

目 次

平成 17 年度の年報によせて	中村 邦彦	1
総合情報基盤センターの概要	神鳥 成弘	2
センター利用状況	瀬野 芳孝	4
平成 16 年度パソコン室利用統計		5
P2P ソフトウェアについて	林 敏浩	8
スパムメール対策について	河内 一芳・川田 延枝・池田 裕子	10
医学部情報教育の感想・問題点	寺中 正人	13
経済学実験の試み	曾 道智・穴戸 栄徳	15
医学部キャンパスにおけるセンターの機器を利用した研究活動	宮中 宏	17
工学部の教育用 PC を利用した地域企業への技術教育	吉村 英徳	19
香川大学ウェブページ検索システムの更新	中村 邦彦	21
Cocoon2 での Xindice 利用 - 基本機能の動作検証を中心として -	青木 昌三	23
PPP 接続ログ集計スクリプトの作成 - スクリプト言語 Groovy を用いて -	青木 昌三	84
センターの広報活動	吉田 裕美	108
研究会等参加報告		110
センタースタッフによる教育活動（平成 17 年度）		111
医学部専門基礎科目を担当して	上原 正宏	113
研究活動報告		
センタースタッフによる研究活動（平成 17 年 1 月～平成 17 年 12 月）		114
反復再生可能型描画システム Polka の開発	林 敏浩	120
クロストリジウム属のバクテリアのコラーゲン分解酵素の進化	竹崎 直子	122
随想 ～ 数学の研究というもの	上原 正宏	124
平成 17 年度活動報告記	今井 慈郎	125
センタースタッフのその他の研究会等参加報告		135
資料		
香川大学総合情報基盤センター規則		137
香川大学総合情報基盤センター運営委員会規則		139
香川大学情報基盤運営専門委員会規程		141
香川大学総合情報基盤センター利用規程		142
香川大学キャンパス情報ネットワーク運営規則		144

香川大学キャンパス情報ネットワーク運営細則	146
2005年開催の会議および行事	149

業務報告

平成16年度利用申請一覧(研究・事務)	151
平成16年度総合情報基盤センターPC利用状況 PCルーム1, 2, 3	158
平成16年度総合情報基盤センターPC利用状況 教育学部情報処理教室1, 2	159
平成16年度総合情報基盤センターPC利用状況 農学部情報処理室1, 2	160
平成16年度総合情報基盤センターPC利用状況 工学部演習室(4301)	161
計算機利用統計	162
平成16年度パソコン室利用統計	163
センタースタッフ, 運営会議, 運営専門委員会メンバー一覧表	166
編集後記	168

平成 17 年度年報によせて

総合情報基盤センター長 中村邦彦

総合情報基盤センターは、平成 15 年 10 月に香川大学と香川医科大学の統合と同時に設置されました。それまで香川大学にあった情報処理センターと香川医科大学にあった情報メディアセンターが単に一緒になったというのではなく、専任教員 8 人と技術職員 3 人を擁する強力な組織として生まれ変わりました。発足時には空席の多かった教員ポストでしたが、17 年 2 月にはすべての教員が揃いました。そうした中で、17 年 10 月に学長の交代があり、それに伴って私がセンター長を勤めることになりました。

実は私は平成 10 年 4 月から 14 年 3 月までの 4 年間、香川大学の情報処理センター長でした。したがって、個人的には 3 年半ぶりにセンター長に復帰したとも言えるのですが、当時の情報処理センターは正式な省令施設ではなく、専任教員がゼロという状態でしたから、今の基盤センターとは比較になりません。当時は情報処理センターを一ランク上の総合情報処理センターへ格上げしようと、何度か文科省へ出向いて要求していました。残念ながら力不足でそれが実現できないまま退任したのですが、今こうして、さらに一ランク上の総合情報基盤センターの長になっていることを考えると、大学がいかに急速に変化しているか、その速さを感じざるを得ません。

さて、この総合情報基盤センターですが、現在いくつかの重要課題を抱えています。

まず、老朽化してきた現ネットワークの維持と更新でしょう。古くなったネットワークスイッチ類について、メーカーが「もう修理できない」と言ってきたのが発端でしたが、これは昨年から今年にかけての関係者のご尽力により、何とか乗り切れそうです。古くなったネットワーク機器は、基幹部分を次期レンタルシステムに組み入れるなどの案も進行中です。

次に、現レンタルシステムの更新があります。現在のシステムは、旧香川大学と旧医科大学がそれぞれ別個に導入し運用してきたものですが、これを平成 19 年 2 月末に統合して更新しようとするものです。医学部とそれ以外の学部との考え方、予算の使い方に違いがあり、その調整が簡単ではありません。また、センター発足時から要求しているセンターの新しい建物は、まだ実現していません。現在、建物の新築は極めて難しい状況ですから、従来の方針にこだわらず、さまざまな案を検討しなければならないでしょう。

さて、この年報は、今回から CD-ROM とウェブを主体にした出版に変えることになりました。これも新しい時代に対応しようというスタッフの意気込みの現れとご理解いただきたいと思います。

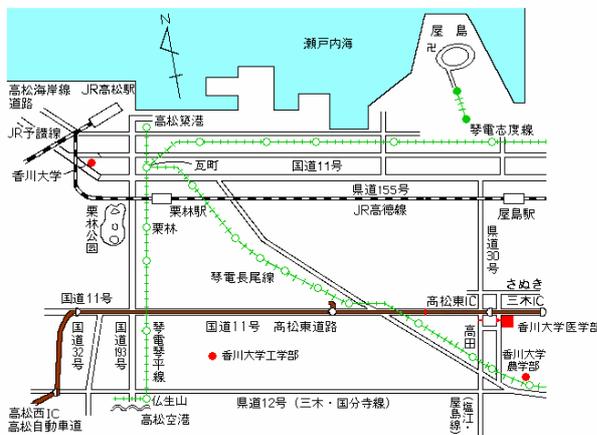
最後に、多忙な中、年報の執筆にに応じていただいた方々にお礼を申し上げます。

総合情報基盤センターの概要

総合情報基盤センター 神鳥成弘

香川大学総合情報基盤センターは、平成 15 年 10 月、旧香川大学と旧香川医科大学の統合に伴い設立されました。設立当時、両大学の協議の結果、当センターの将来構想等を考慮し、本部は医学部キャンパスに、幸町キャンパス（教育・経済・法学部）、工学部キャンパス、農学部キャンパスに、それぞれ分室を設置しています。教授 3、助教授 4、助手 1、技術職員 3 の教職員 11 名が配置され（現員は、教授 2、助教授 5、助手 1、技術職員 3）、各キャンパスに分かれて（医学部キャンパス 6、幸町キャンパス 3、工学部キャンパス 2）、センターの業務、およびそれぞれの専門分野での教育・研究活動を行っています。ここでは、当センターの概要について簡単に紹介します。

キャンパス間ネットワーク： 香川大学は、高松市幸町、林町、高松市に隣接する木田郡三木町にわたって、おもに 4 つのキャンパスを有します。それぞれの位置関係を下に示します。どのキャンパス間も「ちょっと歩いて、そこまで・・・」という距離ではありません。



キャンパス間は ATM（Asynchronous Transfer Mode）専用回線で相互接続されています。各キャンパスでは、ATM スイッチまたは ATM ルータで接続され、そこから教育研究系 LAN(Local Area Network)や事務系 LAN に接続されています。



幸町分室： 幸町キャンパスは高松市街地の西に位置しており、教育・経済・法学部等の他、香川大学本部もここにあり。幸町分室は、経済学部構内の奥まったところであり茶色の煉瓦タイル模様の建物です。



SINET（Science Information Network、日本最大の学術用ネットワーク）には、幸町分室よりファイアウォールを介して接続しています。香川大学は、SINET のノード校で、幸町分室にある SINET ノードから、高松大学、徳島文理大学香川校、詫間電波工業高等

専門学校，香川短期大学，香川県産業技術センターへと接続されています。幸町分室には，香川大学全体のネットワーク基幹設備の他，スーパーコンピューターSX-6i やポスター作成用の大型プリンター等を設置し，学内の研究活動に利用されています。また，幸町キャンパスには 237 台の教育支援用パソコンが設置されており，情報教育（講義・実習）に有効に使用されています。これらのパソコンは，授業のない時間は，学部学生・大学院生に開放されており（一部については 24 時間），学生のレポート作成等，広く自習に活用されています。

医学部キャンパス(本部): 医学部キャンパスは，高松市街地からおよそ 10km 東南，自動車で 30 分程度のところに位置します(木田郡三木町)。医学部キャンパスでは，その性格上，インターネットの接続口等，いくつかのポイントにファイアウォールを設置して，厳しくアクセスの制限を行っています。医学部キャンパスにも，128 台の教育支援用パソコンが設置されており，医学部生に対する情報教育(授業・実習)に使用されるほか，CBT 共用試験(臨床実習開始前に学生の知識・技能を評価する全国共用試験)に対応できるシステム構成となっています。写真(下)は医学部マルチメディア実習室です。実習を効率よく行うため，使用者の間に，中間モニターを設けた仕様になっています。



工学部・農学部分室: 工学部キャンパスは市街地からおよそ 6km 南の香川インテリジェントパークと呼ばれる研究開発機関や産業支援機関が集まっている中にあり，農学部キャンパスは，医学部キャンパスと同じく木田郡三木町にあります。両キャンパスでは，キャンパス内ネットワークを管理する組織を設置していただき，総合情報基盤センターと連携して，キャンパス内ネットワークを分散運用しています。写真(下)は，工学部情報基盤室で管理していただいているサーバー室の様子です。



両キャンパスにも，教育支援用パソコンが，それぞれ 89 台(工学部)，59 台(農学部)設置されており，情報教育に活用されています。

最後になりましたが，当センターの運営には，学内外の皆様のご協力・ご支援が欠かせません。今後とも，どうぞよろしくお願い申し上げます。

著者寸描

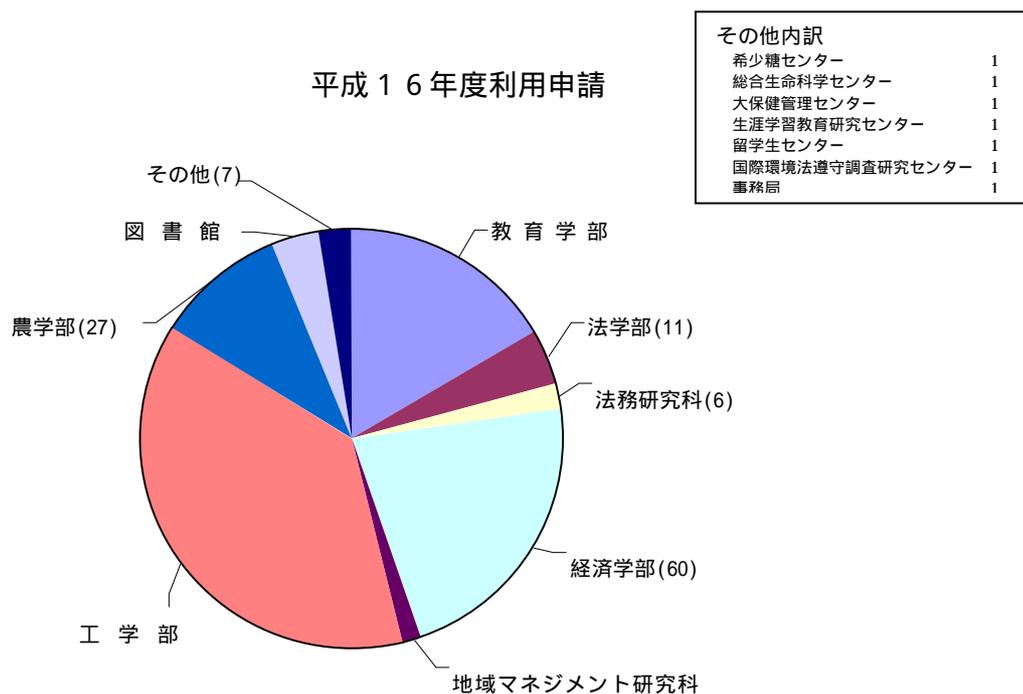
氏名：かみとりしげひろ
所属・職：総合情報基盤センター・教授
教育研究分野：生物物理化学，
構造生物学，生体構造情報科学
座右の銘：ケ・セラ・セラ
趣味：高松に来て 1 年余り。
週末，紫雲山を歩き回っています。



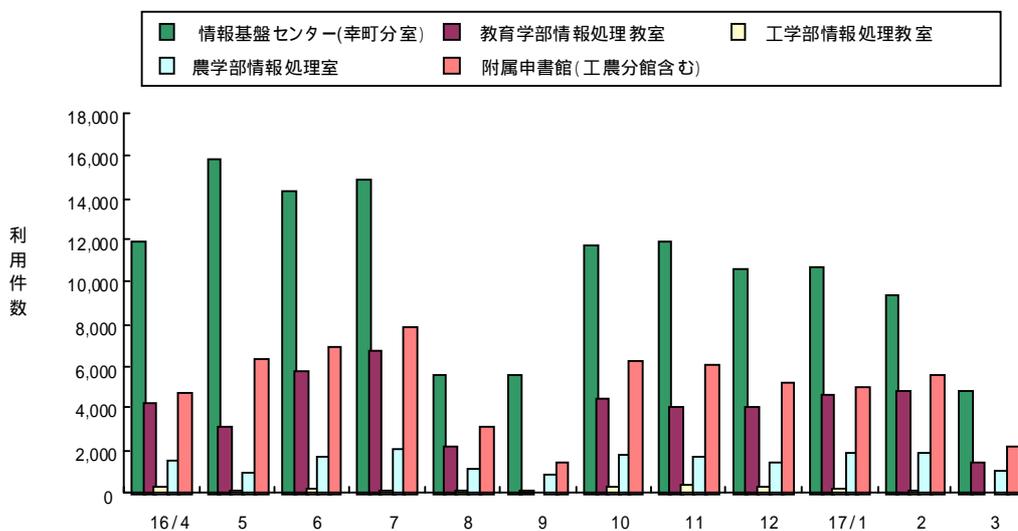
センター利用状況

総合情報基盤センター 瀬野芳孝

平成16年度の計算機利用統計以下に簡単にまとめます。より詳細なデータは巻末に掲載しています。



平成16年度パソコン室利用状況



平成16年度パソコン室利用統計

総合情報基盤センター幸町分室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	272	362	231	384	489	1,086	345	334	258	421	436	209	4,827
法学部	2,901	2,949	3,138	2,905	1,454	1,241	3,136	2,805	2,524	2,578	2,158	1,435	29,224
経済学部	7,644	11,979	10,277	10,618	2,697	2,345	7,493	7,932	7,262	7,219	6,124	2,762	84,352
工学部	126	85	166	266	155	167	169	167	141	110	199	159	1,910
農学部	335	140	309	385	522	317	322	258	179	153	184	195	3,299
その他	671	290	229	296	286	353	303	421	233	244	236	108	3,670
計	11,949	15,805	14,350	14,854	5,603	5,509	11,768	11,917	10,597	10,725	9,337	4,868	127,282

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	12,309	22,961	13,904	30,052	39,964	85,977	18,572	23,154	18,573	25,973	33,075	16,406	340,920
法学部	135,466	175,333	192,791	194,277	110,998	98,160	161,960	186,316	167,268	201,910	180,496	107,932	1,912,907
経済学部	366,367	902,891	715,009	755,341	196,419	175,762	395,971	513,252	509,981	538,629	444,458	200,499	5,714,579
工学部	6,688	4,234	9,931	19,121	11,903	14,300	10,596	11,138	9,340	8,422	17,249	11,369	134,291
農学部	18,252	6,740	23,071	32,118	55,526	24,827	20,367	21,475	17,008	13,046	16,073	14,679	263,182
その他	11,473	5,557	13,274	20,576	12,536	12,076	18,584	19,076	18,128	17,185	20,367	10,022	178,854
計	550,555	1,117,716	967,980	1,051,485	427,346	411,102	626,050	774,411	740,298	805,165	711,718	360,907	8,544,733

教育学部情報処理教室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	3,323	2,158	4,388	4,827	1,569	37	3,422	3,088	2,895	3,481	3,622	1,055	33,865
法学部	21	19	351	540	76	0	320	346	389	371	335	84	2,852
経済学部	163	859	536	681	181	0	378	419	488	442	528	129	4,804
工学部	373	46	245	278	86	0	84	94	79	96	189	101	1,671
農学部	119	37	107	177	116	0	56	41	37	50	66	36	842
その他	66	13	131	215	124	65	138	136	220	227	142	19	1,496
計	4,065	3,132	5,758	6,718	2,152	102	4,398	4,124	4,108	4,667	4,882	1,424	45,530

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	154,889	146,674	321,951	364,703	135,379	1,687	208,947	209,009	211,464	280,162	310,320	78,528	2,423,713
法学部	11,510	1,423	27,252	42,864	7,788	0	25,468	26,398	33,441	34,275	34,343	7,518	252,280
経済学部	16,694	70,442	40,963	63,643	16,029	0	22,534	34,849	44,090	44,451	55,932	10,469	420,096
工学部	7,850	2,002	15,956	19,122	5,752	0	3,800	6,529	5,484	6,901	16,308	6,023	95,727
農学部	4,004	1,770	6,679	12,195	13,202	0	3,175	1,963	3,449	5,278	6,602	2,968	61,285
その他	4,231	874	8,353	20,989	4,260	529	11,640	10,387	21,183	20,122	12,590	1,845	117,003
計	199,178	223,185	421,154	523,516	182,410	2,216	275,564	289,135	319,111	391,189	436,095	107,351	3,370,104

工学部情報処理教室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	0	0	0	0	0	0	0	0	0	0	0	0	0
法学部	0	0	0	0	0	0	0	0	0	0	0	0	0
経済学部	0	0	1	0	0	0	0	0	0	0	0	0	1
工学部	85	67	141	82	14	15	212	259	311	134	86	1	1,407
農学部	0	0	0	0	0	0	0	0	0	0	0	0	0
その他	0	162	2	1	78	0	91	88	13	7	1	17	460
計	85	229	144	83	92	15	303	347	324	141	87	18	1,868

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	0	0	0	0	0	0	0	0	0	0	0	0	0
法学部	0	0	0	0	0	0	0	0	0	0	0	0	0
経済学部	0	0	27	0	0	0	0	0	0	0	0	0	27
工学部	7,088	1,341	7,564	1,226	759	826	28,282	25,651	29,244	12,938	3,743	1	118,663
農学部	0	0	0	0	0	0	0	0	0	0	0	0	0
その他	0	143	195	51	73	0	1,297	983	870	704	58	2,847	7,221
計	7,088	1,484	7,786	1,277	832	826	29,579	26,634	30,114	13,642	3,801	2,848	125,911

農学部情報処理室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	3	0	0	0	1	1	5	7	7	9	4	3	40
法学部	6	0	9	0	3	4	7	4	3	1	8	12	57
経済学部	0	4	1	2	0	0	4	12	8	11	2	2	46
工学部	1	1	5	1	2	0	1	0	2	2	0	0	15
農学部	1,364	828	1,608	1,958	1,044	830	1,637	1,532	1,410	1,825	1,790	947	16,773
その他	135	65	58	68	67	35	70	97	26	64	57	35	777
計	1,509	898	1,681	2,029	1,117	870	1,724	1,652	1,456	1,912	1,861	999	17,708

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	112	0	0	0	159	129	531	434	283	596	406	35	2,685
法学部	262	0	221	0	349	111	276	121	266	82	1,588	470	3,746
経済学部	0	568	117	94	0	0	58	700	680	783	59	268	3,327
工学部	108	108	396	63	106	0	3	0	7	30	0	0	821
農学部	63,200	54,661	91,379	138,537	100,788	49,120	82,562	98,455	99,701	143,683	193,768	74,974	1,190,828
その他	1,566	942	1,455	7,140	1,539	587	2,360	2,426	1,475	6,150	4,739	1,758	32,137
計	65,248	56,279	93,568	145,834	102,941	49,947	85,790	102,136	102,412	151,324	200,560	77,505	1,233,544

附属図書館(工学部及び農学部分館含む)

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	505	674	807	954	445	198	711	718	510	657	657	163	6,999
法学部	587	858	976	1,004	378	201	874	806	705	802	1,007	647	8,845
経済学部	1,289	2,483	1,889	2,137	633	310	1,541	1,746	1,410	1,431	1,450	383	16,702
工学部	844	736	1,201	1,521	393	113	1,371	1,365	1,218	897	965	190	10,814
農学部	1,078	1,421	1,536	1,683	912	443	1,237	1,039	869	850	1,122	642	12,832
その他	375	217	499	564	373	121	502	406	418	375	357	168	4,375
計	4,678	6,389	6,908	7,863	3,134	1,386	6,236	6,080	5,130	5,012	5,558	2,193	60,567

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	15,748	31,816	40,435	49,340	27,766	12,760	33,603	37,468	27,587	36,360	37,284	8,053	358,220
法学部	17,762	40,573	43,860	42,283	18,523	9,270	34,731	40,042	34,130	44,197	49,198	29,889	404,458
経済学部	45,999	129,155	96,878	106,443	31,087	17,442	71,348	89,403	82,122	88,468	68,295	16,394	843,034
工学部	25,905	26,969	44,226	56,863	16,753	5,357	41,175	40,267	36,663	28,202	35,842	7,754	365,976
農学部	44,924	60,065	69,040	76,414	47,178	21,138	48,593	44,741	38,237	41,574	51,234	37,614	580,752
その他	17,041	15,076	30,797	34,566	16,665	10,245	24,533	28,004	31,137	25,313	23,562	13,192	270,131
計	167,379	303,654	325,236	365,909	157,972	76,212	253,983	279,925	249,876	264,114	265,415	112,896	2,822,571

著者点描

氏名：せの よしたか

所属・職：総合情報基盤センター 技術職員

座右の銘：人間万事塞翁馬、そこそこに

趣味：旅行

P2P ソフトウェアについて

総合情報基盤センター 林 敏浩

はじめに

最近、「P2P ソフトウェアが入った PC にウイルス感染して個人情報インターネットに流出した」、「P2P ソフトウェアで著作権保護された映画・音楽データが不正に交換されている」といったニュースが新聞やテレビでよく報道されています。それにより、P2P ソフトウェアの知名度は高くなったと思います。

しかし、P2P ソフトウェアとはどういうものであるのかどの程度きちんと理解されているのでしょうか。また、P2P ソフトウェアの利用の何が問題なのか利用者に正しく理解されているのでしょうか。現在、総合情報基盤センターでは、学内の P2P トラフィックを監視・調査していますが、これらを理解していない利用者が多いことがわかってきました。

本稿では、P2P ソフトウェアに関する正しい理解のために、P2P ソフトウェアとは何なのか、P2P ソフトウェアはなぜ利用制限されるのか、さらに、本センターの P2P トラフィック監視の取り組みについて説明します。

P2P ソフトウェアとは

P2P は Peer to Peer (ピアツーピア) の略で、「個々の間で直接情報などのやり取りを行なう形態」を意味します。この方式でデータ通信を行うのが P2P ソフトウェアです。

これに対し、何らかのサービスを提供するシステム (サーバ) があり、そのサービスを受けるシステム (クライアント) を利用する形態があります。これはサーバ・クライアント (以下、S/C と略記) 方式と呼ばれます。図 1 は P2P と S/C の通信方式を示しています。

電子メールは S/C 方式です。A さんから B さんにパソコンで電子メールを送るには、まず、A さんは自分の PC にあるメーラー (クライアント) でメールサーバにメールを送ります。次に、B さんは自分の PC にあるメーラーを使ってメールサーバからメールをパソコン取り込む形態になります。もし、A さんのメーラーと B さんのメーラーの間で直接メールを送受信できれば、2 人の使っているメーラーは P2P ソフトウェアということになります。

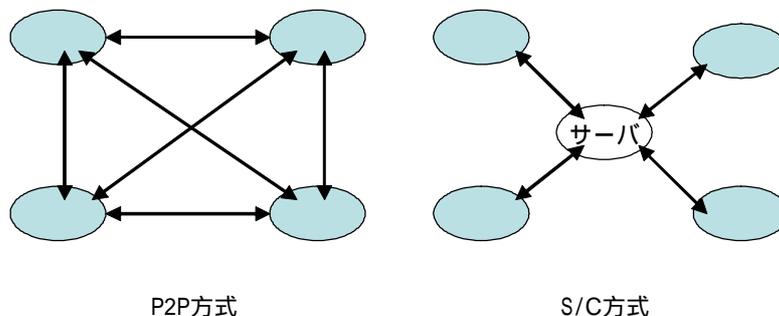


図 1: P2P と S/C の通信方式

このように「P2P」はどの通信方式かを示しているだけです。そのため、P2P ソフトウェアに属するものはたくさんあり、その全てに問題あるわけではありません。ここらへんがP2Pソフトウェアの取り扱いを複雑にしていたりするのですが、次にP2Pソフトウェアの問題点を見ていきたいと思います。

P2P ソフトウェアの利用はなぜいけないのか

まず、P2P ソフトウェアの問題点の一つとして、ユーザ数が増えるとネットワークが混雑するという特徴が挙げられます。図1のP2P方式の例は4ユーザですが、これが100、1000、1万となったら通信（図1では矢印）がどのように増えるか（つまりネットワークが混雑するか）理解できると思います。P2Pソフトウェア利用の増加で他の必要なネットワーク利用に大きな影響を与えてしまいます。

次に、サーバを介さないP2Pの通信方式は、流れているデータのチェックが困難であることを意味します。性善説に立てば、流れるデータをチェックする必要はないということになりますが、残念ながら、現実はそうではありません。著作権を侵害する違法なデータがP2Pソフトウェアを利用して流通しています。

さらにそういったP2Pソフトウェアに感染するウィルスが作成されています。このウィルスの感染によって様々なデータ（個人情報を含む）が一瞬にしてインターネット上にばらまかれたりします。これで職場を辞めるこ

とになった人もいるそうです。

P2P ソフトウェアの利用が制限されるのはこのような理由があるからです。特にファイル交換を目的とするP2Pソフトウェアは著作権侵害と大きく関わります。もちろん、P2Pソフトウェアで、OSなどシステムのアップデートデータを効率的に配信できるというケースもありますが、S/C方式でも同様なことができますので、P2Pソフトウェアの利用を強く擁護する根拠にはなりにくいと考えます。

総合情報基盤センターの取り組み

香川大学でもP2Pソフトウェアによる学内ネットワークの帯域圧迫が確認されるようになってきました。さらに、その利用状況を調査すると著作権侵害に当たるデータの交換があったこともわかりました。

そのために、平成17年8月に総合情報基盤センター長名で「P2Pソフトウェアのトラフィック監視について」の通達を出しました。現在は、それに基づいてトラフィック監視システムを用い、P2Pトラフィックのログ情報を収集しています。また、ログ情報の分析により不審なP2Pトラフィックがあれば該当部局で調査いただいています。場合によって各部局の規程により罰則などの適用もあります。

最後になりましたが、これまで述べてきたP2Pソフトウェアの問題を検討して、香川大学では、原則としてP2Pソフトウェアの利用を禁止しておりますので、ご注意ください。

著者寸描

氏名： はやし としひろ
所属・職： 総合情報基盤センター・助教授
教育研究分野： 教育工学
座右の銘： 未定
趣味： 阿波踊り、柔道、酒を嗜むこと

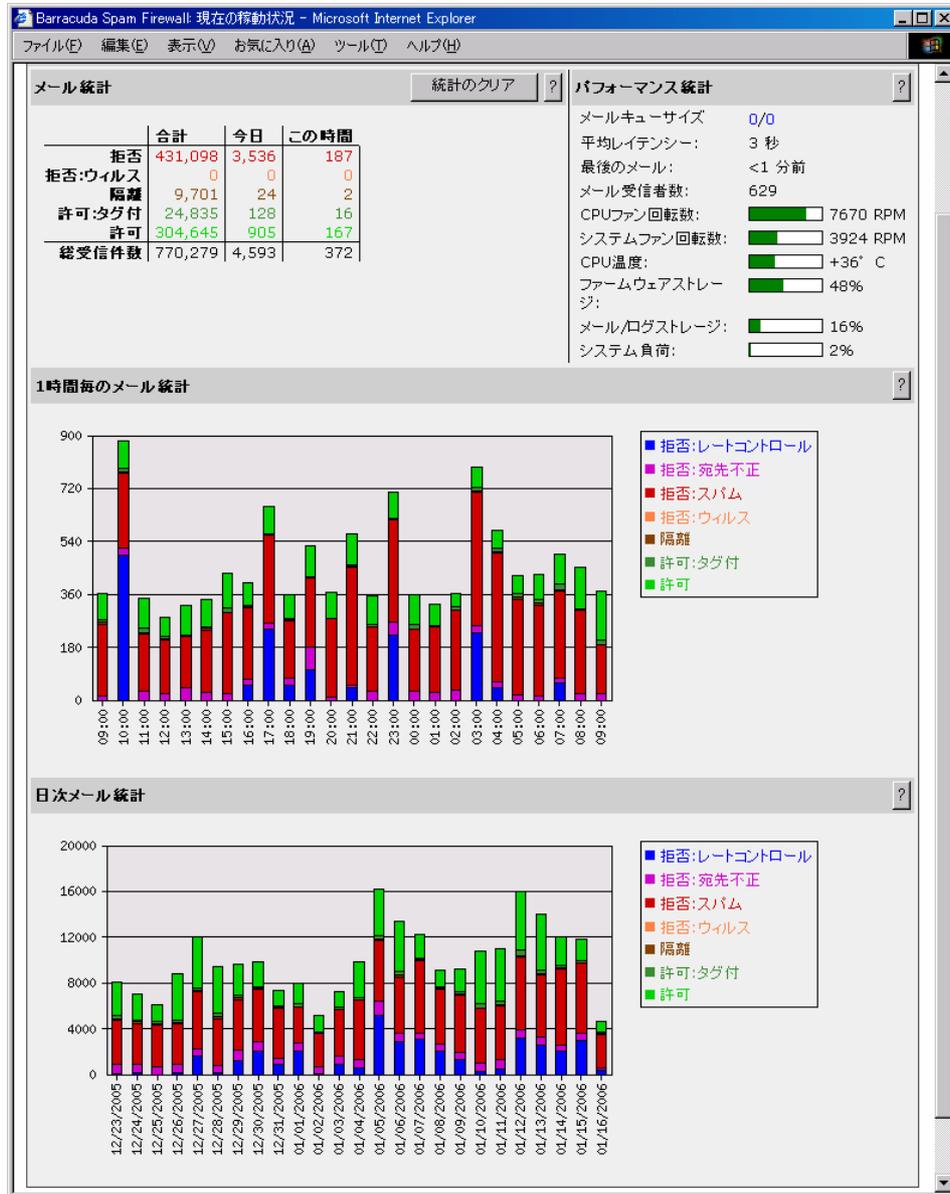


スパムメール対策について

総合情報基盤センター 河内一芳
川田延枝
池田裕子

まず下のグラフをご覧ください。
昨年 11 月中頃から今年の 1 月中頃までに医学部に届いたメールの内訳です。
棒グラフの赤い部分がスパムと判断された

メールで、緑の部分がスパムでないとは判断されてユーザに届いたメールです。
表の合計を見ると送受信件数 77 万通のうちスパムでないとは判断されたのは約 30 万通



だけです。残りはスパムメール、あるいはスパムかもしれないと判断されたメールで、その割合は実に6割以上になります。

では、どのような仕組み（装置）でフィルタリングしているのか簡単に説明します。

医学部に設置しているのは Barracuda Networks Inc. 製のスパムファイアウォールです。

その機能は、まずスパムを送信しているメールサーバのブラックリストがあり、これと照らし合わせてフィルタリングをします。またメールの本文に書かれている Web サイトについても同様のブラックリストがあり、このサイトが含まれているメールもスパムと見なします。これらのブラックリストは自動的に更新されています。

また装置自体に特定の E メールアドレスやドメイン、IP アドレスをブラックリストとして登録し、これらからのメールを拒否することも出来ますし、反対にホワイトリストとして登録し、ここからのメールは無条件に通すことも可能です。このリストは Web ページ上でユーザが独自に登録することが出来ます。

またメールのタイトルや本文に、管理者が設定した特定のキーワードに合致する単語やフレーズが含まれていた場合は拒否したり、隔離したりすることもできます。

次にメールの内容をスキャンしてスパムである確率をスコアとして計算します。その計算方法は、送信元のアドレスの真偽を確認したり、メールの送信プロトコルに違反していないかなどメッセージの信頼性をチェックしたり、またメールの本文からその目的を判断し販売目的だったりスパムとしての特徴を持っていないかなどをチェックしたりして、内

容に応じて加点されます。

スコアは0から9までの10段階で、例えばスコアが2以下ならスパムでないと判断してそのまま送信し、5以下なら「タグをつけて配信」、8以下なら「隔離」、9なら「拒否」といった具合に処理されます。

もし隔離された場合はその通知がユーザ（受信者）に届き、ユーザの判断で再配信（受信）することができます。この作業も Web ページ上で出来ます。「拒否」になったメールも管理者に問い合わせをすれば再配信することが可能です。

