指標を今後検討したい。
---------------------	--------------------------------

### 第3章 東京大学

東京大学では本報告執筆時点（2005.1）で、ようやく（少数の教員の範囲ではあるが）試行運用の端緒が具体化してきたところである。

本実験プロジェクトには情報基盤センター デジタル・ライブラリ係が参加してきたところであるが、試行運用から本格運用へと繋げていくためには政策的な面でも附属図書館との調整・協力が必要である。

附属図書館（事務部長）との協議の結果、学術機関リポジトリ本格運用にいたるまでの文書化された計画表が存在しないことから、おおよそ3期計画からなる基本計画表をまず作成し、附属図書館長等への説明と了解を得る予定である。

この計画表は現在作成中であるが、第1期を試行運用、第2期を本格運用（1）、第3期を本格運用（2）とし、最終段階の第3期は学術機関リポジトリの「理想郷的状态」すなわち教員自らが自発的に当たり前の事として自著論文を学術機関リポジトリに登録する状態・・・とすれば、実質は2期計画とも言えよう。

第1期試行運用は以下の予定である（今後の図書館との協議の結果変わりうる）。

システム構築・整備面では

- NIIの実験と連携した試行運用。ソフトウェアはEPrintsによる。
- 使用するシステムについてEPrints以外のソフトウェアに関する情報を収集し、本格運用にふさわしいソフトウェア、ハードウェアスペックを検討する。
- 本格運用に対応したシステムを構築し、データを移行する。

対象となるコンテンツの検討およびその確保としては

- 当面は、情報システム室委員、附属図書館研究開発室員などの範囲で、論文を収集する。
- 本格運用に向けて、全学的データ収集のための方策を検討する。
- 本格運用時に学術機関リポジトリに移行可能なコンテンツも開拓しておく（例：新領域修士／博士論文本文 etc）

システムの運用としては

- メタデータの登録は、情報基盤センターデジタル・ライブラリ係が代行する。
- 著作権等権利処理の問題を洗い出し、問題があればそれへの対応策を立てておく。また、システム上で処理できるような機構の構築も検討する。

広報面としては

- 教員等の理解を深めるための広報活動を行う。

- ・ 図書館職員への啓蒙活動を行う。
- ・ 教員に対して学術機関リポジトリに関する意識調査を実施する。

を考えている。

すでにこれらの項目の中に、より具体的に考えようとするれば相当困難な課題（著作権処理の問題等）も含まれているが、それらも含めてオーソライズされた計画として文書化しておくことが大切であろうと考える。

試行運用に関してももう少し具体的な動きとしては、附属図書館事務部長を通して（上述の計画案にある通り）、附属図書館長、情報システム室委員（情報システム室＝情報担当理事の下で、既存の計算機システムと新規開発システムによる大学総合情報システムの再構築について検討する）、附属図書館研究開発室員等の教員に対し、個別に働きかけを行っており、許諾書を取り交わした上で学術雑誌掲載済み論文（事前にデジタル・ライブラリ係で「緑化誌」掲載済み論文を調査）、卒業時の博士学位論文（紙原稿をデジタル化する！）の提供を受ける予定である。

事項	状況
使用ハードウェア	DELL PowerEdge 2300 （ハードディスク容量 9GB）
使用ソフトウェア	FedoraCore 1 Apache 2.0.47 Mod_perl 1.99 MySQL 3.23 Eprints 2.3.3
URL	<a href="http://eptest100.dl.itc.u-tokyo.ac.jp/">http://eptest100.dl.itc.u-tokyo.ac.jp/</a>
試行運用開始時期	平成 17 年 2 月
日本語対応	入力可。中間報告時と変わらず
OAI-PMH 対応	NII とのハーベスティング実験済み
MH 可能時期（Base URL, metadataPrefix）	可能な状態にある。



<p>収載コンテンツ</p> <p>◎ 主として注力</p> <p>○ 搭載可能とする</p> <p>△ 検討中</p> <p>× 対象外</p>	<p>(◎)商業誌掲載論文 (ポストプリント)</p> <p>(×)商業誌掲載論文 (プレプリント)</p> <p>(◎)国内学会誌掲載論文 (ポストプリント)</p> <p>(×)国内学会誌掲載論文 (プレプリント)</p> <p>(○)学会発表資料</p> <p>(○)科研報告書</p> <p>(○)紀要論文</p> <p>(○)学位論文</p> <p>(×)その他 (特許情報は扱わないつもり)</p> <p>( )制限せず</p>
登録者の範囲	学内教官 (助手以上) で協力してくれる方は誰でも。
数値目標等 (あれば)	特になし
権利処理について (IR/研究者)	ユーザ登録時 (あるいは論文登録時) にあわせて公開にかかる許諾もシステマ的に取得できないか検討する。
権利処理について (研究者/学術雑誌)	未定だが、緑化誌に関する情報の広報、当初は処理代行 (あるいは教官の権利処理に係る負担を軽減するためのサポート) を図書館等が行わざるを得ないのではないかとと思われるが、この点は図書館側と今後協議する。
館内合意, 学内合意等の状況	附属図書館と協議し、学術機関リポジトリ構築に係る計画を策定。附属図書館長 (次期総長) からも論文提供の許諾を得られた (らしい)。
図書館は何を行うか	少なくとも情報基盤センターとしては、本格運用に耐えうるシステムを検討・構築する。技術面サポートと内外の動向リサーチ。図書館には政策面でのサポートを期待したい。
コンテンツ収集 (増加) の方策	「試行運用システム」の段階では特に考えていないが、計画案に沿ってさらに具体的な方策を検討する。
本プロジェクト終了後の (本格運用など) 見通し	当面 Eprints での試行運用を行うが、その間他システムの評価も行う。計画案に沿って学術機関リポジトリ構築を進めるが、本格運用とする具体的時期については未定である。

## 第4章 東京学芸大学

本学は中期計画のなかで「研究成果内容を公表するシステム（研究内容データベース等）を整備する。」とうたっており、学術機関リポジトリはこれを実施する仕組みの一つと考えられる。

東京学芸大学附属図書館は、学内情報関連組織の連携を図る「総合メディア機構」（平成17年度設置予定）の中核となることから、学術機関リポジトリを主体的に構築すべく作業を進めることとしている。

### 4.1 現状

平成16年6月のテスト構築開始時において、学術機関リポジトリのソフトウェアは、海外機関での実績と論文投稿・査読機能の利用の可能性を考慮して、DSpace 1.1.1を用いることとした。その後、全文検索機能を持つDSpace 1.2がリリースされたため、これにバージョンアップしている。なお、ハードウェアは、いずれもOSにFedora Core 1をインストールしたノートPCを用いている。

同年秋には、日本語対応\*（主要画面の日本語化及び2バイト文字の日本語単語検索機能）を進め、11月にはjunii形式でのメタデータ・ハーベスティング実験に成功した。また、DSpace デフォルトのoai\_dc形式でのメタデータ・ハーベスティングも運用方針が整えば開始可能な状況である。

これに並行して、テストデータの入力及び投稿・査読環境の整備を進め、11月末にプロトタイプとして公開した。現在、図書館委員会・学内情報関連委員会でデモ等を行い、認知度を高めつつ試行の検討に入る準備を行っている。

（\*日本語全文検索も平成16年度内対応予定）

### 4.2 今後の課題

#### [学内合意形成]

現時点のプロトタイプ構築は、すぐに可能なことから実施した試験的な性格が強い。今後プロトタイプシステムによる試行を行い、学術機関リポジトリシステムの有用性を実証しつつ、他の学内情報システムとの整合性の確認や制度上の整備を実施し、学内の合意形成を図る。前述の「総合メディア機構」が平成17年度に立ち上がるため、今後は機構の枠組の中で検討が進められる。

#### [データ収録基準]

実験段階においては、「インターネット公開」許諾済みの本学紀要論文の入力にとどまっ

ており、他の学術雑誌掲載論文（ポストプリント）の収録については、国内学会・出版社の動向も見ながら検討されることになる。学位論文の全文公開については大学として取り組むべき問題になるが、まだ議論を始めていない。また、本学の特色を活かし附属学校等での教育実践データやデジタル教材に重点を置いて収録を進める構想もある。

なお、教材に関連して、当館では、大学での授業における「電子的授業支援サービス」の試行を行っており、講義録等の授業関連資料を積極的に入力している教員もある。永続的な蓄積・社会への発信を行う学術機関リポジトリとは性格を異にするものではあるが、学術機関リポジトリに搭載し得る内容も含まれ、入力インターフェースや管理運営の面からも参考にしたい。

#### [メタデータ入力体制・入力基準]

教員自身に全てのメタデータを入力させることは現実的でないので、教員がどこまで入力するか、図書館側がどのような体制でメタデータをチェックし充実させるかの検討が重要である。また、入力項目等のメタデータの入力基準についても試行運用の中で資料タイプごとに検討を深める必要があり、その際はメタデータ・ハーベスティングを通じた他機関との相互運用性も配慮されるべきである。

#### [論文投稿・査読機能]

学術機関リポジトリのソフトウェアにも関わり運用時に再度検討されるべきではあるが、現時点では本学紀要向けに論文投稿・査読機能を備えたものが望ましいと考えている。

事項	状況
使用ハードウェア	FMV-6000NU/L (ハードディスク容量 20GB )
使用ソフトウェア	DSpace 1.2
URL	<a href="http://repository.u-gakugei.ac.jp/dspace/">http://repository.u-gakugei.ac.jp/dspace/</a>
試行運用開始時期	2004年11月
日本語対応	2バイト文字の日本語単語検索対応済み。画面は概ね対応済み。全文検索は平成16年度中に対応予定。
OAI-PMH 対応	oai_dc形式(DSpace デフォルト), juniii形式で可能。
MH 可能時期 (Base URL, metadataPrefix)	junii は対応確認済み。公開状態での定期ハーベスティングは本格運用後を予定。 ( <a href="http://repository.u-gakugei.ac.jp/dspace-oai/request?">http://repository.u-gakugei.ac.jp/dspace-oai/request?</a> , junii 及び oai_dc)
掲載コンテンツ ◎ 主として注力 ○ 搭載可能とする	<input type="checkbox"/> 商業誌掲載論文 (ポストプリント) <input type="checkbox"/> 商業誌掲載論文 (プレプリント) <input type="checkbox"/> 国内学会誌掲載論文 (ポストプリント)