これら多数の技術を組み合わせてチェックしていますが、それでもスパムメールがユーザに届いてしまう場合があります。この時はユーザがブラックリストに登録します。逆にスパムでないのに隔離された場合はホワイトリストに登録します。

またそれとは別に、スパムか非スパムかの判断をユーザ自身にさせていただき、その結果を以降の解析に反映させることもできます。

またレートコントロールという機能もあり、一定時間内に特定のアドレスから一定以上の数のメールが送られてきた場合、これを拒否することも可能です。前のページの棒グラフでは青色の部分です。多いときには20分間に3,000通も送られてきたことがありましたが、メールサーバに届く前に未然に防いでいます。

以上のようにたくさんの機能が備わった装置ではありますが、ユーザの支援を得てますます賢くなり効果も上がりますので、医学部の皆様、今後とも協力下さいますよう、よろしく願いいたします。

スパムのない快適なメール生活がおくれますように。

氏 名： かわうち かずよし
所属・職： 総合情報基盤センター ・ 技術職員
座右の銘： たなぼた、漁夫の利、果報は寝て待て、など
趣 味： 猫をてがうこと



氏 名： かわだ のぶえ
所属・職： 総合情報基盤センター ・ 技能補佐員
座右の銘： 笑う門には福来たる
趣 味： お菓子作り&試食
現在のマイブームは“岩盤浴”



氏 名： いけだ ゆうこ
所属・職： 総合情報基盤センター ・ 事務補佐員
座右の銘： らしく生きる
趣 味： 映画鑑賞、Wachi Field(キャラクター)グッズの収集



医学部情報教育の感想・問題点

医学部・基礎放射線学 寺中正人

基礎放射線学の寺中です。今年度は、情報科学実習の一番最初の部分を担当させて頂きました。これについて少し書かせて頂きます。

以前より、我々の教室（旧物理学）の方で医学科の情報科学のコーディネーターをしていた関係から、情報科学に関わって、10年目になります。この10年で、インターネットを中心とする情報基盤が、大きく様変わりしたと同時に、学生にも大きな変化が見られます。

その中で、統合以前は、情報科学、統合以後は情報科学実習と題して情報リテラシーについて教えていますが、現実にはPCのハード、ソフトの使用法を教えることがほとんどの割合を占めています。

10年前といえば、ようやくインターネットが広がり始めた頃であり、授業で用いるPCはNECの9800シリーズ、媒体はFDD、通信デバイスはモデム、DOSからWINDOWSへの移行期、教育、医学の分野ではMACも大きな比重を占めるといった頃です。当然学生の方も、入学以前にPCに触れたことがあるのは一部であり、フロッピーの挿入ができなくて（逆に差し込んだりして）泣きついてくる、といったケースも多く見られました。

授業の方も当時は高松大学の非常勤講師の林先生にお願いし、私が担当した1年目はまだN88BASICについて教えているような状態でした。ただし、一部の学生はプ

ログラムの経験をもち、インターネット以前のローカルネット通信などを行っている学生もあり、極端な能力の分散が見られました。

それに比べると、今年度の学生の場合、ワープロソフトやブラウザは、入学時にほぼ全員が使用でき、ADSL回線が一般的で、OSはほぼWINDOWSXP、となります。（さらに来年度からは、小中高で情報教育を受けた学生が入学してきます。）

しかしながら、学生の情報処理能力が、劇的に進歩しているかという点、一部を除いてそうでもないようです。PCの一般的な使用が、1)ブラウザ、2)ワープロ、3)メール、4)ゲーム、5)ブログ、BBS、辺りに特化している、せいぜい表計算ソフトが一部使える、というのが入学時の学生の一般的なレベルです。

以前では一部見られた、プログラミングができた、自分でPCを組み立てたり、音楽ソフトや画像ソフトを使ったり、サーバを立てたりしているような学生はむしろ減っているように思われました。

また、特徴的なのは、情報の伝達のシステムです。

現在の大学生の標準的な情報デバイスは携帯電話であり、情報伝達手段はメールとなっています。学生によっては、インターネットによる検索辺りまでとなります。

その結果、要求された情報を探して呈示してくるまでの時間ははるかに短くなりま

した。

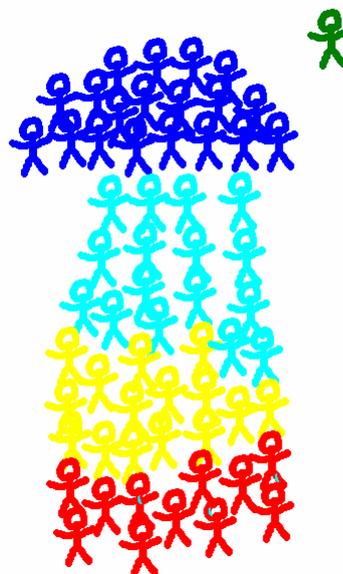
ところが、そうやって得られる情報は断片的で、校正を受けていないこともあって偏ったものが見られ、情報の総量は多いながら、情報の関連づけや深みに乏しいように思われます。また社会的な権利意識に乏しく、参照したURLを明記してやることはめったにありません。

これについては、情報科学より、物理学の課題の場合さらに顕著です。一次文献に当たるといふ習慣が減ってきており、見つけた内容をペーストしてくるだけなので、理解も不十分な場合がほとんどです。レポート、というハードルが、学生にたいしてうまく機能しなくなっていることが伺われます。

興味深い点として、学生個々の処理能力と別に、『群体としての学生』、といったものが出現しています。医学科の学生の場合、個々の総合的な処理能力は比較的高いのですが、授業で特定の課題を出した場合、それに自力で回答してくる学生は一部です。より多くの学生はそれに付随して、他の学生の回答をそのまま参照して提出してきます。これは当然コピーに伴う劣化を加えながら、時間とともに末端の学生まで行き渡ることになります。特に、難しい問題に対して、学生の回答が間違いも含めて大きく数パターンに分類され、交友関係や回答までの時間も含めて系列化できること、情報伝達を遮断した試験などの場合に比べて驚くほど回答パターンが少ないこと、から結構わかるものです。私語するわけではなく、携帯メールやメーラによって、学生の間で

静かに『情報』が広がっていく様子が伺われ、まさに情報化社会だなあと感動することもあるのですが、同時に学生の将来について危惧を覚えます。

こういった傾向に対して、リテラシー提供者として何をなしうるかというところ、結局のところモチベーションを高める手法を模索するしかないと思っています。



(学生のクラゲ化)

中でも学生がどの程度理解しているかを把握し、それに応じてさらに理解することに喜びを感じさせること、を最も重視しています。教えるべき内容の質、量というものには当然要求水準が存在し、ボーダーを落としてより簡単な内容を教えていくのでは、大学という教育機関にふさわしくありません。ただ、同じ内容を教えるにしても、より理解しやすいように教える手法というものがあるのではないかと思いますし、高次の情報化というものは、結局はそういった手法の追求を通じて得られるのではないかと思うのです。

経済学実験の試み

大学院地域マネジメント研究科 曾道智・宍戸榮徳

実験という手法は主として理工系の分野で使われてきましたが、現在では従来文系と思われている経済学にも導入されています。特に、2002年のノーベル経済学賞が、実験経済学の基礎を築きあげた Vernon Smith (George Mason 大学)と Daniel Kahneman (Princeton 大学)に授与されたことにより、実験経済学に対する関心が高まり、現在主流の学会誌では実験経済学に関する研究論文の掲載も増えています。

実験経済学とは、人工的に作り上げた実験室内に検証したい経済理論に必要な環境を設計し、学生などを被験者として、その環境内での行動を調べるものです。実験により、経済理論の予想を検証することができます。そして、従来の理論では予想されなかった部分を調べることもできます。実験の補助手段としてコンピューターやネットワークが多く使われています。そのために専用の経済学実験室を備えるには莫大な資金が必要です。欧米の大学は実験室を備えているところが多くありますが、日本国内にはまだ少ないです。実験手法が経済学の重要な部分になったにもかかわらず、日本で実験経済学を体験できる学生がまだ多くありません。

我々は近年紛争解決法としての仲裁を研究し、2003年に新しい仲裁手順 Amended Final-Offer Arbitration (AFOA)を提案しました。これは、野球選手の年俵に使われてきた Final-Offer Arbitration(FOA)を改善したものです。理論上では、AFOA が FOA より

優れた性質を持っています。しかし、理論だけの研究は実際の仲裁者を説得するのは困難です。仲裁者の不安を取り除くため、米国 Arkansas 大学の Farmer 教授、Deck 博士と組み、実験手法による検証を行うことにしました。それと同時に、香川大学の学生にも実験経済学を体験させたいと思いました。

前述のように、経済実験は多数のコンピューターを使います。幸い、香川大学には総合情報基盤センターがあり、数多くのコンピューターが備えられています。総合情報基盤センターに協力を依頼したところ、ご快諾いただきました。コンフリクトは利害関係者の文化的背景にも関連すると考えられます。今回は一部の実験において、米国 Arkansas 大学の学生もインターネットを通して同時に参加することにしました。総合情報基盤センターのご協力により、これも実現することができました。

もちろん、パソコンを備えているだけでは不十分です。たとえば、経済実験には、「情報の秘匿」が求められており、被験者が利得に関して得る情報などは他の被験者に知られてはいけません。そのため、今回は写真のように各受験者を隔離しました。このような方法についてご理解いただいた総合情報基盤センターにお礼を申し上げます。

結果として、5月18日から26日の1週間と10月27、28日に実験を行うことができました。各学部から合計166人の学生が実験に参加し、経済学の面白さを体験でき

たと思います。数人の参加者から激励・感想のメールも受け取りました。一方、166人の中で、遅刻・欠席者は2名だけでした。Arkansas大学の先生は香川大学の学生の「真面目さ」に驚いていました。これを通して、両大学の実質的な交流ができてよかったと思います。

今回多数の実験者の募集に協力していただいた先生方に感謝します。香川大学では初めての実験でしたので、不安を抱える学生が多

かったのですが、その不安をうまく解消していただいた先生方に御礼を申し上げます。なお、今回の実験をバックアップしていただいた大学院地域マネジメント研究科にも感謝します。

最後に、今回の実験を通して、教育と研究の両面で得られたことが多くありました。これからも引き続きこのような教育・研究活動を行いたいと思っています。総合情報基盤センターの方々、今後ともよろしくお願ひします。



著者寸描

氏名： ぜん だおず

所属・職： 大学院地域マネジメント研究科・助教授

教育研究分野： 地域科学、コンフリクト解消

座右の銘： 座右に「銘」がなくてカレンダーだけです。忙しい...

趣味： 旅行。月に開催される星際会議への出席を楽しみにしております。



医学部キャンパスにおける センターの機器を利用した研究活動

総合生命科学実験センター 実験実習機器部門 宮中 宏

タイトルは、研究活動となっていますが、私
の場合は活動支援ということで、機器部門に
整備されている機器の紹介をさせていただきます。
現在、医学部の実験実習機器部門には、
4年前に情報基盤センター（旧香川医科大学
ネットワーク管理室）のご好意により設置さ
れた多数の機器があります。これは、アナロ
グなメディア（印刷物や各種フィルム等）を
すべてデジタル化し、高品位なデータを速や
かに研究発表や、教育に役立てて行こうと設
置されたもので、この4年間に多数の利用が
あり、満足の行く運用ができたと自負してお
ります。

設置機種と簡単な説明を行っておきます。
入力関係機器として、

フィルムスキャナ1（EPSから4×5サイ
ズまでのネガポジ、マウントスライドは50
枚までの連続スキャン対応）

フィルムスキャナ2（レントゲン・CT等
大判フィルム専用）

A3 ノビ対応フラットベットスキャナ

A3 ノビ対応ドキュメントスキャナ

デジタルカメラ、デジタルビデオカメラ
分析・画像処理として、

LAS-1000PLUS（ゲル・フィルム等より
画像取得、解析が行えます。）

ビデオ編集システム（DVD作成、編集も

可能。なお旧システムと接続しており、ベ
ータ、U-Matic、8mmビデオテープ、
VHS(PAL/NTSC/その他フォーマット対
応)、DV(CAM)、MiniDV(CAM)からの編集
も可能です。テープ間のダビングも可能）
出力装置として、

LFR-MarkIII(プレゼンテーションソフト
等から、35mmフィルムへの出力が行えま
す。フィルムおよび現像は利用者負担。)

A3 カラーレーザープリンタ（両面ユニッ
ト、ソーター付き。部単位での印刷ができ
ます）

A3 対応写真タイププリンタ（富士フィル
ム・ピクトロスタットデジタル400）

B0 ノビ対応インクジェットプリンタ
（EPSON MC-10000 最大で、幅44インチ
×長さ15mまで印刷できます。用紙は専用
紙を利用者負担。）

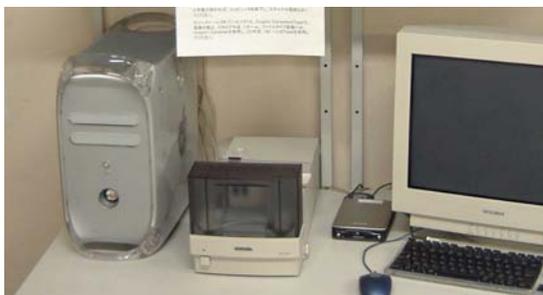
などの機器に加え、研究者に機器部門の機器
を円滑に利用していただく為の機器予約シス
テム。

これらの機器と、機器部門に整備されてい
る多数の機器(詳細については機器部門ホー
ムページ <http://www.med.kagawa-u.ac.jp/~rec/> をご参照ください)との組み合わせに
より、幅広い実験を行うことができます。ま
た取り扱いの難しい機器や、不明なところが

あれば、専任の担当者が分野ごとにいますので、相談いただければサポートいたします。

その他利用方法等の詳細についても、ホー

ムページを参照頂き、センターを有効的に利用し研究に役立てて頂きたいと思います。



Film Scanner 1(35mm - 4×5)



Fuji Film ピクトロスタットデジタル 400
(ネットワークプリント対応)



Film Scanner 2(X-RAY,CT Film)



EPSON MC-10000



A3 ノビフラットベットスキャナ

A3 ノビドキュメントスキャナ



LFR-Mark3(MacOSX, OS9,Windows 各 OS
対応

35mm Film 専用)



Fuji LAS-1000PLUS

工学部の教育用 PC を利用した地域企業への技術教育

工学部 吉村英徳

近年、製品開発における時間的・金銭的コストを削減するため、多数の試作品を製造するのではなく、計算機支援エンジニアリング（Computer Aided Engineering : CAE）技術によるバーチャルでの試行錯誤が主流となっています。自動車部品など多数の部品がプレス加工など塑性加工技術で製造されており、加工工程・金型の設計から、加工条件の最適化、できた製品の強度・性能評価の検討におけるまで、CAE 技術は切り離せないものとなっています。このような成形加工における CAE 技術としては、有限要素法（FEM）と呼ばれるコンピュータシミュレーション解析手法が多く使用されており、現在では多くの有用な市販ソフトが流通して、気軽にシミュレーション（解析計算）ができるようになりました。

さて、最近あちこちで『やり直し材料力学』や『やり直し塑性力学』などの講義が盛んに開催されています。これには大きな 2 個の要因として、(1) 気軽に使えるようになればなるほど市販ソフトがブラックボックス化し、専門知識を学んだ人でなくてもオペレータとして計算を行えるようになったが、うまく計算できない場合に対処が困難になった。(2) シミュレーション結果の妥当性を判断する能力が不足し、計算結果を有効に用いることができない。ということが挙げられるかと思えます。シミュレーションのための条件をデータ入力することによって計算ができるわけですが、製品が複雑化するにつれて、計算のためのモデル化も難しくなり、オペレータには

うまく計算を進めるための高度なノウハウが求められる傾向にあります。また、シミュレーションでは物体の内部の状態まで計算でき、計算終了後、実験結果と形状などの外観を比較するだけではなく、なぜ製品欠陥が発生したのか原因を追究することができますが、解析者にはその能力が強く求められています。これらの能力として習得すべきコンピュータシミュレーション技術は、線形代数学・微分積分学・幾何学・数値解析などの数学的知識、材料力学・弾性力学・塑性力学・材料学などの工学的専門知識の総合的な理論に基づいて構成されております。このような極めて幅の広い知識を身につけるのは大変で、各専門知識の中でシミュレーション技術として特に必要な知識を学びたいという要望から、講義が盛んに行われるようになりました。

香川においては、世界的に優れた技術を有する企業は多くありますが、CAE の利用はまだまだ少ない、また、利用されていても CAE 技術を有効利用されているところは少ないように思われます。CAE に興味は持たれている企業もありますが、わざわざ東京や大阪に向いてまで講義を聞きに行くことは少ないのが現状です。

現在、地方大学の役割として地域貢献も強く求められております。香川大学工学部も平成 9 年に創設されて以来、香川に新産業を起こすため地元企業と産学もしくは産官学にて研究を行ったり、製品開発に関わる技術依頼を受けたりしております。しかし、先端的な

研究開発だけでなく、地元企業のスキルアップも大学の役割であると考えます。私は博士課程の時の所属研究室開催の講習会に始まり、現在は日本塑性加工学会主催の FEM 入門セミナーの講師として、有限要素法の理論を社会人相手に教えてきました。そこで、地元企業のスキルアップを図ろうと志し、昨年秋から工学部にて講義を 2、3 週間に一度開催することに致しました。ある程度のレベルに達するまで同じメンバーで継続的に受講いただいております。顧客対応としてシミュレーション結果の説明をされたり、製品開発に利用されている方 6 名ほどのご参加があって、もう第 7 回目終了しました。是非お役に立てたらと思います。

コンピュータシミュレーションに関する内容ですので、知識をホワイトボードで教えるだけで無く、理解を深めて頂くため工学部にある総合情報基盤センターの分室を借りてパソコンを用いた演習も行っています。図 2 は、表計算ソフト MS-Excel を用いて作成した演習の一例です。これは計算に必要な材料値を求めるためのものですが、有限要素法の計算手順を理解するためのもの、最終的には自作の有限要素法のプログラムを用いた演習も行っていく予定です。

今回の受講者の全講義が終了致しましたら、次回の受講者募集を行いますので、興味ございましたら、是非ご連絡いただきたいと思

氏名： よしむら ひでのり

所属・職： 工学部・助教授

教育研究分野： 材料加工，塑性力学，数値シミュレーション

趣味： 健康のため、ほぼ毎日 5 k m 近く歩いています。

連絡先： TEL&FAX 087-864-2345

E-mail: yosimura@eng.kagawa-u.ac.jp



図 1 演習風景

圧縮 伸び(mm)	高さ	公称ひずみ	公称ひずみ(正解)	対数ひずみ	対数
0	20			0	
1	21			0.05	
2	22			0.1	
3	23			0.15	
4	24			0.2	
5	25			0.25	
6	26			0.3	
7	27			0.35	
8	28			0.4	
9	29			0.45	
10	30			0.5	
11	31			0.55	
12	32			0.6	
13	33			0.65	
14	34			0.7	
15	35			0.75	
16	36			0.8	
17	37			0.85	
18	38			0.9	
19	39			0.95	
20	40			1	

図 1 FEM 解析の入力条件となる材料パラメータを求める演習

います。土曜日に無料で講義を行っておりますし、パソコンも総合情報基盤センターのものが利用できますから、お気軽にご参加ください。



香川大学ウェブページ検索システムの更新

経済学部 中村邦彦

筆者は1998年12月から香川大学内のウェブページ全文検索サービスを提供している。最初は経済学部情報管理学科（現在は廃止）のサーバーで運用していたが、その後2002年から情報処理センター（当時）で購入してもらったサーバーで行ってきた。このサーバーは、CPUがPentium（Coppermine）×2、クロック730MHz、主記憶256Mである。当初は快調であったが、時がたち、収集するページが多くなると、深夜に収集を始めて、夜明けまでにインデックス作成が終わらなくなってきた。それで学内のミラーサーバーや、Java SDKのような大量のドキュメントは収集しないようにしてきたが、それでも時間がかかるようになって来た。最近では文書数25,994、文書量684MBになっていた。さらに多くの文書进行处理するにはこのマシンでは心もとなくなってきたので、研究室のサーバーに引越しすることにした。今の研究室のサーバーはCPUがPentium 4クロック3.40GHz、主記憶1GBである。1秒あたりに処理する文書数では2.3倍、処理文書のバイト数では約10倍の性能が得られた。収集する文書の範囲も拡大したので現在は文書数48,000、文書量1.6GBになっている。

実は今回変更したのはマシンだけではない。これまで、ウェブページの収集にはhtdig[3]を修正したもの、検索エンジンにはNamazu[2]を利用していた。そのあたりは文献[1]に紹介したので繰り返さない。Namazuを取り替えるつもりはなかったが、htdigには不満があった。htdigは収集するページを正規表現で指定することができず、また大きなシステムの一部であるので加工しにくい面があった。例えば、ログツールの一部は、アーカイブを月別に表示する機能があり、また、これらのページがダイナミックに（すなわち要求された時点で）作成されるものがある。もし、ページに前月、次月というようなリンクが存在すると、

収集ロボットは際限なくリンクをたどってこれらのページを収集してしまう。実際、UNIX時間が始まる1970から2038年までのページをすべて収集していた。また、多くのBlogはカレンダーを表示している。これも前月次月のリンクがあると同様に収集されてしまう。こういうものはある程度個別にルールを作り込むしかないと思ったので、収集ロボットを自作することにした。使用言語は手軽に正規表現とハッシュ配列が利用できるPerlにした。まず、類似のものにwwgetallがあるのでこれおを参考にした。主な特徴を紹介する。

wwgetall

- ・吉岡 恒夫(tsuneo@rr.iij4u.or.jp, QWF00133@nifty.ne.jp)作のプリフェッチャー。
- ・Perlで記述されている。
- ・ノンブロッキングモードで動作する。
- ・If-Modified-Sinceヘッダーにより、更新されたページのみ取得することが可能。

これはノンブロッキングモードで動作するので魅力的である。さっそくテストしてみたところ、残念ながら、まれにはあるがIOエラーを起こしてアボートしてしまう。原因を調べてみたがよい対策を思いつかなかったので、結局ノンブロッキングモードで複数のセッションを同時に処理することは断念した。それで、ノンブロッキングモードはタイムアウト処理をするためだけに使うことにしたところ、安定して動作するようになった。これによりウェブページの収集に時間がかかるようになる。特にその時点で何らかの理由で接続できないサーバーがあると、そのサーバーのすべてのページを取得しに行くと、そのたびにタイムアウトするまで待たされることにならないとも限らない。タイムアウトしたサーバーは覚えておいて、もう取りに行かない

ことにした。

新しいシステムに切替えてから間もなく、META 要素で NOFOLLOW を指定したページからリンクされている PDF ファイルがインデックス化されているという指摘があった。Namazu は NOINDEX が指定されているページはインデックス化しないからいいだろうと考えていたが、早速 NOFOLLOW に対応するように変更した。

その他も次々に書き換えたり追加したりしたので、元の作者の吉岡氏には悪いが、ほぼ全面的に書き換えてしまった。今回は使用していないが、収集したページをそのまま表示できるようにスタイルシートや JavaScript のファイルも収集する機能を付け加えている。

収集範囲の指定に正規表現が使えるようになったので、指定がしやすくなった。例えば、香川大学のドメインは、旧香川大学の B クラス1個と、旧医科大学の C クラス32個からなっている。医科大学分はひとつの正規表現で記述できた。

ウェブページを収集する目的は、インデックスを作成するのが目的なので、今は次の拡張子をもつファイルのみを収集している。

html htm txt php pdf xhtml xht doc
xls shtml sht asp

今回新しく収集するようにしたファイルのうち、Word 文書は1871、Excel ワークシートは1325、xhtml は37であった。このうち Excel のファイルは、解析時に頻繁にエラーが出ているので、現在はインデックスを作成していない。他のツールを使うことを検討したい。

1. 著者寸描
2. 氏名： なかむら くにひこ
3. 所属・職： 経済学部 教授
4. 教育研究分野： 計算機科学
5. 趣味；クラシック音楽鑑賞

Word 文書や Excel ワークシートをそのままウェブに掲載するのはどうかと思うが、実際には学内にも結構掲載されているようである。

この他、最近多いのが CGI 形式でダイナミックにページを表示するものである。例えばこういう URI になることが多い。

<http://is.ec.kagawa-u.ac.jp/blog/index.php?itemid=94>

全部集めると意味のないものを集めてくる恐れがあるが、筆者が使用している Blog ツール Nucleus では、ほとんどのページがこの形であり、収集せざるを得ない。今後、収集されている文書を見ながら、ルールを調整していく予定である。

参考文献

- [1] 中村邦彦、『香川大学ウェブページ全文検索システム』、香川大学経済論叢第 73 巻第 3 号、2000 年 12 月
- [2] 『全文検索システム Namazu』、<http://www.namazu.org/>
- [3] 『htdig のホームページ』
<http://www.htdig.org/>
- [4] 『wwgetall のホームページ』
<http://openlab.ring.gr.jp/tsuneo/soft/wwgetall/wwgetall.html>

Cocoon2 での Xindice 利用

- 基本機能の動作検証を中心として -

教育学部 (人間環境教育)

青木 昌三

はじめに

XML をベースとする Web アプリケーションフレームワークである Cocoon2 については、その利用法等について紹介し⁽¹⁾、また、実際にアプリケーションとして「休講通知情報掲示システム」や「学部からのお知らせシステム」を作成し運用してきたところである⁽²⁾。

ところで、Web アプリケーションにおいてはデータベースの効率的な利用が欠かせない。実際に作成・運用しているシステムでは、RDB である MySQL へのアクセスを通して格納された関連データを利用しているが、ネイティブな XML ドキュメントそのものを MySQL のデータとして取り扱っている訳ではない。中心となる休講情報やお知らせ情報そのものは別途 XML ドキュメントとしてファイルに保存し、XML ドキュメントファイルを直接取り扱う Cocoon2 の機能を利用する手法を採っている。ただこうした方法では、1つの XML ファイルに一つ一つの休講情報等を XML 形式に沿って納めていかねばならず、特に情報の修正や削除などを考えると、XML データを取り扱う方法としては今ひとつ汎用性に欠ける面が否めない。それ故、XML ドキュメントそのものをネイティブな形で取り扱う XML データベースを Cocoon2 から利用することが課題となっていた。

Cocoon2 には、ネイティブな XML データベースである Xindice^(*)が組み込まれている。ロジックシートの利用により容易に検索結果を XML 形式で得ることのできる RDB とは違って、サンプルが用意されているとは云えまとまった文献もはなはだ乏しく、Cocoon2 での Xindice 利用については“今から直ぐに”と言う程に敷居が低いことではない。

本稿では、Xindice の基本的な機能の動作検証を通して Cocoon2 での Xindice 利用について報告する。本稿で取り扱っている内容の具体的な項目は次の通りである。

動作検証の実行環境 Standalone モード Xindice の設定

Xindice におけるコレクションの生成と XML ドキュメントの登録

Cocoon2 における Standalone モード Xindice の利用

Cocoon2-Embedded モード Xindice の利用

Jetty Server 上での Cocoon2 における Xindice 登録ドキュメントの修正

(*1)<http://xml.apache.org/xindice>

動作検証の実行環境

本稿で紹介する動作検証については、次の環境のもとで行っている。

OS : Windows2000 (SP4)

Java2SDK : j2sdk1.4.2_03

インストールディレクトリ : c:¥j2sdk1.4.2_03 以後 , %JAVA_HOME%と表す。

Tomcat : jakarta-tomcat-5.0.28

インストールディレクトリ : c:¥Tomcat5.0 以後 , %CATALINA_HOME%と表す。

Cocoon2 : cocoon-2.1.6^(*1)

インストールディレクトリ : %CATALINA_HOME%¥webapps¥cocoon

Xindice : xindice-1.1b4

インストールディレクトリ^(*2) : %CATALINA_HOME%¥webapps¥xindice

%CATALINA_HOME%¥webapps¥xindice¥WEB-INF を %XINDICE_HOME% と表す。

(*1)用いている cocoon.war ファイルは j2sdk1.4.0_03 により作成している。j2sdk1.4.2_03 (1.4.2_09 でも同じ) の下での build war コマンドの実行では警告が多く出され、作成された cocoon.war ファイルではうまく動作しない。

(*2) Cocoon2 とは独立して動作する、いわゆる Standalone モードとしてのインストール先である。

Standalone モード Xindice の設定

Xindice は Apache Software Foundation ^(*1)によるオープンソースのネイティブな XML データベース・ソフトウェアである。上述したように、Cocoon2 には Embedded モードとして既に組み込まれている。以下に示すのは、Cocoon2 と独立してサブレットコンテナである Tomcat のもとで動作する、いわゆる Standalone モードとして動作する場合の Xindice についてである。

(*1)<http://www.apache.org/>

Xindice のインストール

Tomcat 並びに Cocoon2 は既にインストールされているものとする。もし未だの場合には、文献(1)を参照してインストールしてほしい。インストールは次の手順による。

(1) <http://xml.apache.org/xindice> から利用環境に合わせて xml-xindice-1.1b4-war.zip ファイル等をダウンロードする。

(2) xml-xindice-1.1b4-war.zip ファイル等を適当なディレクトリ (フォルダ) 内で展開する。

(3) 得られた xindice-1.1b4 ディレクトリ内の xindice-1.1b4.war ファイルのファイル名を xindice.war に変更し、%CATALINA_HOME%¥webapps 内に配置する。

(4) xindice-1.1b4 ディレクトリ内の xindice-1.1b4.xml ファイルの記述に従って、同ファイルに記述されている <Context> . . . </Context> の部分を、Tomcat のシステムフ

ファイル（%CATALINA_HOME%\conf\server.xml ファイル）にペーストする^(*1)。



図 1 : http://localhost:8080/xindice/へのアクセス画面

以上でインストールは終了である。Tomcat を再起動し，ブラウザで http://localhost:8080/xindice/ にアクセスして図 1，図 2 の 2 つの画面を確認してほしい。図 2 は，図 1 の画面で [db](#) の部分をクリックして得られる画面である。



図 2 : 図 1 の画面で [db](#) の部分をクリックして得られる画面

(*1)xindice.war ファイルを%CATALINA_HOME%\webapps 内ではなく別のディレクトリに配置する場合についての注意が，xindice-1.1b4.xml ファイル内に与えられている。

コマンドラインツール利用のための設定

Xindice に付属するコマンドラインツール xindice.bat（あるいは xindice.sh など）を利用するために環境変数 XINDICE_HOME や PATH を設定する。

(1)環境変数 XINDICE_HOME の設定^(*1)

%CATALINA_HOME%\webapps\xindice\WEB-INF を設定する。

(2)PATH の設定

%CATALINA_HOME%\webapps\xindice\WEB-INF を PATH に加える。

以上で xindice コマンドの利用が可能となる。

(*1)環境変数 XINDICE_HOME に%CATALINA_HOME%\webapps\xindice を設定した場合，xindice コマンドの実行で，NoClassFoundException となる。クラスの検索に%XINDICE_HOME%が起点として用いられるのであるが lib ディレクトリは xindice.bat と同じ WEB-INF ディレクトリ内にある。

コマンドラインツールによる確認

コマンドラインツール `xindice.bat` を用いて Xindice へのアクセスを確認する。コマンドラインツールの利用法については、

<http://xml.apache.org/xindice>

の Documentation の項にある「Xindice 1.1 Administration Guide」や「Xindice 1.1 Commandline Tool Guide」を参照してほしい。尚、以後この Documentation の項を単にドキュメントと呼ぶことにする。

先ず、`xindice` コマンドにより Xindice に登録されているコレクションを表示する。

コマンドは

`xindice lc -c xmldb:xindice://localhost:8080/db`

である。実行の結果は以下の通りとなる(*1)。

```
C:\Documents and Settings\Administrator>xindice lc -c xmldb:xindice://localhost:8080/db
trying to register database

      system
      meta

Total collections: 2
```

これにより、2つのコレクション、`system` と `meta`、が登録されていることが分かる(図 2 をも参照)。

(*1) `xindice lc -c xmldb:xindice://localhost:8080/db` にかえて `xindice lc -c /db` を実行すると

```
C:\Documents and Settings\Administrator>xindice lc -c /db
trying to register database
XMLDB Exception 0: Cannot communicate with the server: http://127.0.0.1:8888/xindice/

Xindice Command Tools v1.1b4

Format: xindice action [switch] [parameter]
===== 以下 コマンドラインツール利用の説明, 略 =====
```

となる。この結果から、`xindice` コマンドのデフォルトでは、`localhost:8888` への接続となっていて、サーブレットエンジン Jetty Server の利用を前提としていることが分かる。