△ 検討中 × 対象外	( )国内学会誌掲載論文 (プレプリント) ( )学会発表資料 ( )科研報告書 (◎)紀要論文 ( )学位論文 ( )その他 ( ) ( )制限せず
登録者の範囲	試行対象は『東京学芸大学紀要』に投稿された論文等で、著者が「インターネット公開」許諾済みのもの。(許諾は2003年以降)
数値目標等 (あれば)	試行対象は、数十件
権利処理について (IR/研究者)	試行対象の『東京学芸大学紀要』収載対象論文については、著者の「インターネット公開」許諾あり
権利処理について (研究者/学術雑誌)	試行対象の『東京学芸大学紀要』については、著者に著作権あり
館内合意, 学内合意等の 状況	本学中期計画の中に「研究成果内容を公表するシステムの整備」があり、その実施方法の一つとして図書館委員会・学内情報関連委員会でデモ等を行い、学術機関リポジトリシステムの認知度を高めている。本格的な合意形成は平成17年度以降の課題。
図書館は何を行うか	平成16年度は図書館側で試行対象の入力を行った。今後図書館が主体となって、データ収録方針・データ入力基準、入力体制・運用体制等の検討を進める予定。
コンテンツ収集 (増加) の方策	平成17年度検討予定
本プロジェクト終了後の (本格運用など) 見通し	<ul style="list-style-type: none"> <li>学術機関レポジトリとしては、DSpaceのみならず市販パッケージを含めたソフトウェアやシステムの比較検討が必要と思われる。運用コストやサポートの受けやすさ等も重要になる。</li> <li>今後の入力対象コンテンツについては、本学の特色を活かした教育実践データやデジタル教材に重点を置く構想がある。いずれにせよ、図書館側で一定の質・量のデータを入力してから、研究者の入力を募るのが現実的と考える。</li> <li>コンテンツ増加策や著作権の扱いについては、平成16年度の実験を踏まえて検討することになる見通し。</li> </ul>

## 第5章 名古屋大学

### 【概況】

名古屋大学では、学術機関リポジトリの試験的構築について附属図書館が中心となって進めているほか、附属図書館研究開発室及び情報連携基盤センター学術情報開発研究部門が種々の側面で連携して対応している。現時点では、学術機関リポジトリの試行について仮のサーバーに研究紀要若干数を搭載した程度で、学内でも関係者が閲覧しているレベルである。

また、今後の推進については、試行用サーバーの準備などのほか、学内研究紀要をさらに搭載して学内に公開し、研究者の理解を得て本格的な稼働状態に進めていく予定とした「名古屋大学における学術機関リポジトリ構築計画」を策定している。

### 【年度計画への盛り込み】

国立大学法人としての中期目標・計画において、附属図書館（部局）の年度計画として、学術機関リポジトリの試行を盛り込んでいる。（情報連携基盤センターでも附属図書館と協力して学術機関リポジトリの研究開発を進めることを盛り込む予定）

### 【構築計画の概要】

名古屋大学附属図書館において策定された「名古屋大学における学術機関リポジトリ構築計画」（平成16年11月25日）の骨子を以下に示す。

- 学術機関リポジトリを巡る動向
- 学術機関リポジトリの特徴等
- 学術機関リポジトリ構築による期待効果
- 学術機関リポジトリ構築のための基本要件
- 名古屋大学における学術機関リポジトリ構築
  - ◇ 基本的な方針
  - ◇ 構築への手順
    - 短期的戦略
    - 全学的な展開への方策（永続的な運営・管理のための方策）

### 【具体的検討組織】

#### 1. 附属図書館商議員会電子図書館推進委員会

附属図書館の各部局の代表からなる最高意思決定機関である商議員会のもとに作られた組織であり、商議員の他、専門的な観点から附属図書館研究開発室、情報連携基

盤センター学術情報開発研究部門、情報メディア教育研究センター及び事務局情報企画課から委員が参加している。

## 2. 情報連携基盤センター学術情報開発専門委員会

電子図書館の基盤技術等を研究開発する情報連携基盤センター学術情報開発研究部門の専門委員会であり、特に学術機関リポジトリは、この委員会のもとに設置された「情報流通 WG」で議論されている。この WG には、附属図書館研究開発室及び附属図書館事務組織からもメンバーが参加している。

### 【その他関連活動】

平成 16 年 12 月 17 日に開催された東海地区大学図書館協議会の主催する研修会において学術機関リポジトリ構築に関する事例紹介を行い、北陸地区を含めた地区内の大学図書館関係者に啓蒙活動を行った。

### 【具体的な試行手順】

現在稼動している紀要情報照会システムの内容を学術機関リポジトリに移植するためには、以下のような手順を予定している。

- ・ 現在の紀要情報照会システムから、メタデータにあたる部分（著者・タイトル・雑誌名等）を抽出する。
- ・ 同システムから、論文本体にあたるファイルを抜き出す。各ページを jpeg 形式で格納していたものについては、まとめて単一の pdf ファイルにコンバートするのが適当である。
- ・ メタデータと論文本体を対にして、順次学術機関リポジトリ内に格納する。

2005 年 1 月現在、紀要情報照会システムには、8 タイトルの紀要について、64 の pdf ファイルと 16,234 の jpeg ファイル（論文 683 タイトル分）が格納されている。このシステム上において公開することの著作者の承諾はすでにあるが、学術機関リポジトリ上においても、改めてそれぞれの著作者に承諾を得る手続きを予定している。

学術機関リポジトリの実装としては DSpace を採用することを検討しており、以下のような画面デザインのもので試行運用を進める予定である。



事項	状況
使用ハードウェア	Dell PowerEdge1800 (仮) (HDD 40GBx2)
使用ソフトウェア	DSpace 1.2.1 (ハンドルサーバーを導入することを検討中)
URL	http://air.nul.nagoya-u.ac.jp/dspace/ (予定)
試行運用開始時期	2005年4月程度を予定
日本語対応	画面とメタデータ部分の検索を日本語対応
OAI-PMH 対応	oai_dc 形式, junii 形式
MH 可能時期 (Base URL, metadataPrefix)	2005年4月程度を予定 http://air.nul.nagoya-u.ac.jp/dspace-oai/ (予定)
収載コンテンツ ◎ 主として注力 ○ 搭載可能とする △ 検討中 × 対象外	(○)商業誌掲載論文 (ポストプリント) (△)商業誌掲載論文 (プレプリント) (○)国内学会誌掲載論文 (ポストプリント) (△)国内学会誌掲載論文 (プレプリント) (△)学会発表資料 (△)科研報告書 (◎)紀要論文