Xindice におけるコレクションの生成と XML ドキュメントの登録

前章で Standalone モードでの Xindice を利用できる環境が整った。ここでは、コレクションを生成して XML ドキュメントを登録する。

コレクションの生成

2つのコレクション、`addressbook` と `kyuukou` をコマンドラインツール `xindice` により生成する。コマンドはそれぞれ、

`xindice ac -c xmldb:xindice://localhost:8080/db -n addressbook`

xindice ac -c xmldb:xindice://localhost:8080/db -n kyuukou

である（以下参照）

```
C:¥MyNote¥Xindice>xindice ac -c xmldb:xindice://localhost:8080/db -n addressbook
trying to register database
Created : xmldb:xindice://localhost:8080/db/addressbook

C:¥MyNote¥Xindice>xindice lc -c xmldb:xindice://localhost:8080/db          確認する
trying to register database

    addressbook
    system
    meta

Total collections: 3
C:¥MyNote¥Xindice>xindice ac -c xmldb:xindice://localhost:8080/db -n kyuukou
trying to register database
Created : xmldb:xindice://localhost:8080/db/kyuukou

C:¥MyNote¥Xindice>xindice lc -c xmldb:xindice://localhost:8080/db          確認する
trying to register database

    addressbook
    system
    meta
    kyuukou

Total collections: 4
```

念のため、ブラウザにより確認する。アクセスする URL は

<http://localhost:8080/xindice/?/db>

である（図3参照）。図2と比較して2つのコレクションが生成されていることが分かる。



図3：コレクション生成の確認画面

XML ドキュメントの登録

コレクションを生成できたので、それぞれのコレクションに XML ドキュメントを登

録する。尚、登録した XML ドキュメントを付録 A に示す。

コレクション addressbook への登録

登録するドキュメントは address1.xml , address2.xml ファイルである。それぞれのキーを address1 , address2 とする。コマンドはそれぞれ

```
xindice ad -c xmldb:xindice://localhost:8080/db/addressbook -f address1.xml -n address1
```

```
xindice ad -c xmldb:xindice://localhost:8080/db/addressbook -f address2.xml -n address2
```

である (以下参照)

```
C:\MyNote\Xindice\examples\xml1>xindice ad -c xmldb:xindice://localhost:8080/db/addressbook -f address1.xml -n address1
trying to register database
Added document xmldb:xindice://localhost:8080/db/addressbook/address1

C:\MyNote\Xindice\examples\xml1>xindice ad -c xmldb:xindice://localhost:8080/db/addressbook -f address2.xml -n address2
trying to register database
Added document xmldb:xindice://localhost:8080/db/addressbook/address2
// 確認する。address1 と address2 が登録されていることが分かる。
C:\MyNote\Xindice\examples\xml1>xindice ld -cxmldb:xindice://localhost:8080/db/addressbook
trying to register database

    address1
    address2

Total documents: 2
```

コレクション kyuukou への登録

登録するドキュメントは kyu1.xml(キー : kyu1) ,kyu2.xml(キー : kyu2) ,kyu3.xml(キー : kyu3) である (以下参照)。

```
C:\MyNote\Xindice\examples\xml2>xindice ad -c xmldb:xindice://localhost:8080/db/kyuukou -f kyu1.xml -n kyu1
trying to register database
Added document xmldb:xindice://localhost:8080/db/kyuukou/kyu1

C:\MyNote\Xindice\examples\xml2>xindice ad -c xmldb:xindice://localhost:8080/db/kyuukou -f kyu2.xml -n kyu2
trying to register database
Added document xmldb:xindice://localhost:8080/db/kyuukou/kyu2

C:\MyNote\Xindice\examples\xml2>xindice ad -c xmldb:xindice://localhost:8080/db/kyuukou -f kyu3.xml -n kyu3
trying to register database
Added document xmldb:xindice://localhost:8080/db/kyuukou/kyu3
```

登録ドキュメントの内容確認

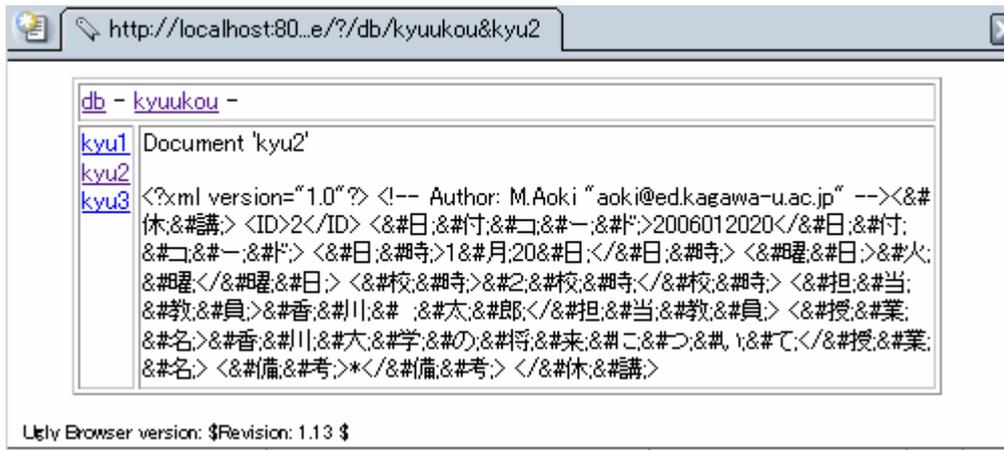
XML ドキュメントの登録ができたので内容を確認しておこう。例として/休講/ID の値が2であるドキュメント (kyu2) を表示する (以下参照)。期待通りにドキュメントが登録され、また検索・表示されることが分かる。付録 A に示した内容と比較してほしい^(*)。尚、用いている XPath による Query については、ドキュメントの「Xindice 1.1 XPath Guide」と文献(3)に詳しく紹介されているので是非一読をお勧めする。

```
C:\MyNote\Xindice\examples\xml2>xindice xpath -c xmldb:xindice://localhost:8080
/db/kyuukou -q "//休講[ID=2]"
trying to register database
<休講 src:col="/db/kyuukou" src:key="kyu2" xmlns:src="http://xml.apache.org/xindice
/Query">
  <ID>2</ID>
  <日付コード>2006012020</日付コード>
  <日時>1 月 20 日</日時>
  <曜日>火曜</曜日>
  <校時>2 校時</校時>
  <担当教員>香川 太郎</担当教員>
  <授業名>香川大学の将来について</授業名>
  <備考>*</備考>
</休講>
```

(*1) ブラウザによる確認では、ドキュメントの登録状況は次のようであったが、



kyu2 の内容表示では以下のように文字化けした。Cocoon2 からの利用でも同じような状況が生じる。パーサの仕様によるものであろう。この点については Cocoon2 からの利用の項でも触れる。



Java プログラムによる登録ドキュメントの内容確認

XML ドキュメントの検索等については、当然のことながら Java プログラムによっても可能であり、プログラミングの方法についてはドキュメントの「Xindice 1.1 Developer Guide」に詳しく説明されている。ここでは、「Xindice 1.1 Developer Guide」での例を参考として Java 用スクリプト言語 Groovy を用いて作成したスクリプトプログラムにより、登録ドキュメントの内容確認を行ってみる。

作成したスクリプトは `xindice_xpath.groovy` で、利用法は次の通りである。

groovy xindice_xpath.groovy collection xpath

collection にはコレクション名を、xpath には検索する XPath を指定する。スクリプトの内容は付録 B に示している。実行の結果を示す（以下参照）。

```
C:\MyNote\nenpo05\xml>groovy xindice_xpath addressbook "//person[fname='Robert']/email"
trying to register database
<email src:col="/db/addressbook" src:key="address2" type="work" xmlns:src="http://xml.apache.org/xindice/Query">robert@mailserv.com</email>
<email src:col="/db/addressbook" src:key="address2" type="home" xmlns:src="http://xml.apache.org/xindice/Query">smith@efg-mail.com</email>

C:\MyNote\nenpo05\xml>groovy xindice_xpath kyuukou "//休講[ID=3]/授業名"
trying to register database
<授業名 src:col="/db/kyuukou" src:key="kyu3" xmlns:src="http://xml.apache.org/xindice/Query">情報学基礎</授業名>
```

尚、Groovy によるスクリプト作成については、文献(4)とそこに示されている参考文献を参照していただきたい。

Cocoon2 における Standalone モード Xindice の利用

Cocoon2 からの Xindice 利用について説明する。Cocoon2 からの利用では、上述してきた Standalone モードの Xindice にアクセスする場合と、予め Cocoon2 に組み込まれている Embedded モードの Xindice にアクセスする場合の 2 通りの利用法がある。Cocoon2 の XML:DB ブロックサンプル（以下では、単にブロックサンプルという。）

に与えられているのが Embedded モードでの利用である。以下では、2つのモードでの利用について順次動作確認を行っていくが、先ず Standalone モードでの利用を採りあげる。

2つの準備

テスト用ディレクトリの作成

Cocoon2 から利用するために、Cocoon2 配下にテスト用のディレクトリ edu/edu20 を作成する(*1)。

(1)%CATALINA_HOME%\webapps\cocoon の下に edu ディレクトリを作成し、更にその内部に edu20 ディレクトリを作成する。

(2)cocoon ディレクトリ内にある sitemap.xmap において edu ディレクトリへのマウントを設定する(以下参照、必要部分のみ呈示)。

```
<map:match pattern="edu/**">
  <map:mount check-reload="yes" src="edu/" uri-prefix="edu"/>
</map:match>
```

(3)edu ディレクトリ内の sitemap.xmap において edu20 ディレクトリへのマウントを設定する(以下参照、必要部分のみ)。edu ディレクトリ内に sitemap.xmap ファイルを未だ作成していない場合には、cocoon\samples\sitemap.xmap などをコピーするなどして作成する。

```
<map:match pattern="edu2*/**">
  <map:mount uri-prefix="edu2{1}" src="edu2{1}/" check-reload="yes"/>
</map:match>
```

(4)edu20 ディレクトリ内に sitemap.xmap ファイルを作成する。

cocoon\samples\blocks\xmlldb\sitemap.xmap

をコピーして用いるとサンプルとなる記述が含まれており簡便である。

(5)edu20 ディレクトリ内に2つのディレクトリ xml1 と xml2 を作成し、そこにテストで用いる XML ドキュメントファイルを置いておく。

xml1 ディレクトリ：コレクション addressbook のための address1.xml など

xml2 ディレクトリ：コレクション kyuukou のための kyu1.xml , kyu2.xml など

(*1)筆者の環境での設定である。cocoon ディレクトリ内に、例えば xmlldb-test ディレクトリを作成し、そこをテストのためのディレクトリとするのも一法である。その場合、cocoon\sitemap.xmap でのマウント記述は、例えば次のようにすればよい。

```
<map:match pattern="xmlldb-test/**">
  <map:mount check-reload="yes" src="xmlldb-test/" uri-prefix="xmlldb-test"/>
</map:match>
```

その上で、サンプルとなる適当な sitemap.xmap ファイルを xmlldb-test ディレクトリ内にコピーして、以後の設定に用いることとなる。読者の環境にあわせて設定してほしい。

cocoon.xconf ファイルでの設定

%CATALINA_HOME%\webapps\cocoon\WEB-INF\cocoon.xconf ファイルを編

集して、以下の修正・追加を行う^(*1)。

(1)既に設定されている xindice-embedded タイプの driver クラスを

org.apache.xindice.client.xmlldb.DatabaseImpl

から

org.apache.xindice.client.xmlldb.embed.DatabaseImpl

に修正する。

(2)次のように 2 つの<!-- 2005.09.17 -->で挟まれた部分を書き加える。

cocoon.xconf の修正・追加箇所：

```
<!--+
      | Xindice 1.1 Embedded driver. See Xindice javadoc for configuration options.
      +-->
<!-- 上記(1)の修正 (太字部分) -->
      <driver class="org.apache.xindice.client.xmlldb.embed.DatabaseImpl" type="xindice-embed">
      = = = = = 中 略 = = = = =
      <managed>true</managed>
      </driver>

      <!-- Add here other XML:DB compliant databases' drivers -->      この行の次に追加
<!-- 2005.09.17 -->      実際の追加記述は次行から
      <driver class="org.apache.xindice.client.xmlldb.DatabaseImpl" type="xindice">
      <configuration>/Tomcat5.0/webapps/xindice/WEB-INF/system.xml</configuration>
      <db-home>/Tomcat5.0/webapps/xindice/WEB-INF/db</db-home>
      <managed>true</managed>
      </driver>
<!-- 2005.09.17 -->      一行上の、</driver>の行までを追加記述する
      </component-instance>
```

以上の cocoon.xconf での追加記述により、以後 Embedded モードでの “ Pseudo Protocol ” である **xmlldb:xindice-embed:**に加えて、Standalone モードでの **xmlldb:xindice:**を用いることができる。具体的な利用法については後述する。

(*1)<http://wiki.apache.org/cocoon/XMLIndiceCocoon2.1> を参照。

Pseudo Protocol を用いての登録ドキュメントの検索

Standalone モードでの Xindice には既に、コレクション addressbook と kyuukou のもとに XML ドキュメントが登録されている。これらの登録ドキュメントについて検索を行ってみる。先に cocoon.xconf ファイルにおいて追加記述した “ Pseudo Protocol ” を利用する。

addressbook での検索

sitemap.xmap ファイルに次の記述を加える(1 行で書くところを便宜上折り返している)。

```
<map:match pattern="xindice-xpath01">
```

```
<map:generate
src="xmldb:xindice://localhost:8080/db/addressbook/#/person[fname='John']"/>
  <map:serialize type="xml"/>
</map:match>
```

ブラウザで <http://localhost:8080/cocoon/edu/edu20/xindice-xpath01> にアクセスして得られる結果を図 4 に示す。

kyuukou での検索

具体的に sitemap.xmap ファイルを編集する前に、文字エンコーディングについて触れておく。sitemap.xmap をはじめとして Cocoon2 で取り扱う全ての XML ファイルのエンコーディングとしては、デフォルトでは UTF-8 を用いることになっている。ところが、これから取り扱う kyuukou コレクション内のドキュメントについて検索するには、漢字を用いて XPath を記述しなければならない。今テストを行っている環境では、OS は Windows2000 であり、また筆者の用いているエディタではいわゆる Shift_JIS のドキュメントのみが編集

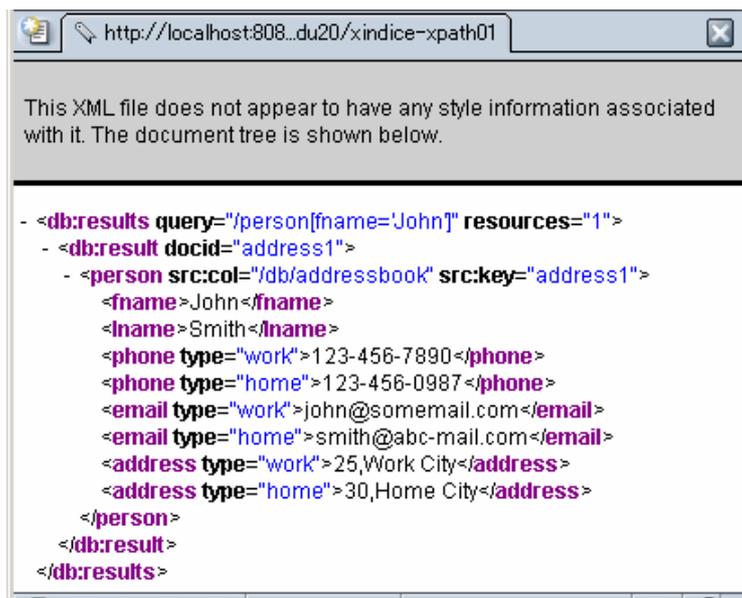


図 4 : addressbook での検索結果

可能である。そのため、Shift_JIS エンコーディングを取り扱う様に記述する。

先ず sitemap.xmap の先頭行を次のように変更する (UTF-8 エンコーディング等の他のエンコーディング環境で編集する場合には適宜その環境に合わせてほしい)。

sitemap.xmap ファイルでの文字エンコーディングの指定 :

```
<?xml version="1.0" encoding="Windows-31J"?>          先頭行をこの内容とする
<!--?xml version="1.0" encoding="UTF-8"?-->          コメント扱いとする
```

次に sitemap.xmap ファイルに次の記述を加える (以下参照)。

```
<map:match pattern="xindice-xpath02">
```

```

<map:generate src="xmldb:xindice://localhost:8080/db/kyuukou/#/休講[ID=3]"/>
<!--map:transform src="kyuukou01.xsl"/-->
<map:serialize type="html"/>
</map:match>

```

上記の XPath から分かるように、ID の値が 3 である休講（kyu3 ドキュメント）を検索している。ブラウザによるアクセスは次の通りとなる（図 5 参照）(*1)。



図 5 : kyuukou コレクション内の kyu3 ドキュメントの検索結果

(*1) 上記の sitemap.xmap ファイルでの記述で <map:serialize type="html"/> に代えて、<map:serialize type="xml"/> とした場合の結果は XML パースエラーとなる。上述した xindice コマンドを用いてのドキュメント内容確認の際に文字化けとなったエラーと恐らく根は同じであろう。



コメント部分を取り除いて

```

<map:transform src="kyuukou01.xsl"/>
<map:serialize type="html"/>

```

とした場合に得られる結果は次の図のようになる。

休 講 通 知						
ID番号	日 時	曜 日	校 時	担当教員	授業名	備 考
3	1月23日	金曜	3校時	情報 次郎	情報学基礎	補講を実施

序なので休講通知全てを表示しておこう。sitemap.xmap に次の記述を加え、

```

<map:match pattern="xindice-xpath03">
  <map:generate src="xmldb:xindice://localhost:8080/db/kyuukou/#/休講"/>
  <map:transform src="kyuukou02.xsl"/>
  <map:serialize type="html"/>
</map:match>

```

xindice-xpath03 にアクセスすると下図に示す結果が得られる。テストでは、休講通知の各 XML ドキュメントは日付コード順に Xindice に登録されているが、実施には登録の順序により日付コードは順不同であろう。下図を得るについては、kyuukou02.xsl において日付コードによりソートを施している。勿論、逆順にも表示が可能である。尚、kyuukou01(02).xsl の内容は誌面の都合上割愛する。

休 講 通 知

日時	曜日	校時	担当教員	授業名	備考
1月19日	月曜	1校時	青木 昌三	物理学概論1	後日補講を実施する
1月20日	火曜	2校時	香川 太郎	香川大学の将来について	*
1月23日	金曜	3校時	情報 次郎	情報学基礎	補講を実施

XMLDBTransformer の利用

Cocoon2 では、予め用意された各種コンポーネントを有効に活用することが肝要である。ここでは、XMLDBTransformer コンポーネントを利用することにより、Xindice へのコレクションの生成、XML ドキュメントの登録、登録済みドキュメントの削除、ドキュメントの修正 (XUpdate) などを行う(*1)。

XMLDBTransformer の確認

%CATALINA_HOME%\webapps\cocoon\sitemap.xmap ファイルに次の記述があることを確認する。コンポーネントを利用する際の type は”xmlldb”である。以下のリスト中太字で表した部分が対象とするデータベースを示していて、この設定では、対象は Embedded モードの Xindice となっている。尚、“Pseudo Protocol”である **xmlldb:xindice-embed:**が利用できるよう、下記のようにコメント内の<driver>・・・</driver>の部分をアンコメントとする(太字部分)(*2)。

```
<map:transformer logger="sitemap.transformer.xmlldb" name="xmlldb" src="org.apache.cocoon.transformation.XMLDBTransformer">
  <!-- Driver class (optional). Uncomment if you want XMLDBTransformer
        to create and register XML:DB database. Keep commented out if
        database created somewhere else (for example, by XML:DB source factory)
  -->
  <driver>org.apache.xindice.client.xmlldb.embed.DatabaseImpl</driver>
  <!-- -->
  <b>base>xmlldb:xindice-embed:///db</base>
</map:transformer>
```

コレクション addressbook2 の生成

コレクションの生成例として、addressbook2 コレクションを生成する。以下の tr-create-coll01.xml ファイルを作成した上で、sitemap.xmap ファイルに追加記述する。XMLDBTransformer 利用に関わる記述において、Standalone モードの Xindice データベースを参照するよう明示的に示している点に注意してほしい(太字の部分、以下同様)。

tr-create-coll01.xml ファイル:

```
<?xml version="1.0" ?>
<page>
  <x:query type="create" oid="addressbook2/"
  xmlns:x="http://apache.org/cocoon/xmlldb/1.0"/>
```

```
</page>
```

sitemap.xmap ファイルへの追加記述：

```
<map:match pattern="tr-create-coll01">
  <map:generate src="tr-create-coll01.xml"/>
  <map:transform type="xmldb">
    <map:parameter name="base" value="xmldb:xindice://localhost:8080/db"/>
  </map:transform>
  <map:serialize type="xml" />
</map:match>
```

tr-create-coll01 へのアクセス結果は図 6 の通りで、コレクション address2 が生成されたことが示されている。

```
- <page>
  <x:query oid="addressbook2" type="create" result="success"/>
</page>
```

図 6：コレクション address2 生成実行画面

当然のことではあるが、再度の tr-create-coll01 へのアクセスでは、

```
<page>
<x:query oid="addressbook2/" type="create" result="failure">Failed to create collection
addressbook2/: 1</x:query>
</page>
```

となる。念のため、ブラウザにより addressbook2 の生成を確認しておく（図 7 参照）。



図 7：コレクション addressbook2 の生成確認

XML ドキュメントの登録

XML ドキュメント登録の例として、付録 A に示した address3.xml と kyu4.xml ファイルを登録する。kyu4.xml の登録は、漢字データを含むドキュメントの登録状況を検証するためのものである。

テスト

XML ドキュメント中に他のドキュメント内容をインクルードする様子を確認する。次の内容を記述した tr-test1.xml ファイルを用意する。

tr-test1.xml ファイル :

```
<?xml version="1.0" ?>
<page>
  <x:query type="create" collection="addressbook" oid="address3" xmlns:x="http://apac
he.org
/cocoon/xmlldb/1.0">
  <i:include src="./xml1/address3.xml" xmlns:i="http://apache.org/cocoon/include/1.0"/>
  </x:query>
</page>
```

次に , sitemap.xmap ファイルに次の記述を加える。

```
<map:match pattern="tr-test1">
  <map:generate src="tr-test1.xml"/>
  <map:transform type="cinclue"/>
  <map:serialize type="xml" />
</map:match>
```

```
- <page>
- <x:query type="create" collection="addressbook" oid="address3">
- <person>
  <fname>Richard</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-1234</phone>
  <phone type="home">456-789-4321</phone>
  <email type="work">richard@mailserv.com</email>
  <email type="home">richard@hij-mail.com</email>
  <address type="work">88,Some City</address>
  <address type="home">55,Abce City</address>
</person>
</x:query>
</page>
```

図 8 : tr-test1 へのアクセス結果

tr-test1 へのアクセスで得られる結果は図 8 の様となる。図 8 から , <x:query> タグと </x:query> タグの間に address3.xml の内容が読み込まれており , XMLDBTransformer によるドキュメント登録の書式に合っていることが分かる。

address3.xml 登録の実際と確認

address3.xml ドキュメントを addressbook コレクションに登録する。sitemap.xmap ファイルに次の記述を加える。

```
<map:match pattern="tr-add-doc01">
  <map:generate src="tr-test1.xml"/>
  <map:transform type="cinclue"/>
  <map:transform type="xmlldb">
    <map:parameter name="base" value="xmlldb:xindice://localhost:8080/db"/>
  </map:transform>
  <map:serialize type="xml" />
</map:match>
```

tr-add-doc01 へアクセスして得られる結果は次の図 9 の様であり , ドキュメントの登録

が成功したことを示している (*3)。

```
- <page>
  <x:query oid="address3" type="create" result="success"/>
</page>
```

図 9 : XMLDBTransformer によるドキュメントの登録

念のため , コレクション addressbook にドキュメントが address3 をキーとして登録されていることを確認しておこう。http://localhost:8080/xindice/?/db/addressbook にアクセスする (図 10 参照)

```
db - addressbook -
address1
address2
address3
```

図 10 : address3.xml 登録の確認

kyu4.xml の登録と確認

sitemap.xml ファイルに次の tr-add-doc02 を追加記述してアクセスする。

```
<map:match pattern="tr-add-doc02">
  <map:generate src="tr-test2.xml"/>
  <map:transform type="include"/>
  <map:transform type="xmlldb">
    <map:parameter name="base" value="xmlldb:xindice://localhost:8080/db"/>
  </map:transform>
  <map:serialize type="xml" />
</map:match>
```

ただし , tr-test2.xml ファイルは , tr-test1.xml ファイルにおいて addressbook を kyuukou に , address3 を kyu4 に , ./xml1/address3.xml を ./xml2/kyu4.xml に置き換えたものである。アクセスの結果は図 9 と同様で , kyu4 が登録されていることも分かる (以下参照)

```
C:\Tomcat5.0\webapps\cocoon\edu\edu20\xml1>xindice ld -c xmlldb:xindice://localhost:8080/db/kyuukou
trying to register database

    kyu1
    kyu2
    kyu3
    kyu4

Total documents: 4
```

登録 XML ドキュメントの削除

例として、先程登録したコレクション addressbook 中の address3 を削除してみる。次の内容の tr-del-doc01.xml ファイルを用意した上で sitemap.xmap ファイルに以下を追加記述する。

tr-del-doc01.xml ファイル：

```
<?xml version="1.0" ?>
<page>
  <x:query type="delete" collection="addressbook" oid="address3" xmlns:x="http://apache.org/cocoon/xmldb/1.0">
    </x:query>
  </page>
```

sitemap.xmap ファイルへの追加記述：

```
<map:match pattern="tr-del-doc01">
  <map:generate src="tr-del-doc01.xml"/>
  <map:transform type="xmldb">
    <map:parameter name="base" value="xmldb:xindice://localhost:8080/db"/>
  </map:transform>
  <map:serialize type="xml" />
</map:match>
```

tr-del-doc01 へのアクセスで得られる結果を図 11 に示す。実際に address3 がコレクション addressbook から削除されていることも、ブラウザによるアクセス、あるいは xindice コマンドによって確かめられる^(*)。

```
- <page>
  <x:query oid="address3" type="delete" result="success"/>
</page>
```

図 11：XMLDBTransformer によるドキュメントの削除

XML ドキュメントの修正

Xindice に登録された XML ドキュメントを修正するためのタイプ “update” が用意されているとのことであるが^(*)、実はどういう訳かうまく動作せず、現時点では解決法を見出し得ていない^(*)。サンプルの xml ファイル等を示す。

先ず tr-update-doc01.xml を作成する（以下参照）。コレクション addressbook 内のドキュメント address3 において、自宅のメールアドレス要素を削除するものである。

tr-update-doc01.xml：

```
<page>
  <x:query type="update" collection="addressbook" oid="address3" xmlns:x="http://apache.org/cocoon/xmldb/1.0">
    <xu:modifications version="1.0" xmlns:xu="http://www.xmldb.org/xupdate">
      <xu:remove select="/person/email[@type = 'home']"/>
    </xu:modifications>
  </x:query>
</page>
```

次に，sitemap.xmap ファイルに以下のコードを追加する。

```
<map:match pattern="tr-update-doc01">
  <map:generate src="tr-update-doc01.xml"/>
  <map:transform type="xmlldb">
    <map:parameter name="base" value="xmlldb:xindice://localhost:8080/db"/>
  </map:transform>
  <map:serialize type="xml" />
</map:match>
```

address3 ドキュメントを再度登録した上で tr-update-doc01 にアクセスして得られる結果は，図 12 に示すように “ failure ” となる。

```
- <page>
  <x:query oid="address3" type="update" result="failure">Failed to update
  resource address3: 1</x:query>
</page>
```

図 12 : XMLDBTransformer による “ update ” の実行結果

(*1) XMLDBTransformer コンポーネントについては，
<http://wiki.apache.org/cocoon/XMLDBTransformer>
<http://cocoon.apache.org/2.1/apidocs/org/apache/cocoon/transformation/XMLDBTransformer.html>

を参照する。また，ブロックサンプル xmlldb に置かれている populate.xml ファイルと sitemap.xmap ファイルの内容を十分に理解することが，XMLDBTransformer 利用への近道である。また，XUpdate に関わる記述については，文献(5)をも参照する。

(*2) sitemap.xmap ファイルでの説明では driver の指定はオプションとなっている。しかしながら，このように driver を明示的に指定しておく方が，「URI が不適切」といったブロックサンプルでのエラーの発生を少なくすることができる様である。

(*3) tr-add-doc01 に幾度アクセスしても同じ “ success ” の結果が得られる。既に登録済みの状態で xindice コマンドにより同じドキュメントを登録してみても結果は同様である(以下参照)。

```
C:\Tomcat5.0\webapps\cocoon\edu\edu20\xml1>xindice ad -c xmlldb:xindice://localhost:8080/db/addressbook -f address3.xml -n address3
trying to register database
```

```
Added document xmlldb:xindice://localhost:8080/db/addressbook/address3
```

このことは，Xindice への XML ドキュメント登録が，同じドキュメントキーの値で登録する限りにおいては，ドキュメント全体の修正(アップデート)と同様の結果を与えることを示している。

(*4) tr-del-doc01 へのアクセスの後 (address3 削除の後)，再度 tr-del-doc01 にアクセスした結果は次のとおりである。address3 が削除され，登録されていないことが分かる。

```
- <page>
  <x:query oid="address3" type="delete" result="failure">Resource
  address3 does not exist</x:query>
</page>
```

(*5) XMLDBTransformer による XUpdate について，同様の問題が次の URL にも報告されている。

<http://www.opensubscriber.com/message/users@cocoon.apache.org/1401609.html>

また，ブロックサンプルでも，XUpdate 部分が result="failure" となっている。ただし，この同じサンプル部分が，後ほど採りあげるサブレットコンテナである Jetty Server 上での cocoon ブロックサンプルでは，result="success" と XUpdate が正常になされている。

xindice コマンドによる XML ドキュメントの修正

addressbook/address3 での修正

XMLDBTransformer による「KML ドキュメントの修正」の項を補足する意味で、xindice コマンドを用いてドキュメントの修正を行う。xindice での書式は次のとおりである^(*)。

xindice xupdate -c /db/test -n document-to-update.xml -f /path/to/xupdate.xml

tr-update-doc01.xml ファイルの update に関わる部分を含む、次の内容の xindice-update01.xml ファイルを作成する（以下参照）。ファイルの内容は、

<email type="home">・・・</email>の要素を削除

<phone type="home">456-789-4321</phone>の電話番号を 321-654-4321 に変更

<email type="home"> smith@mail-efg.com</email>を<email type="work">・・・</email>

行の次行に作成

を示している。

xindice-update01.xml :

```
<?xml version="1.0"?>
<xu:modifications version="1.0" xmlns:xu="http://www.xmldb.org/xupdate">
  <xu:remove select="/person/email[@type = 'home']"/>
  <xu:update select="/person/phone[@type = 'home']">321-654-4321</xu:update>
  <xu:insert-after select="/person/email[@type = 'work']">
    <xu:element name="email">
      <xu:attribute name="type">home</xu:attribute>smith@mail-efg.com</xu:element>
    </xu:insert-after>
  </xu:modifications>
```

次のように実行する。実行結果を次に示す（以下参照）^(*)。

```
C:¥MyNote¥Xindice>xindice xupdate -c xmldb:xindice://localhost:8080/db/addressbook
-n address3 -f ./xindice-update01.xml
trying to register database
1 documents updated
= = = = = 以下は確認，修正は期待通りである = = = = =
C:¥MyNote¥Xindice>xindice xpath -c xmldb:xindice://localhost:8080/db/addressbook
-q "//person[fname='Richard']"
trying to register database
<person src:col="/db/addressbook" src:key="address3" xmlns:src="http://xml.apache.o
rg/xindice/Query">
  <fname>Richard</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-1234</phone>
  <phone type="home">321-654-4321</phone>
  <email type="work">richard@mailserv.com</email><email type="home">smith@mail-ef
g.com</email>

  <address type="work">88,Some City</address>
  <address type="home">55,Abce City</address>
</person>
```

http://localhost:8080/xindice/?/db/addressbook&address3 へのアクセスで得られる結果も併せて図 13 に示しておく。期待した修正がなされていることが分かる。



図 13 : ブラウザによる/addressbook/address3 の内容表示

kyuukou/kyu4 での修正

次に、kyuukou コレクションの kyu4 ドキュメントを対象として次の操作を行う。

<備考>補講は行わない</備考> この要素を削除

<担当教員>基盤 仙太</担当教員> <担当教員>基盤 専多</担当教員> に修正

<備考>都合により補講は休講とする</備考> この要素を追加

修正の内容は、上記の addressbook/address3 に対して行ったものと同様のものである。

準備として以下を内容とする xindice-update02.xml ファイルを作成する。

xindice-update02.xml :

```
<?xml version="1.0" encoding="Windows-31J"?>
<xup:modifications version="1.0" xmlns:xup="http://www.xmldb.org/xupdate">
  <xup:remove select="/休講/備考"/>
  <xup:update select="/休講/担当教員">基盤 専多</xup:update>
  <xup:insert-after select="/休講/授業名">
    <xup:element name="備考">都合により補講は休講とする</xup:element>
  </xup:insert-after>
</xup:modifications>
```

以下を実行する。結果は、設定した修正が行われなかったことを示している(*3)。

```
C:¥MyNote¥Xindice>xindice xupdate -c xmlldb:xindice://localhost:8080/db/kyuukou -n
kyu4 -f ./xindice-update02.xml
trying to register database
0 documents updated
```

Shift_JIS (Windows-31J) エンコーディングの漢字を含む xindice-update02.xml のような修正ドキュメントを用いた場合には、xindice による XUpdate はまだ対応していないようである(*4)。

(*1) “-n document-to-update.xml” を含まない

xindice xupdate -c /db/test -f /path/to/xupdate.xml

の書式もある。但し、このときは、コレクションに含まれる全ての XML ドキュメントが XUpdate の対象となるので、特定のドキュメントのみを対象とする場合には、その指定を何らかの形で (例えば xindice-update01.xml での/person を/person[fname='Richard']とするなど) 修正ドキュメ

ントに明示的に示すことが必要である。

(*2)用いた xindice-update01.xml ファイルにおいては、<email type="home">・・・</email>の要素を削除し、また、内容を変えて同様の要素を追加している。実は、このテストに先立って、<email type="home">・・・</email>の要素を削除するだけの修正が期待どおりになされることを確認している。

(*3) <xup:remove select="/休講/備考"/>のみを記述した xindice-update02.xml ファイルを用いての実行でも、結果は“ 0 documents updated ”であった。

(*4)上記の Sjift_JIS (Windows-31J) エンコーディングによる xindice-update02.xml ファイルに替えて、同ファイルを UTF-8 エンコーディングに変換した xindice-update03.xml ファイルを用いたところ、xindice コマンドによる修正が可能であった (以下参照)

```
C:\MyNote\Xindice>xindice xupdate -c xmldb:xindice://localhost:8080/db/kyuukou -n kyu4
-f ./xindice-update03.xml
trying to register database
1 documents updated
C:\MyNote\Xindice>
```

結果を確認するため xindice-xpath03 (前出の「 Pseudo Protocol を用いての登録ドキュメントの検索」の項目注(*1)を参照) にアクセスして得られた画面を次の図に示す。設定どおりの修正がなされている (担当教員名と備考の内容に注意)

日時	曜日	校時	担当教員	授業名	備考
1月19日	月曜	1校時	青木 昌三	物理学概論1	後日補講を実施する
1月20日	火曜	2校時	香川 太郎	香川大学の将来について	*
1月23日	金曜	3校時	情報 次郎	情報学基礎	補講を実施
1月26日	月曜	4校時	基盤 専多	情報環境論	都合により補講は休講とする

尚、xindice-update03(02).xml 等ファイルの先頭行

```
<?xml version="1.0" encoding="Windows-31J"?>
```

における属性 encoding の設定は今のところ意味をもたない。このことをソースコードにより確認したところ、utf-8 エンコーディングを前提として修正ファイルを読み込んでいることが確かめられた。

Groovy スクリプトによる XML ドキュメントの修正

コマンドラインツール xindice による XML ドキュメントの修正が、修正内容を記述した XML ドキュメントのエンコーディングが UTF-8 でない場合、“ 0 documents updated ” となって修正が不具合に終わることは既に述べた。

ここでは、Groovy スクリプトにより試作したプログラムを用いての XML ドキュメント修正について紹介する。このスクリプトは、UTF-8 以外のエンコーディングにも対応する。尚、スクリプト作成にあたっては、ドキュメントの「Xindice 1.1 Developer Guide」と文献(5)を参考としている。

作成したスクリプトは xindice_xupdate.groovy で、使い方は **groovy xindice_xupdate[.groovy] collection xupdate-xml-file** である。スクリプト内容を付録 C に示している(*1)。

コレクション addressbook での修正

ここでは xindice コマンドによる修正のところを用いた xindice-update01.xml ファイルの

内容にそった修正を行う（以下参照）。

```
C:¥MyNote¥Xindice>groovy xindice_xupdate addressbook ./xindice-update01.xml
trying to register database
```

```
xmldb:xindice://localhost:8080/db/addressbook
documents updated
```

```
C:¥MyNote¥Xindice>
```

xindice_xupdate.groovy では、付録 B に示した xindice_xpath.groovy と同様に try/catch 節を省略している。スクリプトの手軽さでもあるが、実行において例外等について何の表示もなされずに最終行に到達すれば、目的は達せられたものと解釈できる。変更内容の確認については割愛する。

kyuukou/kyu4 での修正

ここでは xindice コマンドによる修正のところ で用いた xindice-update02.xml ファイルの内容を少し変更した xindice-update04.xml にそった修正を行う。修正の結果が容易に確認できるよう、担当教員名と備考欄の内容を変更している。文字エンコーディングは Windows-31J である。

xindice-update04.xml :

```
<?xml version="1.0" encoding="Windows-31J"?>
<xu:modifications version="1.0" xmlns:xu="http://www.xmldb.org/xupdate">
  <xu:remove select="/休講[ID = '4']/備考"/>
  <xu:update select="/休講[ID = '4']/担当教員">幸町 分室</xu:update>
  <xu:insert-after select="/休講[ID = '4']/授業名">
  <xu:element name="備考">補講にかえてレポートを課す</xu:element>
  </xu:insert-after>
</xu:modifications>
```

実行は次の通りである（下記および図 14 を参照）。

```
C:¥MyNote¥Xindice>groovy xindice_xupdate kyuukou ./xindice-update04.xml
trying to register database
```

```
xmldb:xindice://localhost:8080/db/kyuukou
documents updated
```

```
C:¥MyNote¥Xindice>
```

休 講 通 知						
ID番号	日時	曜日	校時	担当教員	授業名	備考
4	1月26日	月曜	4校時	幸町 分室	情報環境論	補講にかえてレポートを課す

図 14 : Windows-31J エンコーディングファイルによるドキュメントの修正

上記は Windows-31J (Shift_JIS) エンコーディングの xindice-update04.xml ファイル

を用いた場合の例であるが、UTF-8 は勿論のこと、EUC_JP、ISO-2022-JP、Unicode エンコーディングに変換した xindice-update04.xml 相当のファイルに対して、何れの場合も作成したスクリプトによって kyu4 ドキュメントの修正がなされることが確認できている。

(*1)xindice_xupdate.groovy スクリプトにおいてコメント文の処理を行っているが、このことは特に必要という訳ではない。コメント文の書式が間違っていない限り、Xindice 側でコメント文に適切に対応する。

XSP ファイルでのプログラミング

例えば文献(1)、(2)にもあるように、HTML ページの FORM による入力に基づいて XML ドキュメントファイルを作成するとともに、そのドキュメント内容を Xindice に登録することを考えてみる。Cocoon2 での処理方法の 1 つは、XML ドキュメント中に Java 言語によるロジックを記述できる XSP (eXtensible Server Pages) を用いることであろう。

ここでは、1 つのサンプルとして作成した XSP ファイル (xindice-adddoc-01.xsp) について紹介する。作成した XSP ファイルの詳細については、付録 D を参照していただきたい(*1)。XSP ファイルでの処理内容は次の通りである。

(1)HTML ページの FORM から「休講通知」についてのデータ入力得られたと仮定して、それらの入力値に基づいて XML ドキュメントファイル(kyu5.xml)を定められたディレクトリ(xml2)内に作成する(*2)。

(2)kyu5.xml ファイルと同内容の XML ドキュメントを、kyu5 なるキー値をつけて Xindice のコレクション kyuukou 内に登録する。

XSP ファイルを用いた結果を眺めてみる。sitemap.xmap ファイルに次の記述を加え、ブラウザによりアクセスする。

sitemap.xmap ファイルへの追加記述：

```
<map:match pattern="xindice-adddoc-xsp01">
  <map:generate src="xindice-adddoc-01.xsp" type = "serverpages"/>
  <map:serialize type="xml" />
</map:match>
```

先ず、ブラウザによるアクセスでは次の結果が得られる(図 15)。XML ドキュメントの作成、コレクション kyuukou への登録が共に成功したことが示されている(付録 D に示されているコード内容を参照)。

```

- <page>
- <to-xml>
  <result>success</result>
</to-xml>
- <to-xindice>
  <collection>kyuukou</collection>
  <oid>kyu5</oid>
  <result>success</result>
</to-xindice>
</page>

```

図 15 : xindice-adddoc-xsp01 へのアクセス結果

実際に，xml2 ディレクトリ内の kyu5.xml ファイルの存在とその内容を確認しておく（下記参照）。

```

C:\Tomcat5.0\webapps\cococon\edu\edu20\xml2>type kyu5.xml
<?xml version="1.0" encoding="Windows-31J"?>
<休講>
  <ID>5</ID>
  <日付コード>2006012810</日付コード>
  <日時>1 月 28 日</日時>
  <曜日>水曜</曜日>
  <校時>1 校時</校時>
  <担当教員>自然 教学</担当教員>
  <授業名>自然科学を如何に学ぶか</授業名>
  <備考>補講日は追って通知する</備考>
</休講>
C:\Tomcat5.0\webapps\cococon\edu\edu20\xml2>

```

次に，http://localhost:8080/xindice/?/db/kyuukou へのアクセスで，コレクション kyuukou 内に XML ドキュメントのキー値 kyu5 が登録されていることを確認する（図 16 参照）。



図 16 : kyuukou に登録されている kyu5 ドキュメント

締めくくりとして，kyu5 ドキュメントの内容をブラウザにより確認する（図 17 参照）^(*)3)。

休 講 通 知

ID番号	日時	曜日	校時	担当教員	授業名	備考
5	1月28日	水曜	1校時	自然 教学	自然科学を如何に学ぶか	補講日は追って通知する

図 17：ブラウザによる kyu5 ドキュメントの検索結果

以上により、XSP での記述が期待した結果を与えていることが確認できた。ここで示したのは XML ドキュメントファイルの作成とその内容の Xindice への登録についてであるが、Xindice へのドキュメント登録は、同じキー値をもつ限りドキュメントの修正（アップデート）ともなっている。実際の Web アプリケーション作成に当って、例えば情報の作成とその修正などの対話的ページ場面において XSP を用いることの有用性が示されたといえよう^(*)。尚、付録 D に示したコード内容からも分かる様に、XML ドキュメント登録の Groovy スクリプト作成は容易であることを付け加えておきたい。

(*1)XSP ファイル作成にあたっては、ドキュメントの「Xindice 1.1 Developer Guide」を参考としている。尚、Collection の CreateResource(String id, String type) Method については、次の URL を参照した。

<http://xmldb-org.sourceforge.net/xapi/api/org/xmldb/api/base/Collection.html>

(*2)作成した XSP ファイルでは、入力値に相当するデータを予め与えている。それらのデータを与える HTML ページについては適宜作成してほしい。

(*3)図 17 は、前出の xindice-xpath02 へのアクセスで得られたものである。但し、saitemap.xmap ファイルにおいて、XPath の記述で ID の値を 5 としている。

(*4)XSP ファイルを用いた Standalone モード Xindice に登録されたドキュメントの修正については、後出の「 Cocoon2-Embedded モード Xindice の利用」における「 XSP ファイルでのプログラミング、XML ドキュメントの修正 XSP」項の(*)を参照してほしい。

Cocoon2-Embedded モード Xindice の利用

Cocoon2-Embedded モード Xindice の利用について、これまでに述べてきた Standalone モードでの利用についてと同様の基本機能の動作確認を行う。Cocoon2 では、インストール時に(ブロックサンプルを導入するようコンパイルされているならば) Xindice も導入され、Cocoon2 の起動と共に Xindice が設定され Embedded モードとして動作している。データベースの所在は、

`%CATALINA_HOME%\webapps\cocoon\WEB-INF\db`
ディレクトリである。

尚、以下において取り扱う XML ドキュメントは、これまで用いてきたものと同じ address1.xml や kyu1.xml 等である。

コマンドラインツール xindice による Embedded モードデータベースへのアクセス

コマンドラインツールである xindice (xindice.bat, xindice.sh など) は、Xindice のインストールに伴って導入されたもので、これまで見てきたように、Standalone モードでの Xindice を操作の対象とする。しかしながら、コレクションの生成や XML ド

コメントの登録等に極めて便利なツールであり、Embedded モードでの Xindice データベースへのアクセスに利用することができれば、何かと重宝する。表題の「Cocon2-Embedded モード Xindice の利用」から少し話がそれて寄り道となるが、Embedded モード Xindice へのアクセスについて、xindice コマンドによるアプローチを試みしてみる。

Configuration ファイルの作成

先ず準備として、データベース Configuration ファイルを作成する。%XINDICE_HOME%内の system.xml ファイルと同内容の system-embed.xml ファイルを同じディレクトリ内に作成し（コピーすればよい）、system-embed.xml ファイルを次のように変更する（関連部分のみを示す。変更等は太字部分）。

system-embed.xml ファイルの変更内容（関連部分）:

```
-           for this database instance.
-->
  <!--root-collection dbroot="/db/" name="db" use-metadata="on"--> コメントアウト
<root-collection dbroot="../../cocoon/WEB-INF/db/" name="db" use-metadata="on">追加
記述
  <!--
    - Query Engine Configuration.
  -->
```

コレクションの検索

先ず Embedded モードでどのようなコレクションが作成・登録されているかを見る。

実行コマンドは

xindice lc -c xmldb:xindice-embed:///db -d %XINDICE_HOME%¥system-embed.xml
である^(*)。実行結果を示す（以下参照）。これから、cocoon、system、meta の3つの
コレクションが生成・登録されていることが分かる。

```
C:¥MyNote¥Xindice>xindice lc -c xmldb:xindice-embed:///db -d %XINDICE_HOME%¥
system-embed.xml
trying to register database
[INFO] DatabaseImpl - -Specified configuration file: 'c:¥Tomcat5.0¥webapps¥xindice¥
WEB-INF¥system-embed.xml'
[INFO] Database - -Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥db

    cocoon
    system
    meta

Total collections: 3
C:¥MyNote¥Xindice>
```

コレクション addressbook20 の生成

コレクション addressbook20 を生成する。コマンドは

xindice ac -c xmldb:xindice-embed:///db -n addressbook20 -d %XINDICE_HOME%

¥system-embed.xml

である（誌面の都合上，1行で示すところを折り返している）。実行結果を示す（以下参照）。コレクション addressbook20 が生成されていることが分かる。

```
C:¥MyNote¥Xindice>xindice ac -c xmldb:xindice-embed:///db -n addressbook20 -d %XINDICE_HOME%¥system-embed.xml
trying to register database
[INFO] DatabaseImpl - -Specified configuration file: 'c:¥Tomcat5.0¥webapps¥xindice¥WEB-INF¥system-embed.xml'
[INFO] Database - -Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥db
[INFO] Collection - -[main] '/db/system/SysSymbols' Set object addressbook20
===== 中 略 =====
[INFO] CollectionManager - -Created a new collection named 'addressbook20'
Created : xmldb:xindice-embed:///db/addressbook20
===== 確 認 =====
C:¥MyNote¥Xindice>xindice lc -c xmldb:xindice-embed:///db -d %XINDICE_HOME%¥system-embed.xml
trying to register database
[INFO] DatabaseImpl - -Specified configuration file: 'c:¥Tomcat5.0¥webapps¥xindice¥WEB-INF¥system-embed.xml'
[INFO] Database - -Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥db

    addressbook20
    cocoon
    system
    meta

Total collections: 4
```

尚，addressbook20 コレクションの生成は，
%CATALINA_HOME%¥webapps¥cocoon¥WEB-INF¥db¥addressbook20
ディレクトリが作成されていることによっても確認できる。

XML ドキュメントの登録

コレクション addressbook20 内に address1 をキー値として./xml1/address1.xml を登録する。コマンドは次の通りである（誌面の都合上，1行で示すところを折り返している）。

```
xindice ad -c xmldb:xindice-embed:///db/addressbook20 -n address1 -f ./xml1/address1.xml -d %XINDICE_HOME%¥system-embed.xml
```

実行の様子と結果を確認する（以下参照）。

```
C:¥MyNote¥Xindice>xindice ad -c xmldb:xindice-embed:///db/addressbook20 -n address1 -f ./xml1/address1.xml -d %XINDICE_HOME%¥system-embed.xml
trying to register database
===== 中 略 =====
Added document xmldb:xindice-embed:///db/addressbook20/address1
===== 確 認 =====
C:¥MyNote¥Xindice>xindice ld -c xmldb:xindice-embed:///db/addressbook20 -d %XINDICE_HOME%¥system-embed.xml
```

==== 中 略 =====

```
[INFO] Database - -Database points to C:\Tomcat5.0\webapps\cocoon\WEB-INF\db
address1
```

Total documents: 1

XML ドキュメントの検索・内容確認

先に登録した address1 ドキュメントの内容を確認する意味で、/person/lname を検索してみる。コマンドは

```
xindice xpath -c xmldb:xindice-embed:///db/addressbook20 -q "/person/lname" -d %
XINDICE_HOME%\system-embed.xml
```

である。実行結果から、lname として “Smith” が検索されている（以下参照）

```
C:\MyNote\Xindice>C:\MyNote\Xindice>xindice xpath -c xmldb:xindice-embed:///db/
addressbook20 -q "/person/lname" -d %XINDICE_HOME%\system-embed.xml
trying to register database
==== 中 略 =====
<lname xmlns:src="http://xml.apache.org/xindice/Query" src:col="/db/addressbook20"
src:key="address1">Smith</lname>
```

XML ドキュメントの削除と修正

ドキュメントの削除

XML ドキュメントの削除については、新たに address2 ドキュメントを登録した上で削除コマンドを実行して確認する。結果は

```
(1) C:\MyNote\Xindice>xindice ad -c xmldb:xindice-embed:///db/addressbook20 -n
addre
```

```
ss2 -f ./xml1/address2.xml -d %XINDICE_HOME%\system-embed.xml
```

```
結果： Added document xmldb:xindice-embed:///db/addressbook20/address2
```

```
(2) C:\MyNote\Xindice>xindice dd -c xmldb:xindice-embed:///db/addressbook20 -n
addre
```

```
ss2 -d %XINDICE_HOME%\system-embed.xml
```

```
結果： [INFO] Collection - -[main] '/db/meta/Metas/addressbook20' Remove addre
ss2
```

```
DELETED: xmldb:xindice-embed:///db/addressbook20/address2
```

となって address2 が削除されたことを示している（表示されることの詳細は省略している）

再度削除コマンドを実行すると

```
(3) C:\MyNote\Xindice>xindice dd -c xmldb:xindice-embed:///db/addressbook20 -n
addre
```

```
ss2 -d %XINDICE_HOME%\system-embed.xml
```

```
結果： [INFO] Database - -Database points to C:\Tomcat5.0\webapps\cocoon\
```

WEB-INF¥db

XMLDB Exception 301: Resource passed is null or its ID is empty.

の様に、address2 が存在しない旨が表示される。

ドキュメントの修正

比較を容易にするため、上述の Standalone モードでの xindice コマンドによるドキュメント修正と全く同じ内容となるよう、新たに address3 ドキュメントを登録した上で修正ファイル xindice-update01.xml ファイルの内容にそって修正を実行する（Standalone モード Xindice の利用の項「 xindice コマンドによる XML ドキュメントの修正」を参照）。実行結果は、現時点では例外の発生により以下に示すように不具合に終わっている^(*)。

(1) C:¥MyNote¥Xindice>xindice ad -c xmldb:xindice-embed:///db/addressbook20 -n
adresse

ss3 -f ./xml1/address3.xml -d %XINDICE_HOME%¥system-embed.xml

結果： Added document xmldb:xindice-embed:///db/addressbook20/address3

(2) C:¥MyNote¥Xindice>xindice xupdate -c xmldb:xindice-embed:///db/addressbook
20 -n

address3 -f ./xindice-update01.xml -d %XINDICE_HOME%¥system-embed.xml

結果： Exception in thread "main" javax.xml.parsers.FactoryConfigurationException:
java.lang.ClassNotFoundException:org.apache.xindice/core/xupdate/XPathQueryFactoryImpl
at org.xmldb.common.xml.queries.XPathQueryFactory.newInstance(Unknown Source) 以下略

(*1)オプション-d を-ld とすると Standalone モードでのデータベースコレクションが表示される。また、オプションを-d としたままで-c xmldb:xindice:///db とする、あるいは Configuration ファイル名のスペルを間違えるなどのときは、「XMLDB Exception 0: Cannot communicate with the server: http://127.0.0.1:8888/xindice/」のエラーが表示される。

(*2)登録されている address1 ドキュメントを対象とした修正でも同様の例外により不具合となる。一方、登録されていない削除済みの address2 ドキュメントに対する同様のコマンド実行では、以下に示すようにドキュメントが存在しないことによる例外は無視されて、0 修正の結果を与えてコマンドは終了する。修正実行の不具合は、ドキュメントが登録されているが故の例外によるエラーである。尚、関連して後出の「 XSP ファイルでのプログラミング」の項(*1)をも参照してほしい。

```
C:¥MyNote¥Xindice>xindice xupdate -c xmldb:xindice-embed:///db/addressbook20 -n  
adresse2 -f ./xindice-update01.xml -d %XINDICE_HOME%¥system-embed.xml  
trying to register database  
[INFO] DatabaseImpl - -Specified configuration file: 'c:¥Tomcat5.0¥webapps¥xindice¥WEB-INF¥system-embed.xml'  
[INFO] Database - -Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥db  
[INFO] Collection - -[main] '/db/addressbook20' Query document address2, query:  
===== 中略 =====  
[WARN] XUpdateQueryResolver - -ignored exception <java.lang.NullPointerException  
>java.lang.NullPointerException  
at org.apache.xindice.core.xupdate.XUpdateQueryResolver$XUpdateQuery.execut
```

```
e(XUpdateQueryResolver.java:142)
```

```
===== 中 略 =====
```

```
0 documents updated
```

Pseudo Protocol の利用

Pseudo Protocol である “ xmlldb:xindice-embed “ を用いて XML ドキュメントを検索する。これまでのところ，登録されているドキュメントはコレクション addressbook20 内の address1 と登録後の修正がうまくいかなかった address3 である。sitemap.xmap ファイルに以下の記述を加え，address1 ドキュメントを検索する。上述した Standalone モードでの xindice-xpath01 に相当するものであるが，下記において太字の部分が Embedded モード対応となっている。

```
<map:match pattern="xindice-xpath21">
  <map:generate
src="xmlldb:xindice-embed://db/addressbook20/#/person[fname='John']"/>
  <map:serialize type="xml"/>
</map:match>
```

xindice-xpath21 へのアクセスによって，図 4 に示したのと同様の結果が得られる。

以後，Embedded モードでの Xindice へのアクセスについては，例えば Standalone モードでの xindice-xpath01 に相当するアクセス先を xindice-xpath21 としたように，01，02，03・・・を 21，22，23・・・に

xmlldb:xindice://localhost:8080/db を xmlldb:xindice-embed://db に

それぞれ必要に応じて書き換えて，検証を進めていく。特に断らない限り，kyuukou20 ならば kyuukou の，tr-create-coll21 ならば tr-create-coll01 の Embedded モード対応版である。

XMLDBTransformer の利用

コレクション kyuukou20 の生成

コレクションの生成例として，kyuukou20 コレクションを生成する。kyuukou20 コレクションを生成するよう tr-create-coll01.xml ファイルを修正した tr-create-coll21.xml ファイルを作成した上で，sitemap.xmap ファイルに tr-create-coll21 を追加記述する。但し，sitemap.xmap ファイルでの tr-create-coll01 における XMLDBTransformer に関わる記述

```
<map:transform type="xmlldb">
  <map:parameter name="base" value="xmlldb:xindice://localhost:8080/db"/>
</map:transform>
```

において 2 行目は不要であり，コメントアウトするか，または単に

```
<map:transform type="xmlldb"/>
```

とする（以下においても同様）。

tr-create-coll21 へのアクセス結果は図 18 の通りで，これは前出の図 6 に相当する。

```

- <page>
  <x:query oid="kyuukou20" type="create" result="success"/>
</page>

```

図 18：コレクション kyuukou20 の生成

XML ドキュメントの登録

ドキュメント登録として，addressbook20 コレクションに address2.xml と kyuukou20 コレクションに kyu1.xml ファイルを登録する。キーの値はそれぞれ address2 と kyu1 である。

tr-test1.xml，tr-test2.xml ファイルをそれぞれ修正して tr-test21.xml，tr-test22.xml ファイルとし，それぞれに対応する sitemap.xmap ファイル上の URL である tr-add-doc21，tr-add-doc22 にアクセスする。図 9 と同様の “success” 結果が得られ，ドキュメントの登録は成功した。Pseudo Protocol “xmldb:xindice-embed” によるドキュメント検索機能を再度確認するため，コレクション kyuukou20 内の ID の値が 1 であるドキュメントを検索するよう設定した xindice-xpath22 にアクセスした結果を図 19 に示す。

休 講 通 知						
ID番号	日時	曜日	校時	担当教員	授業名	備考
1	1月19日	月曜	1校時	青木 昌三	物理学概論1	後日補講を実施する

図 19：Pseudo Protocol によるドキュメント検索の結果

登録 XML ドキュメントの削除

先程登録したコレクション addressbook20 中の address2 を削除する。tr-del-doc01.xml ファイルを修正し tr-del-doc21.xml とした上で，対応する URL：tr-del-doc21 にアクセスすると図 11 と同様の結果が得られる。

XML ドキュメントの修正

問題の XML ドキュメントの修正である。tr-update-doc01.xml を修正して，コレクション addressbook20 内のドキュメント address3 において自宅のメールアドレス要素を削除するよう設定した tr-update-doc21.xml を作成する。対応する URL：tr-update-doc21 へのアクセスによって得られる結果は，Standalone モードでの図 12 と同様，修正は不成功である^(*)。

(*)xindice コマンドツールによるドキュメントの修正が不成功に終わっている。そこでの(*)で述べたことと関連して，存在しないコレクション addressbook202 と登録されていないドキュメント address2 への修正結果を示しておく。

先ず tr-update-doc21.xml において修正対象のコレクションを addressbook202 とする。実行の結果は次の様に addressbook202 が見つからないことから不成功 (failure) である。

```
- <page>
- <x:query oid="address3" type="update" result="failure">
  Failed to update resource address3: Collection xmldb:xindice-embed:///db/addressbook202 not found.
</x:query>
```

次にコレクションを addressbook20 とし、修正対象ドキュメント (oid) を address2 として修正のため tr-update-doc21 にアクセスする。結果は図 12 に示されたのと同様の不成功である。

Groovy スクリプトによる XML ドキュメントの修正

コマンドラインツール xindice と XMLDBTransformer による XML ドキュメントの修正が、共に不首尾に終わっている。Groovy スクリプトにより、ドキュメントの修正が可能なことを確かめておく。

XML ドキュメントの検索スクリプト

準備として、付録 B に示した xindice_xpath.groovy を基に、Embedded モード Xindice 内の XML ドキュメントにアクセスするよう修正した xindice_xpath_embed.groovy を作成する。このスクリプト作成を通じて、修正すべき点が明確となる。

主な修正点は次の 2 点である (付録 B を参照)。

(1) Xindice の Configuration ファイルの設定

```
driver = "org.apache.xindice.client.xmlldb.DatabaseImpl" // OK
c = Class.forName(driver)
database = c.newInstance()
database.setProperty("configuration", "/Tomcat5.0/webapps/cocoon/WEB-INF
  /xindice.xml")
```

DatabaseManager.registerDatabase(database)

の様に database.setProperty(…) 行を追加し、database にプロパティ "configuration" として Xindice の Configuration ファイル名を設定する。ここで、設定されているファイル xindice.xml は cocoon2 インストール時の Configuration ファイルである。先に xindice からのアクセス用に作成した %XINDICE_HOME%\system-embed.xml を使用してもかまわない。また、上記の setProperty(…) に代えて、プロパティ "db-home" として Xindice データベースの所在を与えてもよい。このときは次の通りとする。

```
database.setProperty("db-home", "/Tomcat5.0/webapps/cocoon/WEB-INF/")
```

(2) アクセス uri の変更

```
uri = "xmldb:xindice://localhost:8080/db/" + args[0]
```

を

```
uri = "xmldb:xindice-embed:///db/" + args[0]
```

と変更する。

作成した xindice_xpath_embed.groovy を用いて、例えば

```
groovy xindice_xpath_embed addressbook20 "/person"
```

```
groovy xindice_xpath_embed kyuukou20 "/休講"
```

の実行により、登録されている address1, address3 や kyu1 ドキュメントの内容表示が可能である。

XML ドキュメントの修正スクリプト

Standalone モードでのドキュメント修正スクリプト xindice_xupdate.groovy を変更して xindice_xupdate_embed.groovy を作成する。主たる変更点は上述の(1), (2) に述べた点であるが、コマンド引数に修正の対象とするドキュメントの id 値(キーの値)をとるようにすること、修正した件数を表示することを加えている(これらの点については、付録 C の注(*)を参照のこと)。実行コマンドは

```
groovy xindice_xupdate_embed[.groovy] collection doc-id xupdate-xml-file
```

である。doc-id を “ all ” (“ と ” は不要) としたときには、collection 内の全てのドキュメントが修正の対象となる。

例として、kyuukou20 コレクションのドキュメントを修正する。修正内容を記述したファイルとしては、前出の xindice-update04.xml を変更した xindice-update24.xml を用いる(内容は以下を参照)。xindice-update04.xml との違いは、対象とするドキュメントが指定されるので [ID = '4'] の記述が不要となっていることである。修正箇所は3つである。

xindice-update24.xml の内容：

```
<?xml version="1.0" encoding="Windows-31J"?>
<xu:modifications version="1.0" xmlns:xu="http://www.xmldb.org/xupdate">
  <xu:remove select="/休講/備考"/>
  <xu:update select="/休講/担当教員">幸町 分室</xu:update>
  <xu:insert-after select="/休講/授業名">
  <xu:element name="備考">補講にかえてレポートを課す</xu:element>
  </xu:insert-after>
</xu:modifications>
```

日時	曜日	校時	担当教員	授業名	備考
1月19日	月曜	1校時	香木 昌三	物理学概論1	後日補講を実施する
1月20日	火曜	2校時	香川 太郎	香川大学の将来について	*
1月23日	金曜	3校時	情報 次郎	情報学基礎	補講を実施
1月26日	月曜	4校時	基盤 仙太	情報環境論	補講は行わない

図 20：修正実行前の登録ドキュメント

kyu1 ~ kyu4 ドキュメントを登録した後、コマンドを実行する。先ず実行前の状況を

示しておく（図 20 参照）

修正対象を“all”として実行する（下記参照）。”documents updated : 12”からも分かる様に修正箇所は計 12 箇所（3*4=12）である。

```
C:\MyNote\Xindice>groovy xindice_xupdate_embed kyuukou20 all xindice-update24.xml
trying to register database
2005/10/15 6:23:18 org.apache.xindice.client.xmldb.embed.DatabaseImpl loadConfiguration
情報: Specified configuration file: '/Tomcat5.0/webapps/cocoon/WEB-INF/xindice.xml/'
2005/10/15 6:23:18 org.apache.xindice.core.Database setConfiguration
情報: Database points to C:\Tomcat5.0\webapps\cocoon\WEB-INF\db
2005/10/15 6:23:19 org.apache.xindice.core.Collection setDocument
===== 中 略 =====
情報: [main] '/db/meta/Metas/kyuukou20' Set object kyu3
2005/10/15 6:23:19 org.apache.xindice.core.Collection setDocument
情報: [main] '/db/kyuukou20' Set document kyu4
2005/10/15 6:23:19 org.apache.xindice.core.Collection setObject
情報: [main] '/db/meta/Metas/kyuukou20' Set object kyu4

xmlldb:xindice-embed:///db/kyuukou20 : doc-id = all
documents updated : 12
```

スクリプト実行後の登録ドキュメントの内容をブラウザにより確認する（図 21 参照）。

休 講 通 知					
日時	曜日	校時	担当教員	授業名	備考
1月19日	月曜	1校時	幸町 分室	物理学概論1	補講にかえてレポートを課す
1月20日	火曜	2校時	幸町 分室	香川大学の将来について	補講にかえてレポートを課す
1月23日	金曜	3校時	幸町 分室	情報学基礎	補講にかえてレポートを課す
1月26日	月曜	4校時	幸町 分室	情報環境論	補講にかえてレポートを課す

図 21：修正実行後の登録ドキュメント

次に、修正内容で担当教員を「十月 十五」、備考を「補講なし」に変更した上で、修正対象ドキュメントを kyu3 としスクリプトを実行する（下記参照）^(*)。kyu3 ドキュメントが修正されて図 22 が得られる。

```
C:\MyNote\Xindice>groovy xindice_xupdate_embed kyuukou20 kyu3 xindice-update24.xml
trying to register database
===== 中 略 =====
情報: [main] '/db/meta/Metas/kyuukou20' Set object kyu3

xmlldb:xindice-embed:///db/kyuukou20 : doc-id = kyu3
documents updated : 1
```