	(○)学位論文 ( )その他 ( ) ( )制限せず
登録者の範囲	名大構成員（院生 or 教員）を予定
数値目標等（あれば）	紀要情報照会システムに登録してあるもの（700件程度）を継承することを当面の目標としている
権利処理について （IR／研究者）	紀要情報照会システムへの登録時に用いた事務手続を叩き台に、原案を作成中
権利処理について （研究者／学術雑誌）	検討中
館内合意, 学内合意等の 状況	合意形成の趣旨や手順について原案を作成中 （全体計画は作成済み）
図書館は何を行うか	図書館が学術機関リポジトリを主体的に管理する方向で検討中
コンテンツ収集（増加） の方策	公開許諾手続（検討中）により収集する予定
本プロジェクト終了後の （本格運用など）見通し	平成 17 年度後半の本運用を目標としている



## 第6章 九州大学

九州大学での実装実験は、現代的教育ニーズ取組支援プロジェクト（以下「現代 GP」と言う。）による e-Learning コンテンツのメタデータ整備と並行して実施中である。

### 6.1 システム

当初は、既存のサーバ（富士通製 GP7000 シリーズ/Soralis）を使用して DSpace の日本語化、バッチ処理によるデータの一括変換をテスト的に実施していたが、本年 1 月から現代 GP により新たに調達されたサーバに移行した。サーバの仕様は以下のとおりである。

- ・ハード

NEC Express 5800/110Gb

CPU : Pentium4 3.4 GHz

メモリ : 2GB

ハードディスク : 250GB(RAID 1 構成)

- ・ソフトウェア

OS : Red Hat Enterprise Linux ES Ver3

リポジトリ : DSpace 1.3.33

Web : apache 1.6.2

DBMS : PostgreSQL 7.4.6

アプリケーションサーバ : Tomcat 5.0.28

向う 3 年程度はこのサーバによる運用を行うが、学術機関リポジトリが全学的に認知されデータ量が大幅増加した場合は、別途大規模サーバの導入を想定している。

### 6.2 体制

現行の体制は、附属図書館の担当掛及び情報基盤センターの担当掛各 1 名の職員が実験の実務を行っている。現代 GP の調整は医学・医療系部局及び情報基盤センター教員等による担当者会議により行われているが、図書館の情報システム課長がこの担当者会議のメンバーに加わることにより二つのプロジェクトの仲介役となっている。

実装実験終了後となる 4 月以降は、メタデータ付与等の電子的資料の組織化を担う事務組織が図書館に誕生し、この組織を中心に図書館を挙げて学術機関リポジトリの整備が行われることになる。

## 6.3 収録対象

### (1) 学内刊行雑誌

Web サーバで公開中であり、これまで図書館が電子化・公開を行ってきた ESKIA を学術機関リポジトリへ移行するため、バッチ処理テストを兼ねた作業を実施中である。このほか、いくつかの部局から電子化と電子的公開への要望があり、現在協議を進めている。

### (2) 学位論文

特許等の知的財産に関して管理している部局との合意がないため、XML ファイルの生成に止まっており、まだサーバへの搭載は行っていない。しかし、キャンパス移転に関連して学位論文の図書館への管理移管の要望があるため、今後関係部局との協議を開始する予定である。

### (3) e-Learning コンテンツ

e-Learning 導入の必要性かつ教育的効果が最も高い医療系教育において、統合的な e-Learning システムを構築することによって、より良き医療人を育成することを目的としています。全国で唯一の本学「医療系統合教育研究センター」を中心に医学部・歯学部・薬学部・附属病院が教育コンテンツを協同で開発・提供し、附属図書館が提供された教育コンテンツの権利関係およびメタデータ付与に関する調整を行い、情報基盤センターの e-Learning システムにより教育コンテンツを e-Learning コースとして展開します。

(<http://rd.cc.kyushu-u.ac.jp/gp/> 九州大学 現代的教育ニーズ取組支援プログラム「WBT(Web Based Training)による医療系統合教育) より抜粋

上記により、図書館では、Learning Object (LO) に対し NII の JuNii 形式、独立行政法人メディア教育開発センター(NIME)が予定している LOM (Learning Object Metadata) 規格に基づく索引語を付与する。なお、現在メタデータを付与する単位(コース、講義、素材)、権利処理について先述の担当者会議で協議中である。

## 6.4 今後の計画

コンテンツの収集・搭載は、当面現在進行中の ESKIA、現代 GP を中心にサーバへの搭載を進めてゆく方針である。一方、学位論文・紀要に関して、書誌事項の提供フォーマット、権利処理等について関係者との協議を進め、互いに負担とならないようなルーティンを確立し、実運用を開始したい。実運用が他部局等へアピールとなり、収録範囲の拡大に繋がると考える。

一方、研究者自身の手による学術機関リポジトリ構築を考慮するうえでは、試用中の DSpace のユーザインターフェイスには、以下のような問題があるため、改造または商用アプリケーションの導入を検討する必要がある。

- (1) 例えば DSpace ではログイン後にファイルのアップロードを完了するまでに、画面を4つ展開する必要があり「単にファイルを置くだけ」と考えるならば大変な手間と感じる。研究者に「入力項目が多すぎる」と感じさせてしまつては、日常的な運用は困難であるとする。解決には、研究者向け画面として入力項目を最小限とした登録専用画面と、メタデータを付与するための管理用画面が必要となる。
- (2) 統計機能として、研究者へ利用状況を自動的に報告する仕組みが必要である。自身が登録した情報がどれだけ利用（ダウンロード）されたかが明確になれば、研究並びに搭載への励みとなる。

事項	状況
使用ハードウェア	NEC Express 5800/110Gb (ハードディスク容量 20 GB )
使用ソフトウェア	DSpace 1.2.1
URL	<a href="http://mars.lib.kyushu-u.ac.jp/">http://mars.lib.kyushu-u.ac.jp/</a>
試行運用開始時期	2005年2月
日本語対応	画面日本語化完了, 検索可能
OAI-PMH 対応	oai_dc, JuNii 形式で可能
MH 可能時期 (Base URL, metadataPrefix)	4月1日頃 ( <a href="http://mars.lib.kyushu-u.ac.jp/dspace-oai_oai_dc">http://mars.lib.kyushu-u.ac.jp/dspace-oai_oai_dc</a> )
収載コンテンツ ◎ 主として注力 ○ 搭載可能とする △ 検討中 × 対象外	<input type="radio"/> 商業誌掲載論文 (ポストプリント) <input type="radio"/> 商業誌掲載論文 (プレプリント) <input type="radio"/> 国内学会誌掲載論文 (ポストプリント) <input type="radio"/> 国内学会誌掲載論文 (プレプリント) <input type="radio"/> 学会発表資料 <input type="radio"/> 科研報告書 <input type="radio"/> 紀要論文 <input type="radio"/> 学位論文 <input type="radio"/> その他 (講義資料, 電子教材類) <input type="checkbox"/> 制限せず
登録者の範囲	本学在席の(過去在籍も含む)教員及び研究者の研究成果物。
数値目標等 (あれば)	
権利処理について (IR/研究者)	来年度検討
権利処理について (研究者/学術雑誌)	来年度検討

館内合意, 学内合意等の状況	研究開発室の先生に紹介。「現代的教育ニーズプロジェクト」の活動を通じて学内合意形成を図った。
図書館は何を行うか	権利処理, メタデータ作成
コンテンツ収集 (増加) の方策	研究開発室, 館内合意, 現代的教育ニーズプロジェクトを通じて学内合意を進めていく予定。
本プロジェクト終了後の (本格運用など) 見通し	「現代的教育ニーズプロジェクト」で作成された医学系の e-Learning コンテンツを収集する。 また, 17 年度以降は, 事務組織の改変を行うことで, より組織的に学術機関リポジトリ整備にあたる予定。

## 第7章 金沢大学

金沢大学では、平成12年8月に「キャンパス・インテリジェント化構想検討報告書」がまとめられた。平成12年12月には報告書の構想を踏まえて「キャンパス・インテリジェント化実施計画に関する具体的提案」(1)が策定され、印刷・刊行されている。

提案は、キャンパス全体のIT化について基本的な考え方に触れながら、情報研究の推進、情報処理教育の実施計画、学術情報の蓄積・利用と情報発信、学務情報化、事務情報化など大学に必要なほとんどの情報化を対象として、具体的方策を提起している。

このうち、いわゆる学術情報の発信については、従来は「学協会あるいは出版社等が発行する学術雑誌の研究論文として公表されてきた」が、最近では、「より速く成果を世界に情報発信する活動が盛んになりつつある」として、「大学全体あるいは部局単位の組織的な取組みが必要」とであると指摘している。

そのための方策としては、1. 研究成果の公表・組織的取組み、2. 研究者データベースの充実、3. 地域への情報発信と研究交流、4. 世界への情報発信、5. 情報発信のための機器整備が掲げられ、これにより、「本学からの研究情報、成果発信体制が確立でき、発信情報の内容充実とによって、社会への研究活動の説明責任を果たせる」としている。こうした方針にあわせて、特に附属図書館では、全国的な電子化・マルチメディア化の動向を踏まえて、「研究業績と合わせてインターネットなどによる積極的な公開、情報発信を計画的に実施すべき」と提言している。

一方、金沢大学では法人化に当たって全学の事務組織の改組が行われた。これに伴い、附属図書館、総合メディア基盤センター、情報処理課、社会貢献室、大学開放センターなどが情報部の下に統合された。また、大学の基幹組織として5つの企画会議が設置され、情報部は情報企画会議に所属することになった。情報担当理事が附属図書館長を兼任し、情報企画会議を所掌する。

情報企画会議及び情報部の所掌は、(1)大学全体の情報戦略にかかる枠組みの策定と(2)地域・社会貢献に大別できる。これにより、大学の情報インフラ(=「キャンパス・インテリジェント化実施計画」)に関しては、情報部が取り扱うこととなった。

情報企画会議では先の提案を受けて、平成16年度キャンパス・インテリジェント化WGを設置し、インテリジェント化計画の洗い直しと実施計画の策定を行った。情報部に改組された附属図書館事務部からも2名が参加し、6年間にわたって優先度の高いものから学内予算により実施していくことが決定された。附属図書館からは、Institutional Repositoryを主要な要素とする「学術情報蓄積・発信システム」を提案し、了承された。