休 講 通 知						
ID番号	日時	曜日	校時	担当教員	授業名	備考
3	1月23日	金曜	3校時	十月 十五	情報学基礎	補講なし

図 22：修正実行後の kyu3 ドキュメント

以上で、Groovy スクリプト `xindice_xupdate_embed.groovy` による Xindice データの修正が期待通りであることが示された(*2)。

(*1)修正箇所数を示す数字が、`doc-id` が `all` であるとき (12) と特定のドキュメントを対象とするとき (3 ではなく 1) で異なっている。

全てのドキュメントを対象とする `service.update()`での戻り値は全修正箇所数

特定のドキュメントを対象とする `service.updateResource()`での戻り値はドキュメント数となるようである。

(*2)実は、スクリプトの実行結果は必ずしも常に安定しているとは言えない。ここに示している結果は、Tomcat を再起動した後に得られたものである。スクリプトの実行において何のエラーも報告されず、また修正箇所数もきちんと与えられていることからスクリプトの不具合とは思えないが、例えば `doc-id` を `all` としたときでも、`kyu2` と `kyu3` だけに修正が反映されるなどの現象がみられた。

XSP ファイルでのプログラミング

XML ドキュメントの登録 XSP

Standalone モード Xindice に XML ドキュメントを作成・登録する `xindice-adddoc-01.xsp` を先に作成した。この XSP ファイルを手直しして、Embedded モード対応版を作成する。XSP は Cocoon 上で動作するので、Groovy スクリプトの場合より手直しが簡単である。付録 D の内容と比較しつつ、手直しすべき点について述べる。

(1) ドライバーについて

```
String driver = "org.apache.xindice.client.xmlldb.DatabaseImpl";
```

のままでもよいし、次の様でも構わない。

```
String driver = "org.apache.xindice.client.xmlldb.embed.DatabaseImpl";
```

(2) プロパティの設定について

Groovy スクリプトでは `database.setProperty(・・・)`を記述してプロパティ値を設定したが、XSP コードでは不要である。設定しても差し支えないが、`Configuration` ファイル名やデータベースの所在個所は Cocoon2 サイドでは既知のことである。

(3) アクセス uri について

```
String uri = "xmlldb:xindice-embed:///db/" + coll;
```

```
coll = DatabaseManager.getCollection(uri);
```

とする。coll はコレクション名で `kyuukou` に代えて `kyuukou20` である。

その他、作成する XML ドキュメントを置くディレクトリ名の変更等があるが、たいした問題ではない。実行結果を、図 15 と重複するが図 23 に示す。

```
- <page>
- <to-xml>
- <file-name>
  c:/tomcat5.0/webapps/cocoon/edu/edu20/xml22/kyu5.xml
</file-name>
<result>success</result>
</to-xml>
- <to-xindice>
  <collection>kyuukou20</collection>
  <oid>kyu5</oid>
  <result>success</result>
</to-xindice>
</page>
```

図 23：xindice-adddoc-xsp21 へのアクセス結果

XML ドキュメントの修正 XSP

Embedded モード Xindice に登録のドキュメント修正については、XMLDBTransformer によっても、またコマンドラインツール xindice によっても期待した修正結果を得ることができていない。だとすれば、Cocoon2 からの Xindice 利用を考慮するならば、XSP ファイルによる Xindice のドキュメント修正を何としても取り扱いたいところである。ここでは、XML ドキュメントを登録する XSP ファイルが期待通りに動作することを受けて、ドキュメントを修正する XSP ファイルを作成する。

作成 XSP ファイルの概要

作成の XSP ファイル名は xindice-xupdate-22.xsp で、これを利用するための URL として sitemao.xmap ファイルに次の内容を追加記述する（以下参照）。

```
<map:match pattern="xindice-xupdate-xsp22">
  <map:generate src="xindice-xupdate-22.xsp" type="serverpages"/>
  <map:serialize type="xml" />
</map:match>
```

xindice-xupdate-22.xsp のコード内容については付録 E に示している。これからも分かる様に、修正対象のドキュメントはコレクション kyuukou20 内の kyu1 ,kyu2 ,・・・であり、修正の内容はこれまで取り扱ってきた内容と同じである。コードについての主だった点を列挙しておく。

- (1) 雛型は、既に一定の成果を得ている Groovy スクリプト xindice_xupdate_embed.groovy と上記 XML ドキュメント登録 XSP ファイル Embedded モード対応版である（付録 D の XSP コードをも参照）。
- (2) 修正内容は、例えばインタラクティブな HTML ページの FORM や修正内容を記述したファイル等からデータ入力得られたと仮定して、コード内に記述したところから出発している。尚、修正ファイルからのデータ入力を具体的に記述するには、付録 C

に示した Groovy スクリプトの前半部分を参照して Java 言語スタイルに書き変えていただきたい。

(3)修正対象 XML ドキュメントのキー値(oid)は便宜上デフォルトでは kyu4 である。kyu2 ドキュメントを対象とするには、次の URL でアクセスする。

URL : xindice-xupdate-xsp22?oid=kyu2

oid=all とすれば全てのドキュメントが対象となる。尚、対象ドキュメントの oid が例えば

“oid=休講”の様に漢字を含む場合への対応については、サンプルのコードをコメント文として記述している。

(4)ドライバーの選択、プロパティの設定(今回は不要ではあるが)のためのコードも参考のためコメント文として記述している。

ドキュメント修正の実行と結果

作成 XSP ファイルによるドキュメント修正を実行し結果を確認する。まず kyu1 ~ kyu4 を再登録し、これまでの修正テストにより変更されている内容を初期の状態に戻す。XML ドキュメント登録 XSP ファイルの実行で kyu5 も登録されているので、全体の登録状況は図 20 に示された内容に kyu5 ドキュメントが加わった状態である。

xindice-xupdate-xsp22 にアクセスすると次の結果が得られる(図 24 参照)。

```
- <page>
- <xupdate-xindice>
  <collection>kyuukou20</collection>
  <oid>kyu4</oid>
  <result>success</result>
  <update-num>1</update-num>
</xupdate-xindice>
</page>
```

図 24 : xindice-xupdate-xsp22 へのアクセス結果

結果は成功(success)であり、修正個所数は、この場合特定の kyu4 ドキュメントを対象としているので1である。全体の登録状況の確認から kyu4 ファイルが修正されていることが確認できる。

xindice-xupdate-xsp22?oid=all にアクセスする。得られる結果は図 25 の通りとなる。

```
- <page>
- <xupdate-xindice>
  <collection>kyuukou20</collection>
  <oid>all</oid>
  <result>success</result>
  <update-num>15</update-num>
</xupdate-xindice>
</page>
```

図 25 : xindice-xupdate-xsp?oid=all へのアクセス結果

結果は成功であると共に,kyu1~kyu5の5つのドキュメントを修正の対象としているので,修正箇所数は $3*5=15$ となっている。また,修正実行後の全体の登録状況としてkyu5ドキュメントが加わった図21と同様の図が得られ,kyu1~kyu5ドキュメントの全てにおいて担当教員名欄が“幸町 分室”に,備考欄が“補講にかえてレポートを課す”に修正されていることも確認できる。

Groovy スクリプトによる kyu3 ドキュメントを対象とした修正結果 (図 22 を参照) と同様の結果が得られることを確認するため, XSP コードで

```
String teacher = "十月 十七";
```

```
String comment = "補講は後日に連絡通知";
```

と変更して, xindice-xupdate-xsp22?oid=kyu3 にアクセスする。実行の結果としては <oid>kyu3</oid>とした図 24 と同様の結果が得られ,kyu3 ドキュメントについては図 22 に対応する次の図 26 が得られる。

休 講 通 知						
ID番号	日時	曜日	校時	担当教員	授業名	備考
3	1月23日	金曜	3校時	十月 十七	情報学基礎	補講は後日に連絡通知

図 26 : xindice-xupdate-xsp22?oid=kyu3 へのアクセスによるドキュメントの修正

以上で作成した XSP ファイルによる XML ドキュメントの修正が期待通り適正に行われていることが検証・確認できた^(*)、^(*)。

(*1)存在しないコレクションや登録されていないXMLドキュメントを修正対象とした場合のプログラムの動作について紹介しておく。XUpdateに関わるAPIの振る舞いの一端を知ることができて興味深い。

(1)存在しないコレクションを指定した場合:

コレクションとしてkyuukou202を指定する。xindice-xupdate-xsp22へのアクセス結果は次の様に期待した通りの不成功(failure)となる。プログラムの流れから例外(Exception)を生じていることが分かる。

```
- <page>
- <xupdate-xindice>
  <collection>kyuukou202</collection>
  <oid>kyu4</oid>
  <result>failure</result>
  <update-num>0</update-num>
</xupdate-xindice>
</page>
```

(2)登録されていないドキュメントを指定した場合:

コレクションをkyuukou20とする。xindice-xupdate-xsp22?oid=kyu10へのアクセスによりoid=kyu10を指定する。結果は次の様に成功(success)であるが,修正数は0となっている。

```

- <page>
- <xupdate-xindice>
  <collection>kyuukou20</collection>
  <oid>kyu10</oid>
  <result>success</result>
  <update-num>0</update-num>
</xupdate-xindice>
</page>

```

修正数 = 0 は当然として、例外が生じて不成功となることが期待されるのであるが、“成功”が示されて例外がキャッチされていないことが分かる。修正数が 0 の場合には不成功を示すようにコードを書き換えることも可能であるが、“修正内容が何も無い修正”の実行もあり得ることを考慮すれば、コードはこのままで差し支えないであろう。だとすれば、成功、不成功にかかわらず、この時点では結果として“何も修正されなかった”とのみ解釈するしかない。

この間の事情については、XSP ファイル作成に当って雛型とした Groovy スクリプト xindice_xupdate_embed.groovy による修正の実行が一定の示唆を与えてくれる。

上記の(1)について：

groovy xindice_xupdate_embed kyuukou202 kyu4 xindice-update24.xml を実行する。結果は以下の通りである。

```

C:¥MyNote¥Xindice>groovy xindice_xupdate_embed kyuukou202 kyu4 xindice-update24.xml
trying to register database
2005/10/17 9:34:40 org.apache.xindice.client.xmlldb.embed.DatabaseImpl loadConfiguration
情報: Specified configuration file: '/Tomcat5.0/webapps/cocoon/WEB-INF/xindice.xml/'
2005/10/17 9:34:40 org.apache.xindice.core.Database setConfig
情報: Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥¥db
Caught: java.lang.NullPointerException: Cannot invoke method: getService on null object
at
xindice_xupdate_embed.run(C:¥MyNote¥Xindice¥xindice_xupdate_embed.groovy:128)
at xindice_xupdate_embed.main(C:¥MyNote¥Xindice¥xindice_xupdate_embed.groovy)
C:¥MyNote¥Xindice>

```

これによれば、
col = DatabaseManager.getCollection(uri) ;
でコレクションが存在しないことから col=null となり
col.getService("XUpdateQueryService","1.0") ;
で NullPointerException を生じていることがうかがえる。

上記の(2)について：

groovy xindice_xupdate_embed kyuukou20 kyu10 xindice-update24.xml を実行する。結果を以下に示す（内容が余りにも多いので便宜上途中の経過を省略している）

```

C:¥MyNote¥Xindice>groovy xindice_xupdate_embed kyuukou20 kyu10 xindice-update24.xml
trying to register database
2005/10/17 9:43:27 org.apache.xindice.client.xmlldb.embed.DatabaseImpl loadConfiguration
情報: Specified configuration file: '/Tomcat5.0/webapps/cocoon/WEB-INF/xindice.xml/'
2005/10/17 9:43:28 org.apache.xindice.core.Database setConfig
情報: Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥¥db
2005/10/17 9:43:28 org.apache.xindice.core.Collection queryDocument
===== 中 略 =====
2005/10/17 9:43:28 org.apache.xindice.core.xupdate.XUpdateQueryResolver$XUpdateQuery
execute
警告: ignored exception
java.lang.NullPointerException
at org.apache.xindice.core.xupdate.XUpdateQueryResolver$XUpdateQuery.exe

```

```
cute(XUpdateQueryResolver.java:142)
    at org.apache.xindice.core.xupdate.XUpdateQueryResolver.query(XUpdateQueryR
esolver.java:84)
===== 中 略 =====
xmldb:xindice-embed:///db/kyuukou20 : doc-id = kyu10
documents updated : 0
```

注目したいのは次の 2 行である。

警告: ignored exception

java.lang.NullPointerException

即ち, XUpdate の実行

```
num = service.updateResource(oid,update_data);
```

において NullPointerException が無視され延々と関連する API の呼び出しが続けられ,結果として

documents updated : 0

が与えられている。XSP ファイルによる修正実行で修正数=0 であるが,“成功”が示される訳である。

序ながら,「 コマンドラインツール xindice による Embedded モードデータベースへのアクセス」の項の(*2)で触れたことと関連して,コレクション addressbook20 の address3 と登録されていない address2 ドキュメントについての Groovy スクリプトによる修正実行の様子も示しておく。address3 については,登録されているが故に修正は成功し, address2 については登録されていないが故に例外は無視されて結果として 0 修正となっている。

```
C:¥MyNote¥Xindice>groovy xindice_xupdate_embed addressbook20 address3 xindice-up
date01.xml
trying to register database
2005/10/18 6:08:44 org.apache.xindice.client.xmlldb.embed.DatabaseImpl loadConfiguration
情報: Specified configuration file: '/Tomcat5.0/webapps/cocoon/WEB-INF/xindice.xml/'
2005/10/18 6:08:44 org.apache.xindice.core.Database setConfig
情報: Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥¥db
2005/10/18 6:08:45 org.apache.xindice.core.Collection queryDocument
===== 中 略 =====
xmldb:xindice-embed:///db/addressbook20 : doc-id = address3
documents updated : 1
```

```
C:¥MyNote¥Xindice>groovy xindice_xupdate_embed addressbook20 address2 xindice-up
date01.xml
trying to register database
2005/10/18 6:08:08 org.apache.xindice.client.xmlldb.embed.DatabaseImpl loadConfiguration
情報: Specified configuration file: '/Tomcat5.0/webapps/cocoon/WEB-INF/xindice.xml/'
2005/10/18 6:08:08 org.apache.xindice.core.Database setConfig
情報: Database points to C:¥Tomcat5.0¥webapps¥cocoon¥WEB-INF¥¥db
2005/10/18 6:08:09 org.apache.xindice.core.Collection queryDocument
===== 中 略 =====
警告: ignored exception
java.lang.NullPointerException
===== 中 略 =====
xmldb:xindice-embed:///db/addressbook20 : doc-id = address2
documents updated : 0
```

(*2)折角ドキュメント修正のための XSP ファイルを作成したので,この XSP ファイルによって Standalone モード Xindice に登録されたドキュメントの修正が可能かどうかを確かめておく。XSP ファイルの内容を次の様に変更する(変更部分のみ,付録 E のコードを参照)。

```
String teacher = "伊井 天気";
```

```
String comment = "補講はいつかに";
```

```
String coll = "kyuukou";
```

```
String driver = "org.apache.xindice.client.xmlldb.DatabaseImpl"; // 変更しなくてもよい
```

String uri = "xmldb:xindice://localhost:8080/db/" + coll ;
 http://localhost:8080/cocoon/edu/edu20/xindice-xupdate-xsp22 にアクセスして修正を実行する
 (修正対象ドキュメントは kyu4)。図 24 と同様の<collection>kyuukou</collection>となった
 結果が得られ、修正は成功する。確認のため、http://localhost:8080/cocoon/edu/edu20/
 xindice-xpath02 (対象を ID='4'に設定)にアクセスした結果を示す(下図参照)。設定通りの修
 正がなされている。

休 講 通 知						
ID番号	日時	曜日	校時	担当教員	授業名	備考
4	1月26日	月曜	4校時	伊井 天気	情報環境論	補講はいつかに

Standalone モード, Embedded モード Xindice 何れにおいても, ドキュメントの登録とドキュメントの修正について, XSP ファイルを作成して用いることの有用性があらためて確かめられた。

Jetty Server 上での Cocoon2 における Xindice 登録ドキュメントの修正

これまではサーブレットコンテナである Tomcat 上で動作する Cocoon2 と Xindice を動作検証の対象としてきた。多くの結果が得られたが, XMLDBTransformer コンポーネントを用いた Xindice 登録 XML ドキュメントの修正については, 満足すべき結果が得られていない^(*)。

以下では, XMLDBTransformer の利用, とりわけドキュメントの修正 (XUpdate) に焦点を絞り, サーブレットコンテナ Jetty Server^(**)上で動作する Cocoon2 の Embedded モード Xindice を対象として動作の検証を行う。これまでの結果との比較を容易にし, 併せてこれまでの各種設定に誤りが無いことを再度確認する意味をも込めて, Cocoon2 には同じ Ver.2.1.6 のものを用い^(***), 動作検証に用いるドキュメントや各種 XML ファイル等々もこれまで用いたのと同内容のものを用いることとする。sitemap.xmap ファイルにおける設定についても同様である。

(*1)筆者の検証においてだけではなく, XMLDB サンプルブロックでのドキュメント修正サンプルも "failure" の結果を示している。一方 Jetty Server 上での Cocoon2 ブロックサンプルでは, 同じサンプルが "success" となっている。

(*2)<http://jetty.mortbay.org/>

(*3)本稿執筆時点でのバージョンは 2.1.7 である。筆者の環境下で動作を確認している。

Cocoon2 の作成と起動

Cocoon2 実行ファイルの作成

Cocoon2 のソースファイルを解凍して得られる cocoon-2.16 ディレクトリで build コマンド (war ファイルをも作成するときは build war コマンド) によりコンパイルを実行すれば, サーブレットとして実行可能な Cocoon2 が得られる^(*)。

Cocoon2 の起動

Cocoon2 を起動するには, cocoon-2.1.6 ディレクトリ内で cocoon servlet を実行する

(以下参照)

```
C:\cocoon-2.1.6>cocoon servlet
cocoon.bat: using .\build\webapp as the webapp directory
```

これにより別のコマンドプロンプトのウィンドウが開いて、エラーを生じることなく以下に示す最後の行が表示されれば Jetty Server がスタートしたこととなる。Xindice サーバが成功裏にスタートしていることも読み取れる(以下参照)。停止は ctrl+C を押下する。

```
----- Loading -----
Processing repository: .\tools\jetty\lib
Adding jar: .\tools\jetty\lib\jetty-4.2.19.jar
===== 中 略 =====
- Database points to C:\cocoon-2.1.6\build\webapp\WEB-INF\db
- Database 'db' successfully opened
- Xindice server successfully started
08:26:38.578 EVENT Started SocketListener on 0.0.0.0:8888
08:26:38.578 EVENT Started org.mortbay.jetty.Server@f84386
```

ブラウザにより <http://localhost:8888/> にアクセスして図 27 の画面が得られれば確かに Cocoon2 が動作している。

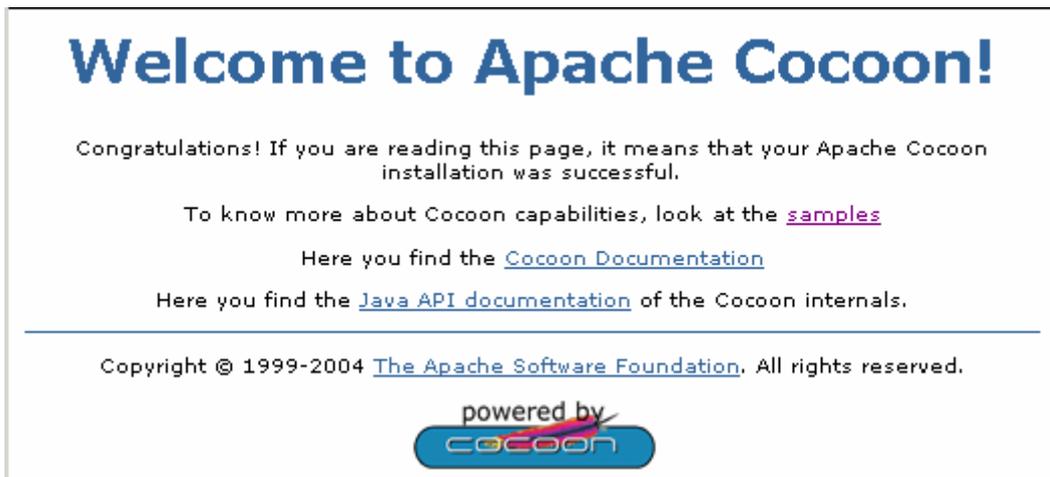


図 27 : ブラウザによる <http://localhost:8888/> へのアクセス画面

(*1) 「動作検証の実行環境」のところでも述べたように、j2sdk1.4.2_03 を用いて build した war ファイル並びに Jetty Server 上での Cocoon サブレットは何故か(筆者の環境では)動作しない。j2sdk1.4.0_03 で build した cocoon-2.1.6 ディレクトリをそっくりそのまま筆者の環境にコピーして、動作検証を行っている。尚、cocoon-2.1.7 は筆者の環境下で build して(多くの警告が出されるものの)、Jetty Server 上で問題なく動作している。

XMLDBTransformer 利用のための設定

edu20 ディレクトリのコピー

これまで用いてきた edu20 ディレクトリをそのまま cocoon-2.16\build\webapp ディレクトリ内にコピーする。sitemap.xml ファイルに記述されたパイプライン設定や

関連ファイルを再利用するためである。

sitemap.xmap への追加記述

cocoon-2.1.6¥build¥webapp¥sitemap.xmap に edu20 へのマウント設定を次の様に記述する。これにより，http://localhost:8888/edu20/*へのアクセスが可能となる。

尚，sitemap.xmap における XMLDBTransformer に関わる設定はデフォルトのままとする(*1)。

sitemap.xmap ファイルへの追加記述：

```
<map:match pattern="edu2*/**">
  <map:mount uri-prefix="edu2{1}" src="edu2{1}/" check-reload="yes"/>
</map:match>
```

(*1)デフォルトの設定下でも，Tomcat 上で動作する Cocoon2 の XMLDB ブロックサンプルにおけるドキュメント修正の不具合は変わらない。

XMLDBTransformer によるコレクションの生成，ドキュメントの登録，削除，修正

コレクションの生成

Tomcat 上での Cocoon2 において Embedded モード Xindice を対象としたのと同様に tr-create-coll21，tr-create-coll22 へのアクセスにより，コレクション addressbook20 と kyuukou20 を作成する（図 28，図 29 参照）。

```
- <page>
  <x:query oid="addressbook20" type="create" result="success"/>
</page>
```

図 28：コレクション addressbook20 の作成

```
- <page>
  <x:query oid="kyuukou20" type="create" result="success"/>
</page>
```

図 29：コレクション kyuukou20 の作成

XML ドキュメントの登録

tr-add-doc21，tr-add-doc22 へのアクセスにより addressbook20 内に address1～address3 ドキュメント，kyuukou20 内には kyu1～kyu4 ドキュメントを登録する。tr-test21.xml，tr-test2.xml の内容を適宜書き換えながらアクセスを実行すればよい（図 30）。kyuukou20 コレクション内の各ドキュメントは図 20 に示されたのと同じ状態のはずである。

```
- <page>
  <x:query oid="address1" type="create" result="success"/>
</page>
```

図 30：ドキュメント address1 の登録結果

XML ドキュメントの削除

tr-del-doc21 へのアクセスにより，コレクション addressbook20 内の address2 ドキュメントを削除する（図 31 参照）。結果は “ success ” である。

```
- <page>
  <x:query oid="address2" type="delete" result="success"/>
</page>
```

図 31：address2 ドキュメントの削除結果

XML ドキュメントの修正

修正その 1

まず address3 ドキュメントを対象として XUpdate を実行する。修正内容を記述したファイルは先に用いた tr-update-doc21.xml で，addressbook20 内のドキュメント address3 において自宅のメールアドレス要素を削除するものである。対応する URL である tr-update-doc21 にアクセスする。結果は，これまでの例えば図 12 に示されているものとは異なり，次の図 32 に示すように成功 “ success ” となる。

```
- <page>
  <x:query oid="address3" type="update" result="success">1 entries updated.</x:query>
</page>
```

図 32：address3 ドキュメントの修正結果

確かに属性が home である email 要素が削除されていることが分かる（図 33 参照）

```
- <db:results query="/person[fname='Richard']" resources="1">
  - <db:result docid="address3">
    - <person src:col="/db/addressbook20"
      src:key="address3">
      <fname>Richard</fname>
      <lname>Smith</lname>
      <phone type="work">123-456-1234</phone>
      <phone type="home">456-789-4321</phone>
      <email type="work">richard@mailserv.com</email>
      <address type="work">88,Some City</address>
      <address type="home">55,Abce City</address>
    </person>
  </db:result>
</db:results>
```

図 33：修正実行後の address3 ドキュメントの内容

修正その 2

コレクション kyoukou20 内の kyu4 ドキュメントを対象とする。修正内容を記述したファイルは tr-update-doc22.xml で，修正の内容それ自体は，これまでに xindice コ

マンドツールによる kyu4 ドキュメント修正に用いたことのある xindice-update02.xml ファイルと同じである（以下参照）。xindice コマンドでは漢字の取り扱いが UTF-8 エンコーディングの場合に限られていたが、XMLDNTTransformer の漢字エンコーディング対応の様子を知る上でも都合のよい例である。

tr-update-doc22.xml ファイル内容；

```
<?xml version="1.0" encoding="Windows-31J"?>
<page>
  <x:query type="update" collection="kyuukou20" oid="kyu4" xmlns:x="http://apache.org/cocoon/xmlldb/1.0">
    <xu:modifications version="1.0" xmlns:xu="http://www.xmlldb.org/xupdate">
      <xu:remove select="/休講/備考"/>
      <xu:update select="/休講/担当教員">基盤 専多</xu:update>
      <xu:insert-after select="/休講/授業名">
        <xu:element name="備考">都合により補講は休講とする</xu:element>
      </xu:insert-after>
    </xu:modifications>
  </x:query>
</page>
```

対応する URL:tr-update-doc22 にアクセスした結果を示しているのが図 34 である。修正結果は、address3 ドキュメントの修正と同様に成功 “ success ” である。

```
- <page>
  <x:query oid="kyu4" type="update" result="success">1 entries updated.</x:query>
</page>
```

図 34：kyu4 ドキュメントの修正結果

漢字のエンコーディングへの対応状況をも確認のため、ドキュメントの内容を示したのが図 35 である。エンコーディング “ Windows-31J ” に対応していることが分かる。

以上により、Jetty Server 上で動作する Cocoon2 においては XMLDBTransformer を用いた XML ドキュメントの修正(XUpdate)が正常に動作することが確かめられた。

休 講 通 知						
ID番号	日時	曜日	校時	担当教員	授業名	備考
4	1月26日	月曜	4校時	基盤 専多	情報環境論	都合により補講は休講とする

図 35：修正実行後の kyu4 ドキュメントの内容

存在しないコレクションを対象とした修正

tr-update-doc22.xml ファイルを修正して kyuukou20 を存在しない kyuukou202 に変更する。oid=kyu4 はそのままとする。tr-update-doc22 へのアクセス結果は図 36 に示す通り、期待通りのものとなっている。

```

- <page>
- <x:query oid="kyu4" type="update" result="failure">
  Failed to update resource kyu4: Collection xmldb:xindice-embed:///db/kyuukou202 not found.
</x:query>
</page>

```

図 36 : コレクション kyuukou202 を対象とした XUpdate の実行結果

存在しないドキュメントを対象とした修正

tr-update-doc22.xml ファイルを再度修正して oid=kyu10 とし ,コレクションについては collection="kyuukou20"とする。tr-update-doc22 へのアクセス結果は図 36 に示す様に , 結果は成功 “ success ” , 修正数は 0 である。

```

- <page>
  <x:query oid="kyu10" type="update" result="success">0 entries updated.</x:query>
</page>

```

図 37 : kyu10 ドキュメントを対象とした XUpdate の結果

尚このとき , コマンドプロンプトの画面には次の内容が表示された (以下参照) 。ここでは ,

ドキュメントが存在しないことによる例外が無視されている (太字部分) (*1)。

```

- [PoolThread-4] '/db/kyuukou20' Query document kyu10, query:
  <xu:modifications xmlns:x="http://apache.org/cocoon/xmldb/1.0" xmlns:xu="http://www.xmlldb.org/xupdate" version="1.0">
    <xu:remove select="/休講/備考"/>
    <xu:update select="/休講/担当教員">基盤 専多</xu:update>
    <xu:insert-after select="/休講/授業名">
      <xu:element name="備考">都合により補講は休講とする</xu:element>
    </xu:insert-after>
  </xu:modifications>

- ignored exception
  = = = = = 以下 略 = = = = =

```

(*1)図 36 , 図 37 を含むこれらの結果は , Embedded モードの Xindice を対象とした「 XSP ファイルでのプログラミング」の項の注(*1)で述べた内容と一致するもので , Jetty Server 上の Cocoon2 では , XMLDBTransformer の振る舞いが Groovy スクリプトや XSP プログラムにおける XUpdate 実行時の振る舞いと全く同じであることを示している。

XMLDBTransformer による Standalone モード Xindice ドキュメントの修正

これまでの検証により , XMLDBTransformer による Embedded モード Xindice に登録されたドキュメントの修正 (XUpdate) は成功 (success) することが確認できた。以下では更に一步踏み込んで , Tomcat 上で動作する Standalone モード Xindice に登録されたドキュメント修正を , Jetty Server 上の Cocoon2 から XMLDBTransformer を用いて試みる。この修正は , Tomcat 上の Cocoon2 からでは不具合に終わっている例である。

cocoon.xconf ファイルへの追加記述

「 Cocoon2 における Standalone モード Xindice の利用 」の項で説明した様に , cocoon.xconf ファイルに , Pseudo Protocol である **xmlldb:xindice:**が利用できる以下の内容を追加記述する。

```
<!-- 2005.10.22 -->
<driver class="org.apache.xindice.client.xmlldb.DatabaseImpl" type="xindice">
  <!-- Specifies path to the Xindice configuration file.
        If missing, buit-in configuration defaults will be used. -->
  <configuration>/Tomcat5.0/webapps/xindice/WEB-INF/system.xml</configuration>
  <!-- Specifies path to the Xindice DB home directory.
        Default is `pwd` (current working directory).
        -->
  <db-home>/Tomcat5.0/webapps/xindice/WEB-INF/db</db-home>
  <managed>true</managed>
</driver>
<!-- 2005.10.22 -->
```

sitemap.xmap ファイルへの追加記述

これまでの修正実行と同様に , コレクション addressbook (addressbook20 ではないことに注意) 内の address3 ドキュメントとコレクション kyuukou 内の kyu4 ドキュメントを修正の対象とする。sitemap.xmap ファイルに以下を追加記述する (太字部分に注意)

```
<map:match pattern="tr-update-doc31">
  <map:generate src="tr-update-doc31.xml"/>
  <map:transform type="xmlldb">
    <map:parameter name="base" value="xmlldb:xindice://localhost:8080/db"/>
  </map:transform>
  <map:serialize type="xml" />
</map:match>
<map:match pattern="tr-update-doc32">
  <map:generate src="tr-update-doc32.xml"/>
  <map:transform type="xmlldb">
    <map:parameter name="base" value="xmlldb:xindice://localhost:8080/db"/>
  </map:transform>
  <map:serialize type="xml" />
</map:match>
```

上記で , tr-update-doc31(2).xml ファイルは tr-update-doc01(2).xml と内容は略同様であるが , 後ほどの確認のため修正内容を次の様にしておく。

tr-update-doc31.xml : 職場のメールアドレス要素を削除する

tr-update-doc32.xml : 備考欄を削除 , 担当教員の名前を “ 十月 秋晴 ” に変更 , 備考欄を追加 (内容は , “ 特になし ”) する

修正 (XUpdate) の実行と修正内容の確認

はじめに address3 , koukou4 ドキュメントを登録し直しておく。ドキュメントの内容は付録 A に示した通りである。

address3 の修正

http://localhost:8888/edu20/tr-update-doc31 にアクセスして修正を実行する。結果は図 38 に示す通りで、修正は成功 (success) となっている^(*)。

```
- <page>
  <x:query oid="address3" type="update" result="success">1 entries updated.</x:query>
</page>
```

図 38 : address3 ドキュメントの修正結果

http://localhost:8080/cocoon/edu/edu20/xindice-xpath01 (address3 ドキュメントを検索・表示するよう適宜変更) にアクセスして、修正内容を Tomcat 上の Cocoon2 から確認する(図 39)。図 33 と比較して、今回は職場のメールアドレス(属性 type='work')要素が削除されていることが分かる。

```
- <db:results query="/person[fname=Richard]" resources="1">
- <db:result docid="address3">
- <person src:col="/db/addressbook" src:key="address3">
  <fname>Richard</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-1234</phone>
  <phone type="home">456-789-4321</phone>
  <email type="home">richard@hij-mail.com</email>
  <address type="work">88,Some City</address>
  <address type="home">55,Abce City</address>
</person>
</db:result>
</db:results>
```

図 39 : 修正実行後の address3 ドキュメントの内容

kyoukou4 の修正

http://localhost:8888/edu20/tr-update-doc32 にアクセスして kyuukou4 ドキュメントの修正を実行する。得られる結果は図 34 と同様で、修正は成功である。