キャンパス・インテリジェント化計画には、「大学情報統括管理システムの開発」が含まれている。これは、総合メディア基盤センターが企画しているシステムで、地球観測などの科学データをはじめとして、論文リスト、特許権、学務情報など学内で体系的に保存された研究情報・業績や事務情報に対して、標準的な規格と総合的なインターフェースを提

供する計画である。文献データベースについては、e-print アーカイブ以来の蓄積がある。標準的インデックスとしてのメタデータと相互運用についてもすでに Institutional Repository として欧米での実績がある。この蓄積を統括システムの一部として整合的に提案していくことにより、インテリジェント化計画の学術情報の発信システムを実装し、世界的に本学の研究の visibility を高めることが重要である。その意味で、附属図書館と総合メディア基盤センターとの緊密な協力の必要性が両組織において認識されつつある。

今年度、金沢大学では電子ジャーナルの充実のため、部局の外国雑誌予算の共通経費化の提案（「金沢大学学術情報基盤整備計画」（2））を行い、全部局において関連教員にたいして説明を行った。その際 SPARC や Open Access, Open Archive Initiative などについても触れ、学術コミュニケーションの改善について広報を行った。総合メディア基盤センターのデータベース研究会が昨年9月に開催したデータベースフォーラム（3）においても、附属図書館職員が「学術雑誌の危機と SPARC JAPAN PROJECT」と題した報告を行い同様の広報活動を行った。

これらの複数の学内的な状況を受けて、平成16年末に附属図書館で5名からなる作業グループを作り、repository の実装を行うことが決まった。ハードウェアについては、新規に購入することになった。当面は本格的なシステムの実装と教員組織への働きかけを含めた学内環境の整備に向けて、技術的な要件を検証することを目的としている。また、海外の Institutional Repository の事例調査のため、図書館職員が平成17年度の科学研究費（奨励研究）に応募している。

ハードウェア及びOSは、富士通の Primergy Econel 40 と Red Hat Enterprise Linux ES (v.3)を使用し、repository システムとしては Eprints を実装する予定である。また、学内教員の協力を仰いで紀要関係のコンテンツを試験的に運用する。本報告書が刊行される平成16年度末にはこの実装実験を終え、平成17年度には総合メディア基盤センターとの協力関係を踏まえながら、システムの本格運用と学内の学術コンテンツ収集の条件等について調査・検討を開始する予定である。

他大学に比して1サイクル遅れではあるが、Big deal による電子ジャーナルの充実を学内的に提起して、広く学術コミュニケーションの改善について訴えてきた附属図書館が、デジタルライブラリーという技術的意味付けに留まらない新たなコミュニケーションツールの実装を試み、提案することは時宜にあった事業であり、附属図書館が今後担うべき学内的な課題は多い。

（金沢大学情報部情報企画課課長補佐 内島秀樹）

- (1) 「キャンパス・インテリジェント実施計画に関する具体的提案」（平成12年12月）
  - ・金沢大学キャンパス・インテリジェント化推進コアグループ
  - ・キャンパス・インテリジェント化計画実施検討ワーキング・グループ
- (2) <http://www.lib.kanazawa-u.ac.jp/jimu/ej/index.htm>

(3) <http://www-db.gipc.kanazawa-u.ac.jp/db-ken/>

事項	状況
使用ハードウェア	富士通 Primergy Econel 40 (ハードディスク容量 72GB)
使用ソフトウェア	Eprints 2-3-7-1 (予定)
URL	<a href="http://digital.lib.kanazawa-u.ac.jp">http://digital.lib.kanazawa-u.ac.jp</a>
試行運用開始時期	平成 17 年 4 月予定
日本語対応	対応予定
OAI-PMH 対応	対応予定
MH 可能時期 (Base URL, metadataPrefix)	本年 3 月後半を予定している
収載コンテンツ ◎ 主として注力 ○ 搭載可能とする △ 検討中 × 対象外	(△)商業誌掲載論文 (ポストプリント) (△)商業誌掲載論文 (プレプリント) (△)国内学会誌掲載論文 (ポストプリント) (△)国内学会誌掲載論文 (プレプリント) (△)学会発表資料 (△)科研報告書 (△)紀要論文 (△)学位論文 (△)その他 ( ) (△)制限せず
登録者の範囲	来年度検討
数値目標等 (あれば)	未定
権利処理について	来年度検討
館内合意, 学内合意等の状況	平成 16 年度は館長決済による館内での試行として行う。来年度は総合メディア基盤センターとも協力しながら, 本格的な運用方法について調査・検討していく。
図書館は何を行うか	来年度検討
コンテンツ収集 (増加) の方策	来年度検討
本プロジェクト終了後の (本格運用など) 見通し	<ul style="list-style-type: none"> <li>• Eprints による暫定運用を行いつつ, 他のシステムとの比較を行い, 採用すべきシステムの比較・検討を行う。</li> <li>• それと平行して, 本格運用に向けて学内の体制作りの方策について館内で検討を始める。</li> </ul>

## 附録 国内対応メーカー・サービス一覧



## E-repository 機能

これからの図書館に求められる機能として開発された「E-repository」は学内で、作成・管理している論文プレプリント、学位論文、ソフトウェア、電子教材、ファクトデータ等に関するメタデータベースの構築、並びに一次情報のアップロード機能などで構成された情報管理システムです。

### 【機能概要】

- ・ OAI-PMH(The Open Archives Initiative Protocol For Metadata Harvesting)というプロトコルを実装しており、NII等の国内、国外の機関とのメタデータを自動的に配信収集できます。
- ・ 各機能における検索につきましては、全文検索機能を実装しております。
- ・ 各アクセスポイントは、半角/全角、大文字/小文字、記号のあるなしに関係なく、検索が可能です。
- ・ 利用者認証を行い、利用者の属性によって利用できる機能に変更可能です。
- ・ 不特定ユーザからの利用を可能とするため、ユーザインターフェイスはブラウザでご用意しています。
- ・ 内部で管理する文字コード（メタデータに関する）はUTF8を使用しており、全て多言語環境です。



● **利用者登録・利用者認証機能**

● **投稿受理機能**

● **投稿受理データ管理機**

● **利用者ポータル機能**

開発元：株式会社 シー・エム・エス

販売元・お問い合わせ：NEC文教ソリューション事業部

第三営業部 TEL:03-3798-6293

E-mail : info@elsd.jp.nec.com



## InfoLib-PMH (メタデータ収穫提供システム)

Repository

収集



蓄積



利用

簡易検索 [詳細検索](#)

タイトル

作者または著者

主題およびキーワード

全部の項目

表示件数

検索対象を限定する場合は選択して下さい。  
 コンテンツ  フォルダ  コレクション

マルチメディア統合検索システムパッケージ  
**InfoLib**  
Integrated multimedia search system

**InfoLib-PMH** は、OAI-PMH ver2.0 に準拠したメタデータの収集および提供を行うシステムです。

メタデータの収集を行うハーベスターとメタデータの提供を行うレポジトリの両方の機能をサポートしています。

【レポジトリ】 ●OAI-PMH に準拠したメタデータ・レポジトリ機能を提供します。[ 任意の metadataPrefix/複数の set 定義/秒精度の UTC datetime/resumptionToken/delete 属性を使用した削除メタデータの提供 ]

【ハーベスター】 ●メタデータ・レポジトリよりメタデータを収集。収集したメタデータはメタデータマネージャにより自動的に蓄積管理します。

【ハーベスターマネージャ】 ●WEB ブラウザで複数のレポジトリを登録。レポジトリ毎に収集スケジュールを複数指定。[ 収集間隔・収集範囲・収集条件等を設定 ] ●登録は BaseURL を入力するだけ。収集履歴と共に一元管理し、収集単位のデータを管理します。

【メタデータマネージャ】 ●収集データをメタデータ毎に分解して一元管理。メタデータ 1 件単位に差分の追加・更新・削除処理を自動実行します。●登録管理されているメタデータは、WEB ブラウザで参照・更新・削除が可能です。

【検索機能連携】 ●メタデータ検索システム InfoLib-META や横断検索 InfoLib-GlobalFinder と連携し、横断検索や Z39.50 で世界への発信が可能です。

※ 動作環：Solaris8, Solaris9, Linux

※ InfoLib その他プロダクトについては下記迄。

**infocom** インフォコム株式会社  
〒101-0062 東京都千代田区神田駿河台3-11  
三井住友海上駿河台別館  
TEL 03-3518-3640 FAX 03-3518-3055  
HomePage  
<http://www.infocom.co.jp>

## USACO リポジトリ

### <システムまたはサービスの概要・特徴など>

#### ■ データプロバイダ機能, 登録機能に特化したシンプルな構成

書誌メタデータ及び書誌本体(電子)の登録, その双方に対するアクセス機能を基本に提供します。  
また, 書誌本体に対するデータ管理, ウィルスチェック, リンク出力機能を提供します。

※ 詳細検索等のサービスプロバイダ機能及びローカルインストールサービスは今後開発予定です。

#### ■ サーバーに Fedora Core を採用, UTF-8 エンコードによる完全日本語対応

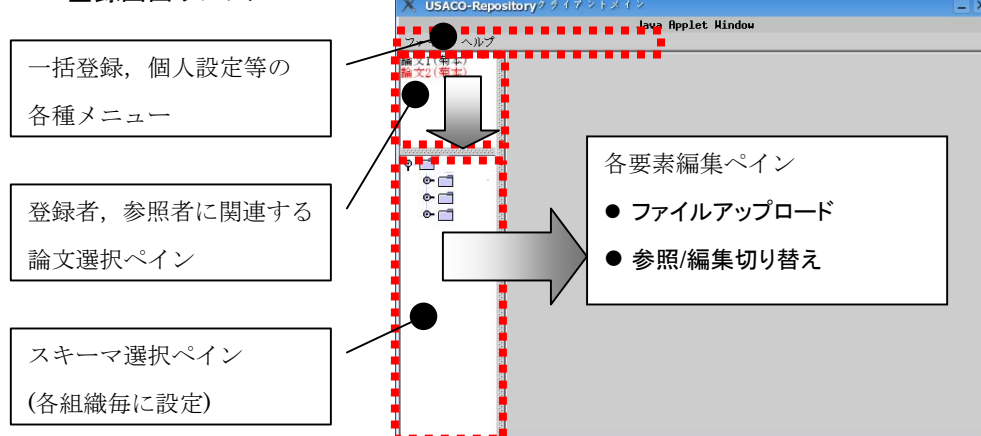
フリーOSによる運用, 開発により, 低価格を実現します。また, 最新機能の利用を促進します。