図 35 同様の結果が得られるかどうかを Tomcat 上の Cocoon2 から確認する。アクセスする URL は http://localhost:8080/cocoon/edu/edu20/xindice-xpath02 (kyu4 ドキュメントを検索・表示するよう適宜変更) で、修正実行後のドキュメント内容を図 40 に示す。設定通りの修正がなされていることが分かる。

休 講 通 知						
ID番号	日時	曜日	校時	担当教員	授業名	備考
4	1月26日	月曜	4校時	十月 秋晴	情報環境論	特になし

図 40 : 修正実行後の kyu4 ドキュメントの内容

以上から、Jetty Server 上で動作する Cocoon2 での XMLDBTransformer 利用によ

る XML ドキュメントの修正 (XUpdate) が, Embedded モード Xindice のみならず Tomcat 上で動作する Xindice をも対象とし得ることが確かめられた。

(*1)再度同じ修正を実行すると,
<x:query oid="address3" type="update" result="success">0 entries updated.</x:query>
なる結果が得られる。

xindice コマンドツールによる Embedded モード Xindice ドキュメントの修正

残された問題が1つある。Tomcat 上で動作している Xindice 付属の xindice コマンドツールを用いて, Jetty Server 上の Cocoon2 に Embedded されている Xindice に登録の XML ドキュメント修正を検証する。この xindice コマンドツールによるドキュメント修正は, Tomcat 上の Cocoon2 に対しては不成功となっている。

Configuration ファイルの作成

xindice が参照する Configuration ファイルを作成する (「Cocoon2-Embedded モード Xindice の利用」の項を参照)。%XINDICE_HOME%内の system.xml ファイルを同じディレクトリ内に system-j-embed.xml ファイルとしてコピーし, 次の内容とする (以下参照, 関連部分のみを示す。変更等は太字部分)。

system-j-embed.xml ファイルの変更内容 (関連部分):

<pre><!--root-collection dbroot="./db/" name="db" use-metadata="on"--></pre>	コメントアウト
<pre><root-collection dbroot="/cocoon-2.1.6/build/webapp/WEB-INF/db/" name="db" use-metadata="on"></pre>	追加記述

コレクションの検索

コレクション addressbook20 に登録されているドキュメントをリストアップしてみる。address2 ドキュメントは削除されているので address1 と address3 が登録されているはずである。次の様に実行する (以下参照)。Configuration ファイルが参照され, アクセス対象のデータベースが Jetty Server 上で動作する Cocoon2-Embedded モード Xindice であることが示されている。

<pre>C:¥MyNote¥Xindice>xindice ld -c xmldb:xindice-embed:///db/addressbook20 -d %XINDICE_HOME%¥system-j-embed.xml trying to register database [INFO] DatabaseImpl - -Specified configuration file: 'c:¥Tomcat5.0¥webapps¥xindice¥WEB-INF¥system-j-embed.xml' [INFO] Database - -Database points to c:¥cocoon-2.1.6¥build¥webapp¥WEB-INF¥db address1 address3 Total documents: 2</pre>
--

address3 ドキュメントの修正

修正内容を記したフィルとしてはこれまでと同じ xindice-update01.xml を用いるが, address3 ドキュメントでは先の修正で自宅のメールアドレスが削除され図 33 に示され

る内容となっていることから、「自宅のメールアドレス削除」を「職場のメールアドレスを削除」に一時変更する。「自宅電話番号の変更」と「自宅メールアドレスの追加」はそのままである。

修正を実行する。結果は次に示すように、Tomcat 上で動作する Cocoon2-Embedded モード Xindice に対すると同様例外の発生により修正は行われぬ(以下参照)^(*1)。

```
C:\MyNote\Xindice>xindice xupdate -c xmldb:xindice-embed:///db/addressbook20 -n
address3 -f ./xindice-update01.xml -d %XINDICE_HOME%\system-j-embed.xml
trying to register database
[INFO] DatabaseImpl - -Specified configuration file: 'c:\Tomcat5.0\webapps\xindi
ce\WEB-INF\system-j-embed.xml'
[INFO] Database - -Database points to c:\cocoon-2.1.6\build\webapp\WEB-INF\db
[INFO] Collection - -[main] '/db/addressbook20' Query document address3, query:
===== 中 略 =====
Exception in thread "main" javax.xml.parsers.FactoryConfigurationException: java.lan
g.ClassNotFoundException: org/apache/xindice/core/xupdate/XPathQueryFactoryImpl
    at org.xmldb.common.xml.queries.XPathQueryFactory.newInstance(Unknown
n Source)
===== 以下 略 =====
```

(*1)ドキュメントの修正が例外の発生により不首尾であったので、以下のコマンドを実行してみた。表示された結果は次の様で修正が首尾よく行われた旨の表示が示された(以下参照)

```
C:\MyNote\Xindice>xindice xupdate -c xmldb:xindice://localhost:8888/db/addressbook20 -n
address3 -f ./xindice-update01.xml -d %XINDICE_HOME%\system-j-embed.xml
trying to register database
1 documents updated
```

address3 の内容を表示してみると(<http://localhost:8888/edu20/xindice-xpath21> にアクセス), 図 33 同様の結果が得られ、確かに職場でのメールアドレスが削除されており、また自宅の電話番号も変更、更に自宅のメールアドレスも追加されている(下図参照)。ただし、追加された要素はこれまでの修正結果とは異なり、<temporaryXUpdateTree>タグで囲まれた子要素となっている。これでは、自宅メールアドレス要素の以後の取り扱いに統一性を欠くこととなり、あまり好ましいことではない。

```
- <db:results query="/person[fname='Richard']" resources="1">
- <db:result docid="address3">
- <person src:col="/db/addressbook20" src:key="address3">
  <fname>Richard</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-1234</phone>
  <phone type="home">321-654-4321</phone>
  <address type="work">88,Some City</address>
  <address type="home">55,Abce City</address>
- <temporaryXUpdateTree>
  <email type="home">smith@mail-efg.com</email>
  </temporaryXUpdateTree>
</person>
</db:result>
</db:results>
```

おわりに

XML ネイティブなデータベース Xindice について、Cocoon2 からのアクセスを中心として動作検証を行いその結果を紹介した。Tomcat 上で動作している Cocoon2 を用い

た Web アプリケーションでの Xindice の利用に資することができればと考えたのが動機である。

文献も乏しく試行錯誤の連続ではあったが、紹介した事項については全てにわたって逐一実際に動作を確認している。筆者の誤解に基づく設定の不備などによる思わぬミスがあることを恐れるが、気が付いた点をご指摘いただければありがたい。

本稿でも何度となく触れていることであるが、Tomcat 上で動作する Cocoon2 での XMLDBTransformer による XML ドキュメント修正については、まだ問題点を整理し得ていない。今後に残された課題としたい。

本稿ではスクリプト言語 Groovy によるスクリプトコードを紹介している。つい最近になって知ったばかりの不慣れな状況下でのスクリプト作成であるが、XSP ファイル作成にあたってのプログラムの動作検証などに、スクリプト言語の手軽さから随分と重宝した。敢えて紹介した次第である。

本稿の内容が読者にとって Xindice をはじめとする XML データベースに興味をもつ契機となれば幸いである。

参考文献

- (1)青木 昌三：「Java による Web アプリ開発の勧め（ ）
- Tomcat と Cocoon2 利用環境の構築 - 」
香川大学総合情報基盤センター 年報 第1号，pp.45-65（2004年2月）
青木 昌三：「Java による Web アプリ開発の勧め（ ）
- Cocoon2 利用による Web ページ作成 - 」
香川大学総合情報基盤センター 年報 第1号，pp.66-90（2004年2月）
- (2)青木 昌三，宮崎 英一：「携帯電話端末に対応した休講通知掲示システムの開発」
香川大学教育実践総合研究 第9号，pp.15-24（2004年9月）
- (3)吉田 晃伸，大田 純：「はじめての XQuery プログラミング 第1回～」
JavaWorld 2004 September，アイ・ディ・ジー・ジャパン
- (4)青木 昌三：「PPP 接続ログ集計スクリプトの作成 - スクリプト言語 Groovy を用いて - 」
香川大学総合情報基盤センター 年報 第3号（2006年2月）
- (5)Arun Gaikwad：「Introduction to Xindice」
<http://www-106.ibm.com/developerworks/web/library/wa-xindice.html>

〔付録〕

付録 A : コレクションへの登録に用いた XML 文書

コレクション addressbook への登録ドキュメント :

address1.xml

```
<?xml version="1.0" ?>
<person>
  <fname>John</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-7890</phone>
  <phone type="home">123-456-0987</phone>
  <email type="work">john@somemail.com</email>
  <email type="home">smith@abc-mail.com</email>
  <address type="work">25,Work City</address>
  <address type="home">30,Home City</address>
</person>
```

address2.xml

```
<?xml version="1.0" ?>
<person>
  <fname>Robert</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-1234</phone>
  <phone type="home">123-456-4321</phone>
  <email type="work">robert@mailserv.com</email>
  <email type="home">smith@efg-mail.com</email>
  <address type="work">88,Some City</address>
  <address type="home">22,Abcd City</address>
</person>
```

address3.xml

```
<?xml version="1.0" ?>
<person>
  <fname>Richard</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-1234</phone>
  <phone type="home">456-789-4321</phone>
  <email type="work">richard@mailserv.com</email>
  <email type="home">richard@hij-mail.com</email>
  <address type="work">88,Some City</address>
  <address type="home">55,Abce City</address>
</person>
```

コレクション kyuukou への登録ドキュメント :

kyu1.xml

```
<?xml version="1.0" encoding="Windows-31J" ?>
<!-- Author: M.Aoki "aoki@ed.kagawa-u.ac.jp" -->
<休講>
  <ID>1</ID>
  <日付コード>2006011910</日付コード>
  <日時>1月19日</日時>
  <曜日>月曜</曜日>
  <校時>1校時</校時>
  <担当教員>青木 昌三</担当教員>
  <授業名>物理学概論1</授業名>
  <備考>後日補講を実施する</備考>
</休講>
```

kyu2.xml

```
<?xml version="1.0" encoding="Windows-31J" ?>
<!-- Author: M.Aoki "aoki@ed.kagawa-u.ac.jp" -->
<休講>
  <ID>2</ID>
  <日付コード>2006012020</日付コード>
  <日時>1月20日</日時>
  <曜日>火曜</曜日>
  <校時>2校時</校時>
  <担当教員>香川 太郎</担当教員>
  <授業名>香川大学の将来について</授業名>
  <備考>*</備考>
</休講>
```

kyu3.xml

```
<?xml version="1.0" encoding="Windows-31J" ?>
<!-- Author: M.Aoki "aoki@ed.kagawa-u.ac.jp" -->
<休講>
  <ID>3</ID>
  <日付コード>2006012330</日付コード>
  <日時>1月23日</日時>
  <曜日>金曜</曜日>
  <校時>3校時</校時>
  <担当教員>情報 次郎</担当教員>
  <授業名>情報学基礎</授業名>
  <備考>補講を実施</備考>
</休講>
```

kyu4.xml

```
<?xml version="1.0" encoding="Windows-31J" ?>
<!-- Author: M.Aoki "aoki@ed.kagawa-u.ac.jp" -->
<休講>
  <ID>4</ID>
  <日付コード>2006012640</日付コード>
  <日時>1月26日</日時>
  <曜日>月曜</曜日>
  <校時>4校時</校時>
  <担当教員>基盤 仙太</担当教員>
  <授業名>情報環境論</授業名>
  <備考>補講は行わない</備考>
</休講>
```

付録 B : Xindice にアクセスする Groovy スクリプト(*)

xindice_xpath.groovy :

```
// XMLDB Sample xindice_xpath.groovy 2005.09.28
// Usage: groovy xindice_xpath[.groovy] collection xpath

//import org.apache.xindice.*
import org.xmldb.api.base.*
import org.xmldb.api.modules.*
import org.xmldb.api.*

// 関数 usage() 定義
def usage() {
  println ""
```

```

println "  利用法: groovy xindice_xpath[.groovy] collection xpath"
println ""
}
//関数定義終わり
if (args.length <=1 ){
    usage()
    System.exit(0)
}

Collection col = null
driver = "org.apache.xindice.client.xmldb.DatabaseImpl"
c = Class.forName(driver)
database = c.newInstance()
DatabaseManager.registerDatabase(database)

uri = "xmldb:xindice://localhost:8080/db/" + args[0]
xpath = args[1]
if (! xpath.startsWith("/") && ! xpath.startsWith("//"))
    xpath = "/" + args[1]

col = DatabaseManager.getCollection(uri)
service = col.getService("XPathQueryService","1.0")
resultset = service.query(xpath)           // 下記の(*)を参照
results = resultset.getIterator()

while (results.hasMoreResources()){
    res = results.nextResource()
    println res.getContent()
}
if (col != null)
    col.close()

```

(*)service.query(xpath)に代えて service.queryResource(oid,xpath)とすることにより，コレクション内の特定の oid 値(キー値)をもつ XML ドキュメントを検索の対象とすることができる。oid 値をコマンド引数として指定することも 1 つの方法である。

付録 C : Xindice での XML ドキュメントを修正する Groovy スクリプト(*)

xindice_update.groovy :

```

// XMLDB Sample xindice_update.groovy  2005.10.07
// Usage: groovy xindice_update[.groovy] collection xupdate-xml-file

//import org.apache.xindice.*
import org.xmldb.api.base.*
import org.xmldb.api.modules.*
import org.xmldb.api.*

import java.io.File

// 関数定義
def usage() {
    println ""
    println "  利用法: groovy xindice_update[.groovy] collection xupdate-xml-file"
    println ""
}

```

```

def getEncode(line){
  if (! line.contains("encoding"))
    return "UTF-8"
  enc = "JISAutoDetect"
  line_l = line.toLowerCase()
  switch (line_l) {
    case ~".*utf-8.*" : enc = "UTF-8" ; break
    case ~".*windows-31j.*" : enc = "Windows-31J" ; break
    case ~".*shift_jis.*" : enc = "Windows-31J" ; break
    case ~".*euc.*" : enc = "EUC_JP" ; break
    case ~".*iso-2022-jp.*" : enc = "ISO-2022-JP" ; break
    case ~".*iso2022jp.*" : enc = "ISO-2022-JP" ; break
    case ~".*unicode.*" : enc = "Unicode" ; break

    default : break
  }
  return enc
}
def printCommentError(fname){
  println ""
  println "    ${fname}ファイルでのコメント文が不完全です"
  println "    <!-- に対応する --> がありません"
  println ""
  System.exit(0)
}
def printCommentError0(){
  println ""
  println "    XUpdate を記述したファイルでのコメント文が不完全です"
  println "    コメント文がネストしています"
  println ""
  System.exit(0)
}
def removeComment(st){
  i = st.indexOf("<!--")
  j = st.indexOf("-->") + 3
  k = st.indexOf("<!--",i+4)
  if ( k== -1 || k > j) {
    sb = new StringBuffer(st)
    return sb.delete(i,j).toString()
  }
  else
    printCommentError0()
}

//関数定義終わり
//ステートメントの始まり
if (args.length <=1 ){
  usage()
  System.exit(0)
}

encode = "JISAutoDetect"
xml_file_name = args[1]

```

```

new File(xml_file_name).eachLine {
  if (it.contains("<?xml"))
    encode = getEncode(it)
}

stbuff = new StringBuffer()
reader = new File(xml_file_name).newReader(encode) //BufferedReader
read_in = false
reader.eachLine { line |
  if(read_in)
    stbuff.append(line)
  if (line =~ "<[a-z][a-z]*:. *modifications") {
    stbuff.append(line)
    read_in = true
  }
  else if (line =~ "</[a-z][a-z]*:. *modifications")
    read_in = false
}
xupdate = stbuff.toString()
while ( true ){
  s = xupdate
  if (! s.contains("<!--"))
    break
  if (! s.contains("-->"))
    printCommentError(xml_file_name)

  xupdate = removeComment(s)
}

Collection col = null
driver = "org.apache.xindice.client.xmldb.DatabaseImpl"
c = Class.forName(driver)
database = c.newInstance()
DatabaseManager.registerDatabase(database)

uri = "xmldb:xindice://localhost:8080/db/" + args[0]

col = DatabaseManager.getCollection(uri)
service = col.getService("XUpdateQueryService","1.0")
service.update(xupdate) // 下記の(*)を参照
if (col != null)
  col.close()
println ""
println "  ${uri}"
println "  documents updated"
println ""

```

(*)service.update(xupdate)に代えて service.updateResource(oid,xupdate)とすることにより、特定の oid 値 (キー値) をもつ XML ドキュメントを修正の対象とすることができる。スクリプト作成にあたって、oid 値をコマンド引数として指定することなどが考えられる。尚、service.update() コマンド等の戻り値は long 値で修正数である。

付録 D : Xindice に XML ドキュメントを登録する XSP コード(*)

xindice-adddoc-01.xsp :

```

<?xml version="1.0" encoding="Windows-31J"?>
<!-- <?xml version="1.0" encoding="Shift_JIS"?> -->
<!-- Author: M.Aoki "aoki@ed.kagawa-u.ac.jp" -->
<xsp:page language="java"
    xmlns:xsp="http://apache.org/xsp"
    xmlns:xsp-request="http://apache.org/xsp/request/2.0"
    xmlns:sendmail="http://apache.org/cocoon/sendmail/1.0">

<xsp:structure>
  <!--xsp:include>org.apache.xindice.*</xsp:include-->
  <xsp:include>org.xmldb.api.base.*</xsp:include>
  <xsp:include>org.xmldb.api.modules.*</xsp:include>
  <xsp:include>org.xmldb.api.*</xsp:include>
  <xsp:include>java.io.*</xsp:include>
</xsp:structure>
<page>
<xsp:logic>
// Request Data from HTML-FORM
// For Example
// String s55 = request.getParameter("weekday");
// String week_day = new String(s55.getBytes("ISO8859_1"),"UTF-8");

String id = "5";
String date_code = "2006012810";
String date = "1 月 28 日";
String week_day = "水曜";
String class_hour = "1 校時";
String teacher = "自然 教学";
String class_name = "自然科学を如何に学ぶか";
String comment = "補講日は追って通知する";
//
// Output to XML-Document
String s0 = "&lt;?xml version=¥"1.0¥" encoding=¥"Windows-31J¥"?>";
String s1 = " &lt;休講>";
String s2 = " &lt;ID>" + id + "&lt;/ID>" ;
String s3 = " &lt;日付コード>" + date_code + "&lt;/日付コード>" ;
String s4 = " &lt;日時>" + date + "&lt;/日時>" ;
String s5 = " &lt;曜日>" + week_day + "&lt;/曜日>" ;
String s6 = " &lt;校時>" + class_hour + "&lt;/校時>" ;
String s7 = " &lt;担当教員>" + teacher + "&lt;/担当教員>" ;
String s8 = " &lt;授業名>" + class_name + "&lt;/授業名>" ;
String s9 = " &lt;備考>" + comment + "&lt;/備考>" ;
String s1_1 = " &lt;/休講>";

String result_xml = "";
String f_dir = "c:/tomcat5.0/webapps/cocoon/edu/edu20/xml2/";
String f_name = f_dir + "kyu" + id + ".xml";
String enc="Windows-31J";
BufferedWriter b_writer=null;
PrintWriter p_writer=null;

try {
result_xml = "success";
FileOutputStream fob = new FileOutputStream(f_name,false);

```

```

OutputStreamWriter osb_writer= new OutputStreamWriter(fob,enc) ;
b_writer = new BufferedWriter(osb_writer) ;
p_writer = new PrintWriter(b_writer) ;
    p_writer.println(s0) ; p_writer.println(s1) ; p_writer.println(s2) ;
    p_writer.println(s3) ; p_writer.println(s4) ; p_writer.println(s5) ;
    p_writer.println(s6) ; p_writer.println(s7) ; p_writer.println(s8) ;
    p_writer.println(s9) ; p_writer.println(s1_1) ;
} catch (Exception e) { result_xml = "failure" ; }
finally {
    p_writer.close() ;
}
//
//Store to Xindice
StringBuffer bf = new StringBuffer(s1+"¥r¥n") ;
    bf.append(s2+"¥r¥n")
    .append(s3+"¥r¥n")
    .append(s4+"¥r¥n")
    .append(s5+"¥r¥n")
    .append(s6+"¥r¥n")
    .append(s7+"¥r¥n")
    .append(s8+"¥r¥n")
    .append(s9+"¥r¥n")
    .append(s1_1) ;
String xml_data = bf.toString() ;

String oid = "kyu" + id ;
String coll = "kyuukou" ;
Collection col = null ;
String result_xindice = "" ;

try {
String driver = "org.apache.xindice.client.xmldb.DatabaseImpl" ;
Class c = Class.forName(driver) ;
Database database = (Database) c.newInstance() ;
DatabaseManager.registerDatabase(database) ;

String uri = "xmldb:xindice://localhost:8080/db/" + coll ;
col = DatabaseManager.getCollection(uri) ;

XMLResource doc = (XMLResource) col.createResource(oid,"XMLResource") ;// 下
記の(*)を参照
doc.setContent(xml_data) ;
col.storeResource(doc) ;
result_xindice = "success" ;
} catch (Exception e){ result_xindice = "failure" ; }
finally {
    try {
        if (col != null)
            col.close() ;
    } catch(Exception e){;}
}
//
</xsp:logic>
<to-xml>

```

```

    <result><xsp:expr>result_xml</xsp:expr></result>
  </to-xml>
  <to-xindice>
    <collection><xsp:expr>coll</xsp:expr></collection>
    <oid><xsp:expr>oid</xsp:expr></oid>
    <result><xsp:expr>result_xindice</xsp:expr></result>
  </to-xindice>

</page>
</xsp:page>

```

(*)col.createResource(oid,"XMLResource")に代えて col.createResource(null,"XMLResource")とした場合、登録 XML ドキュメントのキーの値として、例えば 01ffa7c0776c8edf00000106d1c01e66 のような値が Xindice によって与えられる。

付録 E : XML ドキュメント修正 XSP コード

xindice-xupdate-22.xsp :

```

<?xml version="1.0" encoding="Windows-31J"?>
<!-- <?xml version="1.0" encoding="Shift_JIS"? -->
<!-- Author: M.Aoki "aoki@ed.kagawa-u.ac.jp" -->
<xsp:page language="java"
  xmlns:xsp="http://apache.org/xsp"
  xmlns:xsp-request="http://apache.org/xsp/request/2.0"
  xmlns:sendmail="http://apache.org/cocoon/sendmail/1.0">

<xsp:structure>
  <!--xsp:include>org.apache.xindice.*</xsp:include-->
  <xsp:include>org.xmldb.api.base.*</xsp:include>
  <xsp:include>org.xmldb.api.modules.*</xsp:include>
  <xsp:include>org.xmldb.api.*</xsp:include>
  <!--xsp:include>java.io.*</xsp:include-->
</xsp:structure>

<xsp:logic>
//もしあれば、ここに関数を定義する
</xsp:logic>

<page>
<xsp:logic>
// Request Data from HTML-FORM
// For Example
// String st0 = request.getParameter("oid");
// String st1 = new String(st0.getBytes("ISO8859_1"),"UTF-8");

String st0 = request.getParameter("oid");
String oid = null;
if (st0 == null || st0.length() == 0)
  oid = "kyu4";
else
  oid = st0; // パラメータ oid が漢字を含まないとき
/* ### パラメータ oid が漢字を含むとき,Encoding は oid の返し方による
try{
  oid = new String(st0.getBytes("ISO8859_1"),"Shift_JIS");

```

```

// oid = new String(st0.getBytes("ISO8859_1"),"JISAutoDetect");
// oid = new String(st0.getBytes("ISO8859_1"),"UTF-8");
} catch (Exception e){ oid = "kyu4"; }
### */
// Update Data
String teacher = "幸町 分室";
String comment = "補講にかえてレポートを課す";

String xu_modf_b = "<xu:modifications version=¥"1.0¥" xmlns:xu=¥"http://
www.xmldb.org/xupdate¥">";
String xu_modf_e = "</xu:modifications>";

String xu_remove = " <xu:remove select=¥"/休講/備考¥"/>";
String xu_update = " <xu:update select=¥"/休講/担当教員¥"> + teacher +
"</xu:update>";
String xu_insert_b = " <xu:insert-after select=¥"/休講/授業名¥">";
String xu_element = " <xu:element name=¥"備考¥"> + comment +
"</xu:element>";
String xu_insert_e = " </xu:insert-after>";

StringBuffer bf = new StringBuffer(xu_modf_b);
    bf.append("¥r¥n" + xu_remove)
        .append("¥r¥n" + xu_update)
        .append("¥r¥n" + xu_insert_b)
        .append("¥r¥n" + xu_element)
        .append("¥r¥n" + xu_insert_e)
        .append("¥r¥n" + xu_modf_e + "¥r¥n")    ;
String update_data = bf.toString();

//XUpdate Xindice
String coll = "kyuukou20";
Collection col = null    ;
String result_update = "";
/* // For Property
String property1 = "configuration";
String property2 = "db-home";
String con_file = "/Tomcat5.0/webapps/cocoon/WEB-INF/xindice.xml/";
String db_home = "/Tomcat5.0/webapps/cocoon/WEB-INF/";
*/
long num = 0;
try {
String driver = "org.apache.xindice.client.xmldb.embed.DatabaseImpl";
// String driver = "org.apache.xindice.client.xmldb.DatabaseImpl";
Class c = Class.forName(driver);
Database database = (Database) c.newInstance();
// database.setProperty(property1,con_file) ;
// database.setProperty(property2,db_home) ; // OK
DatabaseManager.registerDatabase(database);

String uri = "xmldb:xindice-embed:///db/" + coll;
col = DatabaseManager.getCollection(uri);

XUpdateQueryService service =
(XUpdateQueryService)col.getService("XUpdateQueryService","1.0") ;

```

```

    if (oid.equals("all"))
        num = service.update(update_data) ;
    else
        num = service.updateResource(oid,update_data) ;
        //特定の oid ドキュメントを修正
result_update = "success" ;
    } catch (Exception e){ result_update = "failure" ; }
    finally {
        try {
            if (col != null)
                col.close() ;
        } catch(Exception e){;}
    }
//
</xsp:logic>
    <xupdate-xindice>
        <collection><xsp:expr>coll</xsp:expr></collection>
        <oid><xsp:expr>oid</xsp:expr></oid>
        <result><xsp:expr>result_update</xsp:expr></result>
        <update-num><xsp:expr>num</xsp:expr></update-num>
    </xupdate-xindice>
</page>
</xsp:page>

```

PPP 接続ログ集計スクリプトの作成

- スクリプト言語 Groovy を用いて -

教育学部 (人間環境教育)

青木 昌三

はじめに

本学総合情報基盤センターでは、学内ユーザに対して自宅等学外から学内 LAN への PPP 接続サービスを提供している。このサービスの利用状況を把握するため、旧香川大学情報処理センターの時代の 1998 年からこれまで、作成したログ集計プログラムを用いて毎月の PPP 接続状況を集計してきた⁽¹⁾。今回、これまで長年にわたって利用してきた Java プログラムに替わるプログラムをスクリプト言語を用いて作成し直したので^(*)、用いたスクリプト言語 Groovy の簡単な紹介をも含めて報告する。

本稿で取り扱っている内容の具体的な項目は次の通りである。

Groovy によるスクリプトの作成 PPP 接続ログ集計スクリプトの作成

以下では、既に JVM(Java Virtual Machine)が利用できる環境が整っているものとする。本稿で紹介するスクリプトの動作確認については、OS が Windows2000 (SP4) の PC に Java2SDK として j2sdk1.4.2_03 がインストールされた環境のもとで行っている。用いた Groovy は groovy-1.0-beta-10 である。

(*)文献(1)にも紹介されている Java 言語によるログ集計プログラムは、PPP 接続状況だけでなく教育用 PC の利用状況をも併せて集計できるよう設計されたもので、今となっては作成した筆者にも分り難い程にコーディング量も多い (A4 判用紙約 62 ページ)。2005 年 8 月、これまでこのプログラムを利用してきた古い PC が故障するという事態となり、さすがに “Write Once, Run Anywhere” と言われるだけあって別途保存していたプログラムは新しい PC 環境下でも動作するが、PC の故障を機に可読性に優れたスクリプトを新たに作成することとした。

Groovy によるスクリプトの作成

Groovy は JVM のための強力かつダイナミックなオープンソースのスクリプト言語である。その歴史は浅く、2003 年に開発が始められた言語で、今も尚改良・開発が進められ日々発展している。

本稿では、“Groovy について” をはじめとして Groovy のインストール、実行方法やスクリプト言語の構文・文法等について詳しくは説明しない。こうした事柄については、文献(2)、(3)や

<http://groovy.codehaus.org/>

での「Using Groovy」や「Language Guide」の項を参照していただきたい。インストール先の docs ディレクトリ (フォルダ) 内にも同様の内容を記述した html ドキュメントが収められている。

以下では、実際の Groovy スクリプト例をいくつか紹介し、本稿の目的である PPP 接続ログ集計スクリプトを理解する準備とする。具体的なコーディング内容を読み取るとともに、スクリプト言語 Groovy の簡潔さ、素晴らしさを直接感じ取っていただきたい。Java によるプログラミングの経験の有無に拘わらず、これほどのことが簡単に記述でき実行できることに驚きを覚えると共に、直ぐにでも Groovy を用いてみたくなるのではなからうか。

Groovy の動作環境としては、Java2SDK 1.4 が必要である^(*)。Java2SDK のインストール先ディレクトリを環境変数 JAVA_HOME に設定するとともに、Groovy のインストール先を環境変数 GROOVY_HOME に設定、更に %GROOVY_HOME%\bin を PATH に加えておく。尚、Unix 上で用いる場合には表記を適当に読み替えてほしい。

(*)古いバージョンの Java2SDK を用いた場合、ファイルを読み込むスクリプトでエラーが生じる。J2sdk1.4.2_03 のもとでは問題なく動作する次のスクリプト

```
// ### test1.groovy ###
import java.io.File

println "Hello World"
new File(args[0]).eachLine { println it }
```

は、"Hello World"を表示した後、引数に与えられたファイルの各行を順次表示するものであるが、j2sdk1.4.0_03 のもとでは次の様なエラーが表示された。

```
C:\MyProgram\Javaetc\Mygroovy>groovy test1 step1.groovy
Hello World
Caught: java.nio.charset.UnsupportedCharsetException: MS932
    at test1.run(C:\MyProgram\Javaetc\Mygroovy\test1.groovy:5)
    at test1.main(C:\MyProgram\Javaetc\Mygroovy\test1.groovy)
C:\MyProgram\Javaetc\Mygroovy>
```

“ Hello World ” スクリプト hello.groovy

まずは、” Hello World ” を表示するスクリプトである。

```
println "Hello World !"
```

と 1 行だけ記述した hello.groovy ファイルを用意し次のように実行する。

```
C:\MyNote\Scripts\MyGroovy>groovy hello.groovy
Hello World !
```

実に単純明快である。尚、groovy hello.groovy にかえて、groovy hello でもよい。

引数の内容を順次表示するスクリプト hello2.groovy

引数の内容を順次表示するスクリプトである。引数を順次イテレートする様子に注意してほしい。次の内容を記述した hello2.groovy ファイルを用意する。

```
args.each { println " ${it}さん、お元気ですか !" }
```

実行結果は次の様となる。

```
C:\MyNote\Scripts\MyGroovy>groovy hello2 "兵庫" "岡山" "香川"
```

兵庫さん、お元気ですか！
岡山さん、お元気ですか！
香川さん、お元気ですか！

ファイル I/O にかかわるスクリプト

ファイルの入出力 (I/O) を取り扱うスクリプトを紹介する。

Grep コマンドスクリプト ggrep.groovy

このスクリプトではファイルからの入力と正規表現 (regex) を取り扱っている。利用法は , groovy ggrep regex-pattern input-file-name である。スクリプトの例を示す。

```
// Grep Sample ggrep.groovy 2005.09.07
// Usage: groovy ggrep[.groovy] regex-pattern file-name

import java.io.File

if(args.length < 2){
    println " Usage : groovy ggrep[.groovy] regex-pattern file-name"
    println ""
    System.exit(0)
}
new File(args[1]).eachLine {
    if (it =~ args[0])
        println it
}
```

実行例として、sampletxt.s ファイルから文字列 “ 購入 ” を含む行を表示してみる。

```
C:¥MyNote¥Scripts¥MyGroovy¥MySamples>groovy ggrep "購入" sampletxt.s
雑誌の購入についてのアナウンス
既にメールでお知らせしたところですが、来年度の雑誌購入に関して、
継続、或いは購読中止、新規購入についてご意見がありましたら、恐れ
尚、物品（備品、消耗品）の購入についてもご希望がありましたら、連絡
```