#### ■ クライアントに Java アプレットを採用(日本語対応, 登録及び管理)

Web ブラウザでの操作不全の解消, インストール負荷の軽減, 及び拡張性の確保を行います。

データの一括登録, 書誌データに対するアクセス統計通知, 登録に際してのチェック(ワークフロー)等の機能を実装します。

登録画面サンプル



#### ■ 各種メタデータフォーマット対応(junii, oai-dc)の OAI データプロバイダ機能

OAI-PMH のメタデータ指定オプションとして実装します。

※ 登録時のスキーマは, 組織ごとに構成及び入力項目(必須等)を設定可能とします。

### <価格, リリース>

1 機関 年間 ¥525,000.-(税込)を予定しております。2005 年 2 月よりデモ利用可能です。

### <参考 URL>

デモサイト : <http://trooper.usaco.co.jp/UR/> ※ 2005 年 2 月より公開予定です。

## iLisSurf e-Lib



iLisSurf e-Lib は、大学、機関に存在するあらゆる形態のデジタル資料（論文、研究成果、貴重書、画像、動画など）を、データベース上で一括管理し、公開、検索を行うことが可能です。

利用者は統一された検索操作により、その特性に合わせて情報を参照することが可能です。また、従来の図書館システムとの親和性を持ちつつ、インターネット上での情報発信・情報検索のための電子図書館機能を提供します。

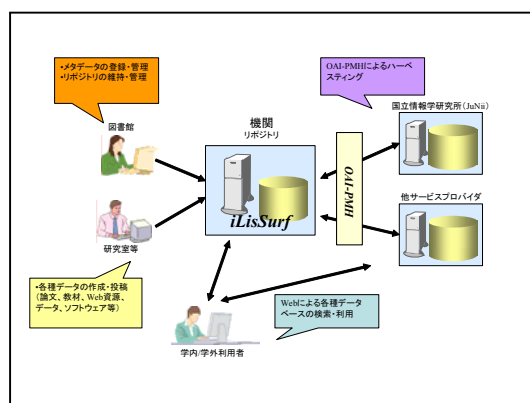
メタデータは国際規格ダブリンコアのデータ形式に準拠しています。また一つのシステムの中で、複数種類のメタデータ形式をサポート可能です。これにより、お客様自身の個々のニーズに合わせて、自由に設定が可能になります。

### ■ OAI-PMH 対応

国立情報学研究所メタデータ・データベースの仕様に準じた OAI-PMH データプロバイダ機能を提供しています。

### ■ システムの特徴

- ・ 高速、高機能な検索・公開サービス
- ・ 全文検索機能
- ・ 図書館業務システムとの親和性
- ・ 多言語のサポート
- ・ 一括登録機能
- ・ 標準への準拠と信頼性・セキュリティの確保
- ・ 安定したシステム、サービス



## NALIS 図書館情報システム

NALIS（次世代図書館情報システム）では、電子図書館システムで実績のあるメタデータ、画像取り扱いの技術をベースに学術機関リポジトリとして使用可能な環境を提供致します。

NIIをはじめとした他機関とのメタデータの交換機能に加え、NALIS 電子図書館システムと共通のインターフェースによるWebブラウザによるメタデータ検索／表示機能（画像表示、他URLサイトへのリンク等）が可能です。（画面、色等については、ご要望によりカスタマイズが可能です。）

主な機能として、以下のものがあります。



### <主な機能>

- ・ メタデータの登録／変更／削除
- ・ OAI-PMH に準拠したメタデータの交換 (oai\_dc, NII メタデータ記述要素 対応)
- ・ ユーザ管理機能（操作者権限設定(管理者, データ登録者等), ユーザ／パスワード管理)
- ・ PDF, 画像等の電子ファイル登録機能
- ・ Web ブラウザによる検索／参照機能

参考 URL : <http://www.nttdata-kyushu.co.jp/product/index.html>

照会先：株式会社 NTTデータ九州

第一開発部 NALIS 営業担当

〒812-0011 福岡市博多区博多駅前 1-17-21 NTTDATA 博多駅前ビル

(電話) 092-475-5129

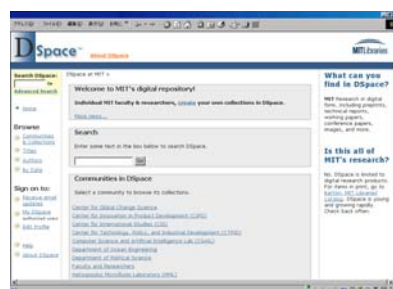
(email) haradamt@nttdata-kyushu.co.jp

## DSpace 構築サービス

マサチューセッツ工科大学 (MIT) 図書館とヒューレット・パッカー研究所(HP)が、共同で開発したオープンソースのリポジトリシステム「DSpace」の導入・構築/付加機能開発等のシステム構築サービスを行います。

本構築サービスは、デジタル化された研究業績結果や貴重資料等をブラウザにより収集・公開することを目的とし館内のみならず全世界へ情報発信することを最終目的とします。OAI-PMH により国内、海外の大学・研究機関への情報提供・収集も行います。

各種一次データ (PDF/HTML/JPEG/MP3 等) の投稿機能、投稿の際に付与されたメタデータ (ダブリンコア / Dublin Core Metadata Element Set) の検索に対応し、またそのメタデータを利用した OAI-PMH のメタデータ・ハーベスティングに対応しています。



### 【主な構築支援サービス】

- ・ DSpace インストール
- ・ ユーザが適用するリソースに適した, qualifire,scheme の実装
- ・ 既存データから DSpace への変換, 投入
- ・ 日本語検索エンジン
- ・ わかり機能実装
- ・ OAI-PMH によるデータプロバイダ機能実装試験 等

お問い合わせ：株式会社 シー・エム・エス

〒171-0021東京都豊島区西池袋3-30-3(西池本田ビル5F)

部署：営業推進部 担当：須藤, 小川

TEL:03-5954-1151 FAX 03-5954-1151

E-mail: ecats@cmsc.co.jp