Cat コマンドスクリプト gcat.groovy

Unix でお馴染みの Cat コマンドの Groovy 版スクリプトの例である。コメントとされた部分が多いが何れの行も同じように動作する。利用法は , groovy gcat file1,file2,... である。ファイル名が省略された場合には標準入力の対象となる。

```
// Cat Sample gcat.groovy 2005.09.08
// Usage: groovy gcat[.groovy] [file-name1[,filename2,...]
// 標準入力に対応
import java.io.File

if(! args.length )
    // System.in.eachLine { println it }
    System.in.newReader().eachLine { println it } // BufferedReader
else
    for ( st in args )
        new File(st).newReader().eachLine { println it } // BufferedReader
    // new File(st).eachLine { println it }

/* OK
```

```

c = { line | println line }          (1)
if(! args.length )
// System.in.eachLine { println it }
System.in.newReader().eachLine(c)    (2)
// System.in.newReader().eachLine { println it } // BufferdReader
else
  for ( st in args )
    // new File(st).newReader().eachLine { println it } // BufferdReader
    new File(st).newReader().eachLine(c) // BufferdReader          (3)
  // new File(st).eachLine { println it }
*/

```

上記スクリプトで注目してほしいのは(1)～(3)の部分である。c = { line | println line } は Closure と呼ばれるコードブロックで、line はこの Closure の引数、“println line” は受け取った引数を表示する。変数 c に Closure が代入され、c は(2)、(3)において eachLine メソッドの引数となっている (eachLine(c))。内容的には同じである前後の記述、eachLine { println it } と比較してほしい。尚、これまでに何度か目にした { println it } の it は Closure のデフォルト引数である。

テキストビューアースクリプト `gtxtviewer.groovy`

テキストファイルの内容を表示するスクリプトであるが、対象とするテキストの文字コードを指定できる。例えば、JIS コードのテキストファイル `jtext` の内容を表示するのであれば、

```
groovy gtxtviewer -jis jtext
```

とする。示したスクリプトの内容から分るように、文字コードが省略された場合には `JISAutoDetect` が設定されるので、JIS、日本語 EUC、Windows-31J (Shift_JIS) の各コードが自動判定の対象となる。UTF-8、Unicode のテキストも表示が可能である。文字コードを具体的に指定することによりテキストファイルの文字コードを識別したり、リダイレクト機能を利用して、各文字コードのテキストをシステムにデフォルトの文字コードファイルに変換するなどにも利用できる。

```

// gtxtviewer.groovy 2005.09.07
// Usage: groovy gtxtviewer[.groovy] [-charset] file-name

import java.io.File

if(args.length == 0){
  println ""
  println " Usage : groovy gtxtviewer[.groovy] [-charset] file-name"
  println "      charset : sjis for Windows-31J(Shift_JIS), euc for EUC_JP"
  println "      charset : jis for ISO-2022-JP, utf for UTF-8, uni for Unicode"
  println ""
  System.exit(0)
}
encode = "JISAutoDetect"
filename = args[0]
if (args.length == 2) {

```

```

filename = args[1]
code = args[0] - '-'
code = code.toUpperCase()
switch (code) {
    case "SJIS" : encode = "Windows-31J" ; break
    case "EUC" : encode = "EUC_JP" ; break
    case "JIS" : encode = "ISO-2022-JP" ; break
    case "UTF" : encode = "UTF-8" ; break
    case "UNI" : encode = "Unicode" ; break
    default : break
}
}
reader = new File(filename).newReader(encode) //BufferedReader
// println reader.getText() //OK
reader.eachLine {
    println it
}

```

文字コード変換スクリプト gconv.groovy

前出の gtxtviewer.groovy を下敷きとして、テキストファイルの文字コードを変換して指定された文字コードのファイルに出力するよう書き加えたスクリプトである。利用法は

groovy gconv[.groovy] -charset1 filename-from -charset2 filename-to
で、例えば JIS コードのテキストファイル jtext.txt から UTF-8 コードのファイル utext.txt を得るには

groovy gconv -jis jtext.txt -utf utext.txt
とする。スクリプトの例を付録 A に示している。

クラスとリストを用いるスクリプト

クラスとリストの使用例として1つのレポート課題を採りあげる^(*)。

課題：

次のようなデータがある。1行目はフィールド名を記していて、それぞれ、学籍番号、氏名、ユーザ ID、読みを示している。

```

sno      name      uid      yomi
11L021,兵庫 輝夫,s11l021,hyougo teruo
11L040,大阪 智紀,s11l040,osaka tomoki
11L054,堺 佑輔,s11l054,sakai yusuke
08L018,岡山 友子,s08l018,okayama yuuko
08L043,和歌山 伸章,s08l043,wakayama shinsyo
07L074,徳島 英和,s07l074,tokushima eiwa
11X016,愛媛 豊 ,s11x016,ehime yutaka

```

(1)プログラム 1：データをフィールド名をつけた配列に代入せよ。

- (2)プログラム 2 : 代入したデータをすべて印刷せよ。
- (3)プログラム 3 : データを読み(yomi)について辞書式の小さい順序に並べて, yomi , name , sno を印刷せよ。
- (4)プログラム 4 : 利用者が入力した sno を読み込み, 一致する (大文字, 小文字の区別をしない) sno があれば, uid , name , yomi を印刷する。一致する sno がなければ「存在しない」を出力する。sno として空行が入力された時点でプログラムは終了する。

解答例 :

配列を用いるのではなく, 4つのプロパティ sno , name , uid , yomi をもつクラス Member を定義し, データファイル (members.dat) から読み込んだ各データをもとに Member のインスタンスを作成, それらをリスト Members の各要素とする。問い(1) ~ (3)はリスト Members の各要素を適宜プリントアウトすることで解決する。問い(4)は, 標準入力からの入力値と一致する sno 値をもつ Member 要素を Members リストから選び出し, それをプリントアウトすればよい。解答例のスクリプト report.groovy を付録 B に示す。

(*)このレポート課題は, 本学工学部の古川 善吾教授が学生に Java による初等プログラミング課題として課したもので, 許可を得て利用させていただいた。勿論, 学籍番号や氏名は適宜変更している。

データベースにアクセスするスクリプト

Groovy を用いるとデータベースへのアクセスも簡単に実行できる。RDB としての MySQL と XML データベースである Xindice^(*)にアクセスするスクリプト例を紹介する。

(*)Xindice は Apache Software Foundation (<http://www.apache.org/>)によるオープンソースのネイティブな XML データベース・ソフトウェアである。<http://xml.apache.org/xindice> を参照。

MySQL にアクセスするスクリプト gsql.groovy

MySQL への接続は実に簡単に実行可能である^(*)。既にインストールされている MySQL にはデータベース mail に 2つのテーブル, lists と users が収められている。lists テーブルを対象としてテーブルの内容を表示したり新たなデータを挿入・削除するスクリプトを作成した。MySQL への接続部分のみを記すと次の様である。

```
import groovy.sql.Sql
dbname = "mail"
dbURL='jdbc:mysql://localhost:3306/' + dbname
dbdriver = 'org.gjt.mm.mysql.Driver'
// dbdriver = 'com.mysql.jdbc.Driver' //OK
sql = Sql.newInstance(dbURL,dbdriver) //MySQL に接続
//===== 以後 sql を用いて問い合わせ等を実行する =====
```

スクリプト内容とそれによる実行結果を付録 C に示す。

(*)「Groovy SQL」(<http://groovy.codehaus.org/Groovy+SQL>)並びに文献(3)を参照。gsql.groovy 作成にあたっては「Groovy SQL」を参考とした。

Xindice にアクセスするスクリプト gxmlldb1.groovy

筆者の PC 環境では、ネイティブ XML データベース Xindice は既にサーブレット解釈実行エンジンである Tomcat 上で動作しており^(*1)、Xindice には addressbook と kyuukou の 2 つのコレクションが登録されている。さらに各コレクションには次のドキュメントが登録されている^(*2)。

コレクション addressbook :

address1.xml (キー : address1) , address2.xml (キー : address2)

コレクション kyuukou :

kyu1.xml (キー : kyu1) , kyu2.xml (キー : kyu1) , kyu3.xml (キー : kyu3)

作成したスクリプト gxmlldb1.groovy では、Xindice とのコネクションを確立するとともに Xpath の指定により該当するデータをプリントアウトするようにしている。Xpath や XQuery については、Xindice ではなく同様の XML データベース eXist を取り扱ったものではあるが、文献(4)に詳しく説明されているので是非参照していただきたい。尚、スクリプト gxmlldb1.groovy は「Xindice 1.1 Developer Guide」^(*3)内のサンプル Java コードを参考として、変数のタイプや try/catch 節を省略するなど Groovy スタイルに書いたものである。付録 D にスクリプト内容とその実行結果等について示す。尚、スクリプトの実行にあたっては、Groovy が Xindice にかかわるライブラリを参照できるよう設定しておくことが必要である (付録 D 参考(1)を参照)。

(*1)xml-xindice-1.1b4-war.zip ファイル等を解凍して得られる xindice-1.1b4.war を xindice.war と名前変更して Tomcat のインストール先 (%CATALINA_HOME%) の webapps フォルダ内にコピーすれば、Xindice が Tomcat を実行エンジンとして動作する。尚、xindice-1.1b4.xml ファイルの記述に従って同ファイルに記述されている<Context>・・・</Context>の部分を、Tomcat のシステムファイル (%CATALINA_HOME%\conf\server.xml ファイル) にペーストしておけばより丁寧である。

(*2)コレクションやドキュメントの登録については、「Xindice 1.1 Administration Guide」, 「Xindice 1.1 Commandline Tool Guide」(<http://xml.apache.org/xindice> の Documentation) を参照。

(*3)<http://xml.apache.org/xindice> の Documentation を参照。

PPP 接続ログ集計スクリプトの作成

本稿の目的であるログ集計スクリプトを作成する。ログの集計は、認証サーバ RADIUS (Remote Authentication Dial-In User Service) によって得られるログファイルをもとに、ひと月単位で各利用者並びに全体としての PPP 接続回数、接続時間(秒)を求めている。

ログファイルの書式

RADIUS によって作成されるログファイルでは本来、ひと月単位ではなく日々連続して PPP 接続にかかわる接続 (Start)・切断 (Stop)の状況が記録されている。が、こ

れでは集計に不便なので、サーバマシン上 CronJob により概ね 1 ヶ月単位のログファイルを得るようにしている。例えば、2005 年 8 月のログは、dt200508 ファイルにまとめられている。

ログファイルの書式は次のようである（集計に必要な部分を抜粋している）。

Fri Aug 5 11:12:59 2005 Acct-Status-Type = Start ===== 中 略 ===== User-Name = "somename1" ===== 中 略 ===== Acct-Session-Id = "5900172c" ===== 中 略 =====	行の先頭から Fri Aug 行の先頭はタブ
Fri Aug 5 11:13:01 2005 Acct-Status-Type = Stop ===== 中 略 ===== Acct-Session-Time = 1 ===== 中 略 ===== User-Name = "somename1" ===== 中 略 ===== Acct-Session-Id = "5900172c" ===== 中 略 =====	
Fri Aug 5 11:13:14 2005 Acct-Status-Type = Start ===== 中 略 ===== User-Name = "somename1" ===== 中 略 ===== Acct-Session-Id = "5900172d" ===== 中 略 =====	
Fri Aug 5 11:26:06 2005 Acct-Status-Type = Stop ===== 中 略 ===== Acct-Session-Time = 1754 ===== 中 略 ===== User-Name = "regg0260" ===== 中 略 ===== Acct-Session-Id = "5900172b" ===== 中 略 =====	
Fri Aug 5 11:38:12 2005 Acct-Status-Type = Start	

後ほどの集計に不必要と思われる行は省略しているが、上記の様子から、ログの書式について次のことが言える。

接続・切断の記録開始は、行の先頭から Fri Aug 5 11:12:59 2005 のように、曜日 月 日 時刻 年 で始まっている。また、記録と記録は空行で区切られている。

他の行は先頭がタブ記号（"¥¥t"）で始まっている。

接続開始あるいは接続終了は

Acct-Status-Type = Start あるいは Acct-Status-Type = Stop
の記述によって区別される。

接続終了の記録には

Acct-Session-Time = 1754

のように、接続時間が秒単位で記されている。

同じ利用者にかかわる接続開始・終了であるかどうかは、User-Name と Acct-Session-Id
の項目を比較することにより判別し得る。

記録は時系列にそっているので、接続時間の長短により、必ずしも開始、終了、開始、終
了、・・・のように順序よく並んでいる訳ではない。

スクリプト作成の手順

最終的なスクリプト作成に当たっては、次の 4 つの手順に従いそれぞれのステップご
とに順次スクリプトを作成しつつ進めることとした。

ステップ 1 : step1.groovy

ここでは先ず、RADIUS ログデータを読み込んで、接続開始の部分と接続終了の部
分を切り分ける。接続開始部分については

先頭行、User-Name 項目を含む行、Acct-Session-Id 項目を含む行
を、接続終了部分については

先頭行、Acct-Session-Time 項目、User-Name 項目及び Acct-Session-Id 項目を含む
行

をそれぞれ切り出して、入力データファイル名が例えば dt200508 であれば、
dt200508.start ファイルと dt200508.stop ファイルにその内容を書き出す。実行は

```
groovy step1 data-file-name
```

による。

ステップ 2 : step2.groovy

クラス LogCntData を定義する。ステップ 1 で切り分けた各データから LogCntData
のインスタンスを作成し、それらを接続開始 / 終了に基づいて接続開始用リストあるい
は接続終了用リストの要素とする。クラス LogCntData は次のように定義している。

```
class LogCntData{  
    date ; time ; user_name ; session_id ; session_time  
    String toString(){  
        "${date},${time},${user_name},${session_id},${session_time}"  
    }  
}
```

ステップ 2 の最終段階では、リストに収められた各要素を接続開始、接続終了に分けて
ファイルに出力する。接続終了に関わる出力の例を示す。

```
2005/08/01,02:08:23,rggg0080,590016c2,23
2005/08/01,07:07:06,somename2,590016c3,588
2005/08/01,07:16:29,somename2,590016c4,67
2005/08/01,08:20:08,rec0290,590016c5,30
```

ステップ3 : step3.groovy

接続開始と終了のデータを利用者名 (User-Name) とセッション ID (Acct-Session-Id) をもとにクラス LogCnt のインスタンスとしてまとめるとともに、まとめられたインスタンスを要素としてリストに加える。LogCnt の定義は次の通りである。

```
class LogCnt{
    date_start ; time_start ; date_stop ; time_stop ; user_name ; session_id
    session_cnt ; session_time
    String toString(){
        "${date_start},${time_start},${date_stop},${time_stop},${user_name},${session_id}
,${session_cnt},${session_time}"
    }
}
```

接続開始と終了のデータから LogCnt のインスタンスを作成する際に2つの方法が考えられる。接続開始にかかわる LogCntData インスタンスを要素とするリストを log_cnt_start , 接続終了にかかわるそれを log_cnt_stop とする。

(1)log_cnt_start , log_cnt_stop リストが完成して後に , log_cnt_start の各要素に対してペアとなる log_cnt_stop の要素を検索し , 見つかったならば2つの要素から LogCnt のインスタンスを作成して , 例えばリスト log_cnts に要素として加えていく。ペアとなった log_cnt_start , stop の要素はそれぞれリストから削除する。

(2)接続終了にかかわる LogCntData のインスタンスを作成するたびにペアとなる要素を log_cnt_start リストから検索し , 見つかったならば2つの要素から LogCnt のインスタンスを作成して log_cnts に要素として加えていく。ペアとなった log_cnt_start 要素はリストから削除する。また , ペアが見つからなかった場合には , 作成した LogCntData インスタンスは log_cnt_stop の要素とする。

上記(1)の方法で step3.groovy , (2)の方法で step3v2.groovy を作成した。接続件数が非常に多い場合のメモリの消費やコーディングの際の簡潔さ等を考えると(2)の場合を採択する方が良いように思われる。尚 , (1) , (2)何れの場合でも , log_cnt_start , log_cnt_stpo リストに最終的に残っている要素はペアが見つからなかった要素である。念のため , これらはエラー要素としてファイルに出力するのが適切だろう。log_cnts リストの各要素を出力するよう設定しているが , その一部を示す。

```
2005/08/01,02:08:00,2005/08/01,02:08:23,rggg0080,590016c2,1,23
2005/08/01,06:57:18,2005/08/01,07:07:06,somename2,590016c3,1,588
```

```
2005/08/01,07:15:23,2005/08/01,07:16:29,somename2,590016c4,1,67
2005/08/01,08:19:38,2005/08/01,08:20:08,reec0290,590016c5,1,30
2005/08/01,08:23:13,2005/08/01,09:13:18,rtuv0130,590016c6,1,3005
```

ステップ 4 : step4.groovy , step4v2.groovy

ステップ 3 で得られた log_cnts リスト (各要素はクラス LogCnt のインスタンス) をもとにデータを集計する。先ずそのための LogCntSum クラスを次のように導入する。

```
class LogCntSum{
    year ; month ; user_name ; acc_num ; total_time
    String toString(){
        "    ${year}年 ${month}月,${user_name},${acc_num},${total_time}"
    }
}
```

log_cnts の先頭の要素を選び , その要素の接続開始の年 , 月と利用者名が一致する要素の部分リストを求め , それらの要素について集計して作成した LogCntSum クラスのインスタンスをリスト log_sums に加えていく。集計の終わった部分リストは log_cnts リストから削除する。

最後に , log_sums リストの要素を利用者名順にソートするなどして集計結果をファイルに出力する。

ログ集計スクリプトの完成 pppacc.groovy , pppaccv2.groovy

上述の手順に従い step4.groovy と step4v2.groovy が作成された。各ステップに於いて動作を確認しつつ作成されており , 最終的に得られたスクリプトの動作にも問題は無い。step4.groovy を pppacc.groovy に , step4v2.groovy を pppaccv2.groovy に名前を変更しておこう。用い方は次のとおりである。

```
groovy pppacc.groovy log-file-name
groovy pppaccv2.groovy log-file-name
```

pppaccv2.groovy スクリプトの内容を付録 E に示す。最終結果を出力するまでいくつかのファイルを作成している。どのようなファイルが作成されるかスクリプト内容を眺めてほしい。あわせて , dt200508 ログファイルから得られた結果も参考までに付録に示しておく。

おわりに

スクリプト言語 Groovy を用いての PPP 接続ログ集計スクリプト作成について紹介した。新しい言語を用いて何か具体的なツール作成を考えていた矢先 , 古い集計プログラムが動作していた PC が故障するというアクシデントに見舞われ , このことが Groovy による集計プログラム作成への動機となった。出来上がったスクリプトは約 5 ページ , 思ったよりもコーディング量も少なく , 何よりも極めて短時間に作成できてしまった。

使い勝手も悪くはないし，変更等も容易である。総合情報基盤センターのログ集計作業に役立てていただけることを期待したい。

Groovy の文法等に詳しく触れることなく幾つかのサンプルスクリプトを紹介した。内容がいささか初等的に過ぎ，その割には随分と紙数を割いてしまったが，Java 言語との親和性も高く，迅速でダイナミックなスクリプト言語であると呼ばれる Groovy に少しでも魅力を感じていただければ幸いである。

参考文献

- (1)青木 昌三，瀬野 芳孝，曾根 計俊，丸山 久美子：「ログ集計プログラムの作成」
香川大学情報処理センター 年報 第7号（1999年2月）
- (2)関谷 和愛，上原 潤二：「Groovy の魅力を探る」
JavaWorld 2004 November，アイ・ディ・ジー・ジャパン
- (3)マークボルクマン著 / 角谷 新太郎訳：「Groovy - Java 用スクリプト言語」
<http://www.kakutani.com/trans/ociweb/jnbFeb2004.html>
art.lang.jre コラム：「Groovy に触ってみよう」
<http://www.atmarkit.co.jp/fjava/rensai3/devworks05/devworks05.html>
- (4)吉田 晃伸，大田 純：「はじめての XQuery プログラミング 第1回～」
JavaWorld 2004 September，アイ・ディ・ジー・ジャパン

〔付録〕

付録 A：文字コード変換スクリプト

gconv.groovy：

```
// gconv.groovy 2005.09.07
// Usage: groovy gconv[.groovy] -charset1 filename-from -charset2 filename-to
// Convert text encode1 to encode2

import java.io.File

def print_usage(){
    println ""
    println "Usage : groovy gconv[.groovy] -charset1 filename-from -charset2
filename-to"
    println " charset1(2) : sjis for Windows-31J(Shift_JIS), euc for EUC_JP"
    println " charset1(2) : jis for ISO-2022-JP, utf for UTF-8, uni for Unicode"
    println ""
}
// ##### Script
if(args.length < 4){
    print_usage()
    System.exit(0)
}
```

```

    }
    encode = "JISAutoDetect"
    encode_to = "Windows-31J"
    filename_from = args[1]
    filename_to = args[3]
    code1 = args[0] - '-'
    code2 = args[2] - '-'
    code1 = code1.toUpperCase()
    code2 = code2.toUpperCase()
    switch (code1) {
        case "SJIS" : encode = "Windows-31J" ; break
        case "EUC" : encode = "EUC_JP" ; break
        case "JIS" : encode = "ISO-2022-JP" ; break
        case "UTF" : encode = "UTF-8" ; break
        case "UNI" : encode = "Unicode" ; break
        default : break
    }
    switch (code2) {
        case "SJIS" : encode_to = "Windows-31J" ; break
        case "EUC" : encode_to = "EUC_JP" ; break
        case "JIS" : encode_to = "ISO-2022-JP" ; break
        case "UTF" : encode_to = "UTF-8" ; break
        case "UNI" : encode_to = "Unicode" ; break
        default : break
    }

    reader = new File(filename_from).newReader(encode) //BufferedReader
    writer = new File(filename_to).newPrintWriter(encode_to)
    reader.eachLine {
        writer.println it
    }
    // writer.flush()
    writer.close()

```

付録 B : レポート課題の解答例

report.groovy : (1 行内に記すべきコードを便宜上折り返して表している)

```

import java.io.File

class Member{
    sno; name; uid; yomi;
    String toString(){
        " ${sno} ${name} ${uid} ${yomi} " }
    String toString3(){
        " ${yomi} : ${name} : ${sno} " }
    String toString4(){
        " ${uid} : ${name} : ${yomi} " }
}

// Program 1 : Read members.dat file
println ""
println "## Program 1 : Reads members.dat file and shows it's each line ##"
println ""
new File("members.dat").eachLine { println it}

```

```

println ""

// Program 2
println ""
println "## Program 2 ##"
println ""

Members=[]
new File("members.dat").withReader {reader |
    while(line=reader.readLine()){
        LineList = line.tokenize(',')
Members.add new Member(sno:LineList[0],name:LineList[1],uid:LineList[2],
        yomi:LineList[3])
    }
}

println " sno          name          uid          yomi"
Members.each { println it }

// Program 3
println ""
println "## Program 3 ##"
println ""
MembersSorted = []
Members.each { MembersSorted.add it } // Copy
MembersSorted.sort { it.yomi } // Sort
MembersSorted.each {println it.toString30 }

// Program 4
println ""
println "## Program 4 ##"
println ""

while (true){
    print "sno を入力してください: "
    st= System.in.readLine()
    if (st.length() == 0)
        break
    found = Members.find { it.sno.equalsIgnoreCase(st) }
    if (found)
        println found.toString40
    else
        println " 該当する sno は存在しません"
    println ""
}

println ""
// END

```

付録 C : MySQL にアクセスするスクリプト

gsql.groovy :

```

// SQL Sample gsql.groovy 2005.09.12
// Usage: groovy gsql.groovy

```

```

import groovy.sql.Sql

dbname = "mail"
dbURL='jdbc:mysql://localhost:3306/' + dbname
dbdriver = 'org.gjt.mm.mysql.Driver'
// dbdriver = 'com.mysql.jdbc.Driver' //OK

sql = Sql.newInstance(dbURL,dbdriver) //MySQL に接続

//テーブルの表示
println "テーブルの表示(show tables) :"
sql.eachRow('show tables'){ println " " + it.Tables_in_mail}
println ""

//lists テーブルのフィールド項目を表示
println "lists テーブルのフィールド項目を表示(describe lists) :"
sql.eachRow('describe lists') {
    println " Field = ${it.Field} ; Type = ${it.Type} ; Key =  ${it.Key}"
}
println ""

//lists テーブルの各データ内容を表示
count = 1
println 'lists テーブルの各データ内容を表示(select * from lists) :'
sql.eachRow('select * from lists') {
    println " (${count})listName = ${it.listName} ; ListSubscriber = ${it.listSubscriber}"
    ++count
}
println ""

//lists テーブルにデータを挿入
dst_in = "Insert into lists values (¥'list-kyu5¥','¥test-user@cc.kagawa-u.ac.jp¥)"
sql.executeUpdate(dst_in)
println "データの挿入： 挿入されたデータ"
sql.eachRow("select * from lists where (listName = ¥'list-kyu5¥') and (ListSubscriber
REGEXP ¥'^test.*@cc¥') ") {
    println " listName = ${it.listName} ; ListSubscriber = ${it.listSubscriber}" }
print " lists テーブルのデータ数： "
sql.eachRow('select count(*) from lists') {
    println it[0]
}
println ""

//lists テーブルからデータを削除
println "データの削除： 先に挿入したデータを削除"
dst_del = "delete from lists where (listName = ¥'list-kyu5¥') and (ListSubscriber
REGEXP ¥'^test.*@cc¥') "
sql.executeUpdate(dst_del)
sql.eachRow("select * from lists where listName = ¥'list-kyu5¥'") {
    println it.listName + " " + it.ListSubscriber
}
print " 削除後の lists テーブルのデータ数： "
sql.eachRow('select count(*) from lists') {
    println it[0]
}

```

```
}  
println ""  
sql.close() // コネクションを閉じる
```

gsql.groovy の実行結果 :

テーブルの表示(show tables) :

```
lists  
users
```

lists テーブルのフィールド項目を表示(describe lists) :

```
Field = listName ; Type = varchar(64) ; Key = PRI  
Field = listSubscriber ; Type = varchar(255) ; Key = PRI
```

lists テーブルの各データ内容を表示(select * from lists) :

```
(1)listName = list-kyu ; ListSubscriber =  
r-user01@end-pro-1000.aoki-home.kagawa-u.ac.jp  
(2)listName = list-kyu ; ListSubscriber = r-user01@raavie01.aoki-home.kagawa-u.ac.jp  
(3)listName = list-kyu ; ListSubscriber = r-user01@sl-epson.aoki-home.kagawa-u.ac.jp  
(4)listName = list-kyu ; ListSubscriber =  
s-user01@end-pro-1000.aoki-home.kagawa-u.ac.jp  
(5)listName = list-kyu ; ListSubscriber = s-user01@[192.168.0.4]  
(6)listName = list-kyu ; ListSubscriber = s-user01@[192.168.0.8]  
(7)listName = list-kyu1 ; ListSubscriber = aoki@ed.kagawa-u.ac.jp  
(8)listName = list-kyu1 ; ListSubscriber =  
s-user01@end-pro-1000.aoki-home.kagawa-u.ac.jp  
(9)listName = list-kyu2 ; ListSubscriber =  
s-user01@end-pro-1000.aoki-home.kagawa-u.ac.jp  
(10)listName = list-kyu3 ; ListSubscriber =  
s-user01@end-pro-1000.aoki-home.kagawa-u.ac.jp
```

データの挿入 : 挿入されたデータ

```
listName = list-kyu5 ; ListSubscriber = test-user@cc.kagawa-u.ac.jp  
lists テーブルのデータ数 : 11
```

データの削除 : 先に挿入したデータを削除

```
削除後の lists テーブルのデータ数 : 10
```

付録 D : Xindice にアクセスするスクリプト

gxmlldb1.groovy :

```
// XMLDB Sample gxmlldb1.groovy 2005.09.15  
// Usage: groovy gxmlldb1.groovy  
  
//import org.apache.xindice.*  
import org.xmlldb.api.base.*  
import org.xmlldb.api.modules.*  
import org.xmlldb.api.*  
  
Collection col = null  
driver = "org.apache.xindice.client.xmlldb.DatabaseImpl"  
c = Class.forName(driver)  
database = c.newInstance()  
DatabaseManager.registerDatabase(database)
```

```

uri = "xmldb:xindice://localhost:8080/db/addressbook"
//uri = "xmldb:xindice://localhost:8080/db/kyuukou"
col = DatabaseManager.getCollection(uri)

name = "¥Robert¥"
xpath = "//person[fname=${name}]"
//xpath = "//休講[曜日='金曜']"
// service = (XPathQueryService) col.getService("XPathQueryService","1.0") //OK
service = col.getService("XPathQueryService","1.0")

resultset = service.query(xpath)
results = resultset.getIterator()
while (results.hasMoreResources()){
    res = results.nextResource()
    println res.getContent()
}
if (col != null)
    col.close()

```

gxmlldb1 の実行結果 :

```

C:¥MyNote¥Scripts¥MyGroovy¥MySamples>groovy gxmlldb1
trying to register database
<person src:col="/db/addressbook" src:key="address2" xmlns:src="http://xml.apach
e.org/xindice/Query">
  <fname>Robert</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-1234</phone>
  <phone type="home">123-456-4321</phone>
  <email type="work">robert@mailserv.com</email>
  <email type="home">smith@efg-mail.com</email>
  <address type="work">88,Some City</address>
  <address type="home">22,Abcd City</address>
</person>
C:¥MyNote¥Scripts¥MyGroovy¥MySamples>

```

次に、スクリプト中の uri と xpath を次のように変更してコレクション kyuukou に対して検索を実行する。

```

uri = "xmldb:xindice://localhost:8080/db/kyuukou"
xpath = "//休講[曜日='金曜']"

```

結果は、

```

C:¥MyNote¥Scripts¥MyGroovy¥MySamples>groovy gxmlldb1
trying to register database
<休講 src:col="/db/kyuukou" src:key="kyu3" xmlns:src="http://xml.apache.org/xind
ice/Query">
  <ID>3</ID>
  <日付コード>2006012330</日付コード>
  <日時>1月23日</日時>
  <曜日>金曜</曜日>
  <校時>3校時</校時>
  <担当教員>情報 次郎</担当教員>
  <授業名>情報学基礎</授業名>
  <備考>補講を実施</備考>

```

</休講>

C:¥MyNote¥Scripts¥MyGroovy¥MySamples>

(参考1) Xindice に関係するライブラリを Groovy から参照できるように設定する。

(a) %XINDICE_HOME%¥lib 内にある jar ファイルを %GROOVY_HOME%¥lib フォルダ内にコピーする。但し, commons-logging-1.0.3.jar ファイルは既に存在するのでコピーしない。

(b) xindice.jar ファイルの作成

%XINDICE_HOME%¥classes フォルダ内で次のコマンドを実行し xindice.jar ファイルを作成する。

```
%JAVA_HOME%¥bin¥jar cfv xindice.jar org
```

これにより作成された xindice.jar ファイルを %GROOVY_HOME%¥lib フォルダ内にコピーする。

(参考2) Xindice に登録されているドキュメント例

address1.xml :

```
<?xml version="1.0" ?>
<person>
  <fname>John</fname>
  <lname>Smith</lname>
  <phone type="work">123-456-7890</phone>
  <phone type="home">123-456-0987</phone>
  <email type="work">john@somemail.com</email>
  <email type="home">smith@abc-mail.com</email>
  <address type="work">25,Work City</address>
  <address type="home">30,Home City</address>
</person>
```

kyu1.xml :

```
<?xml version="1.0" encoding="Windows-31J" ?>
<休講>
  <ID>1</ID>
  <日付コード>2006011910</日付コード>
  <日時>1月19日</日時>
  <曜日>月曜</曜日>
  <校時>1校時</校時>
  <担当教員>青木 昌三</担当教員>
  <授業名>物理学概論 1</授業名>
  <備考>後日補講を実施する</備考>
</休講>
```

付録 E : PPP 接続ログ集計スクリプト

pppaccv2.groovy :

```
// ##### Groovy スクリプト pppaccv2.groovy ##### 2005.09.09
// Davias ログデータを, セッション開始と終了の部分に切り分ける
// 各セッションのデータをクラス AsdLogCntData のインスタンスとし, List に収める
// セッション開始と終了のデータを利用者名とセッション ID をもとにまとめる
// まとめられたデータを集計する。集計のためのクラスが AsdLogCntSum である

import java.io.File
// For step3.groovy
class AsdLogCnt{
```

```

date_start ; time_start ; date_stop ; time_stop ; user_name ; session_id
session_cnt ; session_time
String toString(){
    "${date_start},${time_start},${date_stop},${time_stop},${user_name},${session_id},
${session_cnt},${session_time}"
}
}
// For step3.groovy
class AsdLogCntData{
    date ; time ; user_name ; session_id ; session_time
    String toString(){
        "${date},${time},${user_name},${session_id},${session_time}"
    }
}
class AsdLogCntSum{
    year ; month ; user_name ; acc_num ; total_time
    String toString(){
        "${year}年 ${month}月,${user_name},${acc_num},${total_time}"
    }
}
// 関数定義
def getSessionDate(line){
    st = line.trim().tokenize()
    month = st[1]
    day = st[2]
    if (day.length() == 1)
        day = "0" + day
    year = st[4]
    switch (month) {
        case "Jan": month = "01" ; break
        case "Feb": month = "02" ; break
        case "Mar": month = "03" ; break
        case "Apr": month = "04" ; break
        case "May": month = "05" ; break
        case "Jun": month = "06" ; break
        case "Jul": month = "07" ; break
        case "Aug": month = "08" ; break
        case "Sep": month = "09" ; break
        case "Oct": month = "10" ; break
        case "Nov": month = "11" ; break
        case "Dec": month = "12" ; break
        default : break
    }
    st = "${year}/${month}/${day}"
}
def getStartStopTime(line){
    st = line.trim().tokenize()
    time = st[3]
}
def getUser_name(line){
    st = line.trim() - "User-Name = "
    st = st - "'" - "'"
}
def getSessionId(line){

```

```

    st = line.trim() - "Acct-Session-Id = "
    st = st - "" - ""
}
def getSessionTime(line){
    st = line.trim() - "Acct-Session-Time = "
    st.toLong()
}
// for version 2
def getMatchCntIndex(stop_cnt,start_list){
    if (start_list.size() == 0) return -1
    st_user = stop_cnt.user_name
    st_id = stop_cnt.session_id
    match_num = (0...start_list.size()).find {
        start_list[it].session_id == st_id && start_list[it].user_name == st_user
    }
    if (match_num != null && match_num >= 0)
        return match_num
    else
        return -1
}
// for step4
def getSessionYear(line){
    st = line.tokenize('/')
    year = st[0]
}
def getSessionMonth(line){
    st = line.tokenize('/')
    month = "" + Integer.parseInt(st[1])
}
// 関数定義終わり

if(! args.length){
    println "  使用法 :  groovy pppaccv2[.groovy] file-name "
    println ""
    System.exit(0)
}
f_data = new File(args[0])
f_data_parent0 = f_data.getParent()
f_data_parent = (f_data_parent0 == null)? "" : f_data_parent0 + "¥¥"
f_data_name = f_data.getName()
    if (f_data_name.contains('.'))
        f_data_name = f_data_name.tokenize('.')[0]
//ファイル
fname_start = f_data_parent + f_data_name + ".stt$" // for output
fname_stop = f_data_parent + f_data_name + ".stp$" // for output

fname_data = f_data_parent + f_data_name + ".dat$" // for output
fname_err = f_data_parent + f_data_name + ".err$" // for output

fname_sum = f_data_parent + f_data_name + ".sum$" // for final output

start_or_stop = "non"

asd_log_cnt_start = [] // List for "Start"

```

```

asd_log_cnt_stop = [] // List for "Stop"
asd_log_cnts = [] // List for AsdLogCnt Data, for version2

prwriter_start = (new File(fname_start)).newPrintWriter()
prwriter_stop = (new File(fname_stop)).newPrintWriter()

//prwriter_data = (new File(fname_data)).newPrintWriter()
//prwriter_err = (new File(fname_err)).newPrintWriter()

f_data.withReader {reader |
    while(line = reader.readLine()){
        if (line =~ "^¥¥t.*")
            ;
        if (line =~ "[A-Za-z]{3} ") {
            cnt=new AsdLogCntData(date:getSessionDate(line),time:getStartStopTime(line),
            user_name:"UNKNOWN",session_id:"",session_time:0)
            line2 = reader.readLine()
            if (line2 && line2.contains("Start"))
                start_or_stop = "start"
            else if (line2 && line2.contains("Stop"))
                start_or_stop = "stop"
            else
                start_or_stop = "non"

while ((line = reader.readLine()) && line.length() != 0 && line =~ "^¥¥t.*") {
    switch (start_or_stop){
        case "start": {
            if (line.contains("User-Name"))
                cnt.user_name = getUserName(line)
            else if (line.contains("Acct-Session-Id"))
                cnt.session_id = getSessionId(line)
            else
                ;
            break
        }
        case "stop": {
            if (line.contains("User-Name"))
                cnt.user_name = getUserName(line)
            else if (line.contains("Acct-Session-Id"))
                cnt.session_id = getSessionId(line)
            else if (line.contains("Acct-Session-Time"))
                cnt.session_time = getSessionTime(line)
            else
                ;
            break
        }
        default : break
    } // switch
} //while
    switch (start_or_stop) {
        case "start": { asd_log_cnt_start.add cnt
            prwriter_start.println cnt
            break
        }
    }
}

```

```

        case "stop" : { prwriter_stop.println cnt
            match_index = getMatchCntIndex(cnt,asd_log_cnt_start)
            if(match_index == -1)
                asd_log_cnt_stop.add cnt
            else{
                mct = asd_log_cnt_start[match_index]
                asd_ct = new AsdLogCnt(date_start:mct.date,time_start:mct.time,date_stop:cnt.date,
time_stop:cnt.time,user_name:mct.user_name,session_id:mct.session_id,session_cnt:1,
session_time:cnt.session_time)
                asd_log_cnts.add asd_ct
                asd_log_cnt_start.remove(match_index)

                }
                break
            }
        default : break
    } //switch
} //if
} // while
} // Closure end
//ファイルのクローズ
prwriter_start.close()
prwriter_stop.close()

// Step3 Programming
// write data to data_file
prwriter_data = (new File(fname_data)).newPrintWriter()
asd_log_cnts.each { prwriter_data.println it }
prwriter_data.close()

// write data to error_file
prwriter_err = (new File(fname_err)).newPrintWriter()
err_list = asd_log_cnt_start + asd_log_cnt_stop
if(! err_list.size()) {
    err_list.sort { it.date }
    err_list.each { prwriter_err.println it }
    err_list.clear() //remove all elements from List
}
prwriter_err.close()

// ##### Step4 Program #####
asd_log_sums = []
asd_log_cnts.sort { it.user_name } //sort

while(asd_log_cnts.size() > 0){
    fst_cnt = asd_log_cnts[0]
    y_fst = getSessionYear(fst_cnt.date_start)
    m_fst = getSessionMonth(fst_cnt.date_start)
    u_fst = fst_cnt.user_name
    c_fst = 0 //fst_cnt.session_cnt
    t_fst = 0 //fst_cnt.session_time
    sum_of_each = new AsdLogCntSum(year:y_fst,month:m_fst,user_name:u_fst,
acc_num:c_fst,total_time:t_fst)
    list_fst = (0...asd_log_cnts.size()).findAll {

```

```

    asd_log_cnts[it].user_name == u_fst && getSessionYear(asd_log_cnts[it].date_start) ==
y_fst && getSessionMonth(asd_log_cnts[it].date_start) == m_fst
    }
    list_fst.each {
        c_fst += asd_log_cnts[it].session_cnt
        t_fst += asd_log_cnts[it].session_time
    }
    sum_of_each.acc_num = c_fst
    sum_of_each.total_time = t_fst
    asd_log_sums.add sum_of_each
    list_fst.reverse().each { //list_fst.reverseEach OK
        asd_log_cnts.remove(it)
    }

} // while

//asd_log_sums.sort { it.month }
//asd_log_sums.each { println it }
// for Print
prwriter_sum = (new File(fname_sum)).newPrintWriter()
year_min = Integer.parseInt((asd_log_sums.min { it.year } ).year)
year_max = Integer.parseInt((asd_log_sums.max { it.year } ).year)
month_min = Integer.parseInt((asd_log_sums.min { Integer.parseInt(it.month) } ).month)
month_max = Integer.parseInt((asd_log_sums.max { Integer.parseInt(it.month) } ).month)

for (ye in year_min..year_max)
    for (mo in month_min..month_max){
        total_acc_num = 0
        total_acc_time = 0L
        li = asd_log_sums.findAll { Integer.parseInt(it.year) == ye &&
Integer.parseInt(it.month) == mo }
        if (li != null && li.size() !=0) {
            prwriter_sum.println ""
            prwriter_sum.println "  ${ye}年 ${mo}月,利用者名,接続回数,接続時間 ( 秒 ) "
            prwriter_sum.println ""
            li.each {
                total_acc_num += it.acc_num
                total_acc_time += it.total_time
                prwriter_sum.println it }
            prwriter_sum.println ""
            prwriter_sum.println "      ${ye}年 ${mo}月,計,${total_acc_num},${total_acc_time}"
            prwriter_sum.println ""
        }
    }
//asd_log_sums.each { prwriter_sum.println it }
prwriter_sum.close()
asd_log_sums.clear()

println ""
println " Done !"
println " Refer to the file : ${fname_sum} "
println ""

```

(参考)pppaccv2.groovy による結果例 : groovy pppaccv2 ./Data/dt200508

2005 年 8 月,利用者名,接続回数,接続時間 (秒)

2005 年 8 月,rec0290,4,1356
2005 年 8 月,regg0100,19,43794
2005 年 8 月,regg0260,12,17569
2005 年 8 月,regg0320,4,1657
2005 年 8 月,rggg0080,140,3472
2005 年 8 月,rmnn0190,6,2184
2005 年 8 月,rmnn0280,8,9607
2005 年 8 月,rttt0210,2,1312
2005 年 8 月,rtuv0130,30,23337
2005 年 8 月,rtuv0140,2,1527
2005 年 8 月,rtuv0180,4,639
2005 年 8 月,saaa0170,12,5043
2005 年 8 月,sabc0050,2,1491
2005 年 8 月,sacc0260,6,472
2005 年 8 月,sacc0340,5,5327
2005 年 8 月,sddd0040,2,2384
2005 年 8 月,sdef0360,2,1764
2005 年 8 月,sdef0410,16,13181
2005 年 8 月,sdef0490,161,40701
2005 年 8 月,secret,1,338
2005 年 8 月,somename1,40,46920
2005 年 8 月,somename2,58,16064

2005 年 8 月,計,536,240139

2005 年 9 月,利用者名,接続回数,接続時間 (秒)

2005 年 9 月,rggg0080,1,22

2005 年 9 月,計,1,22

著者寸描

氏名 : あおき まさかず

所属・職 : 教育学部人間環境教育講座・教授

教育研究分野 : 基礎物理学,メディア環境

座右の銘 : 特になし,あえて言えば「ときには,ゆるぎ・たるぎ」

趣味 : プログラミング,ソフトテニス,ソフトテニス指導・ボランティア。一層練習に勤しみ
自らの技術の維持・向上とソフトテニス普及に努めたいと思っています。

センターの広報活動

総合情報基盤センター 吉田裕美

平成 17 年度は総合情報基盤センターが主催するセミナーや公開講座をはじめ、センターの紹介を中心に以下のような広報活動を行ってまいりました。

2005 年 3 月 7 日	センターミニ講演会
2005 年 3 月 2 2 日	スーパーコンピューター利用者講習会
2005 年 4 月 2 3 日	香川大学総合情報基盤センター公開講座「エイズと HIV」の開催
2005 年 8 月	キャンパスウォークでのセンターの紹介（計 4 回）
2005 年 1 1 月 2 日	幸町キャンパス見学（香川県立丸亀高等学校）
2005 年 1 1 月 2 4 日	医学部キャンパス見学（香川県立高松西高等学校）
2006 年 1 月 2 6 日	第 2 回 総合情報基盤センター先端研究セミナー 「免疫グロブリン遺伝子の分子進化」

センターミニ講演会は、佐賀大学の渡辺健次先生を講師として迎え、佐賀大学学術情報処理センターについて講演をいただきました。講演内容は学術情報処理センターの概要、センターシステム構成、独自に開発したシステムなど、特にセンターシステムではディスクリート PC を用いた Windows 端末群の構成、さらに、独自に開発したシステムとして利用者認証システム OpenGate についての講演がありました。

スーパーコンピューター利用者講習会では、スーパーコンピューターを利用して研究および教育をされる方（学生を含む）を対象に、NEC からの講師を招いて概要を含めたベーシックコースとベクトル化を含む高速化プログラミング技法の説明が含まれるアドバンストコースの講習会を開きました。

4 月に開かれた公開講座では、センターで行っている先進的研究領域の一つである生命情報科学を、ウィルスを題材として紹介し、今後爆発的増加が予想される HIV 感染を、特に若い世代が身近な問題として考える機会を提供するという趣旨で公開講座を開催しました。

本公開講座については翌日の 24 日の四国新聞の記事にも掲載されました。



公開講座にて講演していただいた五條掘先生

昨年のキャンパスウォークに引き続き、今年も高校生を対象とする香川大学の施設見学コースの一つとして総合情報基盤センターを紹介しました。今年のキャンパスウォークは計4回(8月3、17、24、31日)行われ、当日は幸町キャンパスにて、香川大学における当センターの役割、業務内容、施設の紹介等を集まっていた高校生成に説明しました。



第2回キャンパスウォークの風景



第3回キャンパスウォークの風景

また、11月には、幸町キャンパスでは丸亀高等学校(写真左下)、医学部キャンパスでは高松西高等学校から訪れた高校生達(写真右下)に総合情報基盤センターのネットワーク、教育用PCなどのセンターが提供するサービス、センターの研究内容などを紹介しました。



著者寸描

氏名： よしだ ひろみ

所属・職： 総合情報基盤センター・助教授

教育研究分野： 分子生物学, タンパク質工学, 遺伝子工学

座右の銘： 毎日変わります。ちなみに今日は「忍耐あるのみ」。

趣味： シュノーケリング, ドライブ, 海。四国の海岸を制覇したいと思っています。



研究会等参加報告

第 2 回国立大学法人情報系センター協議会

日時：平成 17 年 6 月 24 日

場所：一橋大学 兼松講堂

総合情報基盤センター 林 敏浩、堀 幸雄、河内一芳

本協議会は午前と午後の 2 部に分かれており、午前の部では、2 つの分科会が開催された。第 1 分科会（計算サーバー系 + 教育システム系）には、林、堀が、第 2 分科会（ネットワーク・システム系 + 情報サービス系）には河内が参加した。午後の部は、開会の挨拶後、「最近の学術情報行政を取り巻く同行」など 2 件の講演があった。次に、各地区報告、分科会報告、情報基盤システムの有効活用と点検・評価指針に関して報告、討論が行われた。その後の総合討論では文部科学省、国立情報研究所への質疑・応答が行われた。特に情報系センターの予算に関して議論があった。また、現在進めている情報系センターの評価アンケート結果の取り扱いについて種々の議論があったが結論は持ち越しとなった。

第 17 回学術および総合情報処理センター研究交流・連絡会議 及び第 9 回学術情報処理研究集会

日時：平成 17 年 9 月 15 ~ 16 日

場所：佐賀大学工学部多目的セミナー室

総合情報基盤センター 林 敏浩

15 日の研究交流・連絡会議は、佐賀大学の渡辺義明学術情報処理センター長を議長に選出し、連絡会議および第 20 回学術および総合情報処理センター長会議への要望の議論を行った。連絡会議では事前アンケート（アンケート内容は研究交流・連絡会議資料として製本配布）に基づき種々のテーマについて活発な議論が行われた。

16 日の研究集会では、17 件のセンター業務等（利用者認証、無線 LAN、遠隔講義・e-Learning など教育支援、スパム・ウィルス対策のメールシステムなど）に関わる研究報告が行われ、活発な質疑・応答がなされた。発表予稿は「学術情報処理研究 No.9」として製本配布された。原著論文のカテゴリもあり、学術色も強いものとなっている。

センタースタッフによる教育活動（平成 17 年度）

1. 学内担当科目等

授業科目等		開講時期	担当者	担当数 / 開講時限数, または担当%
医学部専門基礎科目	情報科学実習	2 年次前期	竹崎・岩間・神鳥・吉田 他 2 名	15/30
医学部基礎医学系科目	分子生物学	2 年次後期	神鳥・吉田・竹崎・岩間	26/26
医学系研究科必修科目	研究ストラテジー	前期	神鳥・吉田	2/12
医学系研究科選択科目	構造生物学特論・実習	通年	神鳥	100%(不定期開講)
農学部専門基礎科目	情報科学	2 年次前期	林・神鳥・竹崎	4/13
医学部専門基礎科目	数学解析入門	2 年次前期	上原正宏	
医学部専門基礎科目	医用統計学	2 年次後期	上原正宏	
医学部専門基礎科目	科学表現論	2 年次前期	上原正宏	オムニバス方式
教育学部専門科目	現代解析学	4 年次前期	上原正宏	
教育学部専門科目	関数論	2 年次前期	上原正宏	
全学共通科目	教養ゼミナール	1 年次前期	上原正宏	
全学共通科目	数学 E	1 年次前期	上原正宏	
全学共通科目	生命科学 D	1 年次後期	上原正宏	
工学部専門科目	計算機アーキテクチャ	3 年次前期	林	
工学部専門科目	テクニカルプレゼンテーション	3 年次後期	林	8/15 (少人数分割授業)
工学部専門科目	卒業研究	3 年次後期 ~	林	3 年次 3 名、4 年次 3 名
工学系研究科専門基礎科目	情報工学論	後期	林	7/15 (英語による講義)

工学系研究科専門科目	知識工学特論	後期	林	1/15
------------	--------	----	---	------

2. 学外非常勤担当科目等

徳島大学医科学教育部プロテオミクス医科学専攻大学院セミナー

神鳥成弘

「X線構造に基づく好熱性放線菌 α -アミラーゼのシクロデキストリン多重認識機構の解明」 Complex Structures of Thermoactinomyces vulgaris R-47 α -Amylase 2 with Cyclodextrins Demonstrate the Multiple Substrate Recognition Mechanism

平成 17 年 7 月 21 日, 14:00 ~ 15:00, 徳島大学分子酵素科学研究センター

放送大学面接授業

林敏浩

コンピュータの構造

平成 17 年 5 月 4 日 ~ 5 日 (5 回), 放送大学香川学習センター

佐賀大学工学部知能情報システム学科専門基礎科目集中講義

林敏浩

計算機組織論 II

平成 17 年 8 月 18 日 ~ 22 日 (15 回), 佐賀大学工学部知能情報システム学科

佐賀大学工学部電気電子工学科専門科目集中講義

林敏浩

アルゴリズム論

平成 17 年 9 月 26 日 ~ 30 日 (15 回), 佐賀大学工学部電気電子工学科

JICA 国別特設「インドネシア:高等教育行政と情報技術」研修

林敏浩

e-Learning I & Demonstration of e-Learning

平成 17 年 10 月 19 日, 佐賀大学

医学部専門基礎科目を担当して

総合情報基盤センター 上原正宏

医学部の専門基礎教育のうち、その科目の名称は時々変更があったりしたのだが、長年にわたり数学解析入門と医用統計学を担当している。統合以後において医学部の教養教育の大幅な改変により、特殊な時間割でこの二つの科目の授業時間が構成されている事を知る人は少ない。特に今年度は統合後に入学し幸町キャンパスで教養教育のほとんどを学んできた学生に対しての授業であった。元来、数学と云う学問を理解する上で、講義を十分に聴く事は当然であるが、本質的に必要不可欠な作業がある。それは演習である。誰にも覚えのある事であろう。今年の大部分の学生は、進級のルールが変わった事もあり、教養教育の中で、数学の科目を履修しないまま、従って当然数学の演習をもしないまま2年生での授業である。1年生のときの反動からだろうが、鉛筆が思うに任せて動かない。内容の理解が、把握がなかなか覚束ないようであ

る。潜在能力はあるのである。昨年度の反省から、この学年にふさわしい内容でと考えて、中身を、少しばかり工夫をして教えているのに、十分に理解されない事での焦りが少しばかり滲む。どうしたものであろうか。その答えは簡単である。数学の内容に馴染ませる時間、演習の時間が必要である。最終目的を変更してでも、理解されない事には意味がない。何度かの演習の時間を与える事で、昨年度と同じ程度の深さ迄、専門基礎の数学の内容が理解されたと思う。

カリキュラムの大幅な変更は、学部が、大学が十分に考えての事であろうが、習う側にも教える側に、大いなる意識改革が必要である事が実感された一年であった。試しではいけないのである。その学年は、その学生は、その年度にしか受講出来ない授業科目である。教える側の我々は、常に最良の授業を提供しなくてはならないと思う。

著者寸描

氏名： 上原正宏

所属・職： 総合情報基盤センター・助教授

教育研究分野：数学 関数論

座右の銘： 悔いのない人生

趣味： 椿栽培 落語鑑賞 将棋 古川柳 自転車 灯台巡り

センタースタッフによる研究活動

センタースタッフおよび指導している学生にはアンダーラインが引いてあります。

1. 原著論文およびプロシーディング (査読あり)

- (1) Abe, A., Yoshida, H., Tonozuka, T., Sakano, Y. & Kamitori, S. (2005). Complexes of *Thermoactinomyces vulgaris* R-47 α -Amylase 1 and Pullulan Model Oligosaccharides Provide New Insight into the Mechanism for Recognizing Substrates with α -(1,6) Glycosidic Linkages. *FEBS J.* **272**, 6145-53.
- (2) Kamitori, S., Akihiro, Iguchi, A., Ohtaki, A., Yamada, M. & Kita, K. (2005). X-ray structures of NADPH-dependent carbonyl reductase from *Sporobolomyces salmonicolor* provide insights into stereoselective reductions of carbonyl compounds. *J. Mol. Biol.*, **352**, 551-558.
- (3) Mizuno, M., Ichikawa, K., Tonozuka, T., Ohtaki, A., Shimura, Y., Kamitori, S., Nishikawa, A. & Sakano, Y. (2005). Mutagenesis and structural analysis of *Thermoactinomyces vulgaris* R-47 α -amylase II (TVaII). *J. Appl. Glycosci.* **52**, 225-231.
- (4) Ichikawa, K., Tonozuka, T., Mizuno, M., Tanabe, Y., Kamitori, S., Nishikawa, A. & Sakano, Y. (2005). Crystallization and preliminary X-ray analysis of *Thermoactinomyces vulgaris* R-47 maltooligosaccharide-metabolizing enzyme homologous to glucoamylase. *Acta Crystallogr. F* **61**, 302-304.
- (5) Miyauchi, M., Hoshino, T., Yamaguchi, H., Kamitori, S. & Harada, A. (2005). A [2]rotaxane capped by cyclodextrin and guest: Formation of Supramolecular [2]rotaxane polymer. *J. Am. Chem. Soc.* **127**, 2034-2035
- (6) Yoshida H., Hensgens C.M.H., van der Laan J.M., Sutherland J.D., Hart D.J. and Dijkstra B.W., An approach to prevent aggregation during the purification and crystallization of wild type acyl coenzyme A: Isopenicillin N acyltransferase from *Penicillium chrysogenum*, *Protein Expr Purif.*, **41**, 61-67, (2005)
- (7) Yokomine, T, Shirohzu, H, Purbowasito, W, Toyoda, A, Iwama, H., Ikeo, K, Hori, T, Mizuno, S, Tsudzuki, M, Matsuda, Y, Hattori, M, Sasaki, Y and Sasaki, H. Structural and Functional Analysis of a 0.5-Mb Chicken Region Orthologous to the Imprinted Mammalian *Ascl2/Mash2-Igf2-H19* Region. *Genome Res.* **15**, 154-65 (2005)
- (8) Matsumoto, T. et al. ----- (Area of Physical Maps and Sequencing)
Iwama, H. et al. ----- (Area of Annotation and Analysis)
Benjamin, B. ----- (Coordination)
The map-based sequence of the rice genome. *Nature* **436**, 793-800 (2005)
- (9) 林敏浩, 渡辺健次, 大谷誠, 田中久治, 岡崎泰久, 林田行雄, 近藤弘樹 (2005). 「高
精細メディア機器と超高速ネットワークを用いた遠隔環境での板書型講義の設計

と運用」 教育システム情報学会誌 22(1), 3-14.

- (10) Fujita, N., Hayashi, T. & Yamasaki, T. (2005). E-learning teaching material support system for public vocational training. Proceedings of ICALT'05, 391-393.
- (11) Lu, S., Yamashita, N., Tominaga, H., Hayashi, T. & Yamasaki, T. (2005). Japanese Learning System for Chinese Native Speakers - Development of Database for Learning Kanji Which Have Difference between Chinese and Japanese -. Proceedings ICALT'05, 536-537.
- (12) Kinugasa, Y., Yamashita, N., Hayashi, T., Tominaga, H. & Yamasaki, T. (2005). Operation-Style Answering in Multimedia Testing System DrillS-M for Kanji Letter Shape Learning. Proceedings of ICALT'05. 647-649.
- (13) Moriya, M., Hayashi, T., Tominaga, H. & Yamasaki, T. (2005). Video Tablet Based on Stereo Camera - Human-Friendly Handwritten Capturing System for Educational Use-. Proceedings of ICALT'05. 909-911.
- (14) Tominaga, H., Kinugasa, Y., Yamashita, N., Hayashi, T. & Yamasaki, T. (2005). DrillS-M : Multimedia Testing System with Operation-Style Answering on Pen Tablet for Kanji Letter Shape Learning. Proceedings of ED-MEDIA2005. 4686-4693.
- (15) Tominaga, H., Kinugasa, Y., Yamashita, N., Hayashi, T. & Yamasaki, T. (2005). DrillS-M : Multimedia Testing System with Operation-Style Answering - Application to Kanji Letter Shape Learning. Proceedings of ICCE2005. 474-481.
- (16) Yoshiro Imai, Makoto Ooga, Daisuke Yamane, Osamu Sadayuki, Yutaka Iwamoto, and Shin'ichi Masuda, "Mobile Phone-enhanced User Interface of Remote Monitoring System", Proceedings of the 4th IEEE International Conference on Mobile Business (ICMB2005@Sydney), pp.63 - pp.68 , 2005
- (17) Yoshiro IMAI, Yuichi Sugiue, Akira Andatsu, Daisuke Yamane, Hirofumi Kuwajima, Shin'ich Masuda, "Remote Control Facilities of Web-based Surveillance System for Electric Power Appliance and Network Camera", Proceedings of the 2nd International Conference on Informatics in Control, Automation and Robotics (ICINCO2005@Barcelona), Vol.1, pp.331 - pp.314, 2005
- (18) Yoshiro Imai, Yuichi Sugiue, Akira Andatsu and Shin'ichi Masuda, "DEVELOPMENT OF WEB-BASED MONITORING SYSTEM WITH DATABASE AND HOMING FACILITIES", Proceedings of the 4th IADIS International Conference on WWW/Internet 2005 (ICWI2005@Lisbon), Vol.2, pp.124 - pp.128, 2005
- (19) Yoshiro Imai, Yuichi Sugiue, Yukio Hori, Yutaka Iwamoto, and Shin'ichi Masuda, "An Enhanced Application Gateway for some Web services to Personal Mobile Systems", Proceedings of the 5th International Conference on Intelligent Agents, Web Technology and Internet Commerce (IAWTIC'2005@Vienna), 2005

(currently a CD-ROM version only, now publishing)

- (20) Hori, Y., Gotoh, T., The Visualization Technique based on Cone Tree for Large Scale Knowledge Base with Semantic Relations, Proc. WSCG'2006, 549-555.

2. 総説・著書

- (1) 今井慈郎 「情報社会とコンピュータ(大学生の新教養科目)」富田眞治・藤井康雄 編著, 昭晃堂, 2005年5月発行 ISBN 4-7856-3153-8 11章共同執筆 (pp.128-pp.143)

3. 学会発表 (招待講演や依頼講演はその旨, 記入してください。)

- (1) 山田貢, 吉田裕美, 神鳥成弘, 十川亜紀子, 水野正浩, 殿塚隆史, 西河淳, 坂野好幸 「好熱性放線菌シクロデキストリン膜輸送に関わる CBP の X 線結晶解析」日本結晶学会年会, 2005年12月7日 (姫路)
- (2) 安部暁美, 吉田裕美, 殿塚隆史, 坂野好幸, 神鳥成弘 「Crystal structures of *Thermoactinomyces vulgaris* R-47 α -amylase 1 (TVA1) complex with pullulan-analog substrates」日本生化学会年会, 2005年10月21日 (神戸)
- (3) 山田貢, 今井嘉紀, 吉田裕美, 高坂新一, 神鳥成弘 「Crystal structure of macrophage/microglia-specific calcium-binding protein Iba1」日本生化学会年会, 2005年10月20日 (神戸)
- (4) 吉田裕美, 神鳥成弘, 大滝証, 松村洋寿, 養王田正文, 大野弘幸, 中村暢文 「Refined structure of Thermophilic cytochrome P450 from *Sulfolobus tokodaii* at 2.0 Å」日本生化学会年会, 2005年10月20日 (神戸)
- (5) 山田貢, 吉田裕美, 神鳥成弘, 十川亜紀子, 水野正浩, 殿塚隆史, 西河淳, 坂野好幸 「好熱性放線菌 *Thermoactinomyces vulgaris* R-47 由来シクロデキストリン結合タンパク質の X 線結晶解析」日本生化学会 - 中国・四国支部例会 -, 2005年5月28日 (松山)
- (6) 岩間久和, 竹崎直子, 五條堀孝, 「転写因子遺伝子における遺伝子上流配列の高い保存度 - ヒト-マウスのオルソログ比較解析」日本遺伝学会第77回大会, 口頭発表, 2005年9月29日 (東京)
- (7) Naoko Takezaki, Shin-ichi Nakamura, Akinobu Okabe, and Osamu Matsushita. "Frequent domain organization changes in the evolution of clostridial collagenases", 2005, Molecular Biology and Evolution Meeting 2005, Auckland, New Zealand.
- (8) Naoko Takezaki, Shin-ichi Nakamura, Akinobu Okabe, and Osamu Matsushita. "Frequent domain organization changes in the evolution of

- clostridial collagenases”, 2005, 日本遺伝学会, 東京.
- (9) 林敏造, 樋口朋美, 林田行雄 (2005). 統合型漢字学習環境 WWW-KALIST における学習者モデルの共有手法. 教育システム情報学会研究報告 19(5). 37-44, 2005年1月22日(東京)
 - (10) 高志修, 富永浩之, 林敏造, 山崎敏範 (2005). 対話的な授業支援のための一問一答式クイズ AQuaAs - ファジー推論による学習者の解答傾向の推定 -. 電子情報通信学会技術研究報告 ET2004-108. 37-42, 2005年3月5日(徳島)
 - (11) 衣笠裕, 山下直子, 林敏造, 富永浩之, 山崎敏範 (2005). 操作式解答に適したマルチメディア試験システム Drills-M - 漢字の字形学習に適した領域と軌跡による解答方式の実現 -. 電子情報通信学会技術研究報告 ET2004-109. 43-48, 2005年3月5日(徳島)
 - (12) 林敏造, 林田行雄 (2005). WWW-KALIST: 漢字学習のための WWW 型統合学習環境. 電子情報通信学会技術研究報告 ET2004-114. 73-78, 2005年3月5日(徳島)
 - (13) 盧颯, 山下直子, 林敏造, 山崎敏範 (2005). 中国語母語話者のための日本語学習システム - 日中語学習教材の分析と整理 -. 電子情報通信学会技術研究報告 ET2004-115. 79-84, 2005年3月5日(徳島)
 - (14) 花房佑馬, 富永浩之, 林敏造, 山崎敏範 (2005). VRシミュレーションによる力学実験のためのエデュテイメント教材 Interlude の提案. 電子情報通信学会技術研究報告 ET2005-12. 7-12, 2005年6月18日(愛知)
 - (15) 大西洋平, 富永浩之, 林敏造, 山崎敏範 (2005). 作業過程を重視した LEGO プログラミング演習支援システム GoalPost の提案. 電子情報通信学会技術研究報告 ET2005-15. 23-28, 2005年6月18日(愛知)
 - (16) 中山迅, 林敏造 (2005). 描画と振り返りによってモデル的な思考を促す反復再生可能型描画システム Polka. 日本理科教育学会全国大会発表論文集 3. 69-70, 2005年8月4-5日(徳島)
 - (17) 大西洋平, 富永浩之, 林敏造, 山崎敏範 (2005). LEGOプログラミングによるゲームコンテスト形式のグループ演習の授業提案. 教育システム情報学会 30周年記念全国大会講演論文集. 273-274, 2005年8月25-27日(金沢)
 - (18) 山崎琴望, 林敏造, 林田行雄 (2005). 多地点間通信に基づく遠隔ゼミ支援システムの構築 ~ 通信制御機構の設計と実装 ~. 教育システム情報学会 30周年記念全国大会講演論文集. 319-320, 2005年8月25-27日(金沢)
 - (19) 大木陽介, 富永浩之, 林敏造, 山崎敏範 (2005). ステレオカメラを用いたビデオタブレットの試作. 教育システム情報学会 30周年記念全国大会講演論文集. 355-366, 2005年8月25-27日(金沢)
 - (20) 盧颯, 山下直子, 富永浩之, 林敏造, 山崎敏範 (2005). 中国語と日本語の差異に着目したドリル型日本語学習システムの開発. 教育システム情報学会 30周年記

- 念全国大会講演論文集. 533-534, 2005年8月25-27日(金沢)
- (21) 林敏浩, 中山迅 (2005). 理科教育を支援する反復再生可能型描画システム Polka ~しおり機能の実装~. 教育システム情報学会 30周年記念全国大会講演論文集. 547-548, 2005年8月25-27日(金沢)
- (22) 中山迅, 林敏浩 (2005). 反復再生可能型描画システム Polka を使う科学的モデルの学習支援(2). 日本科学教育学会年会論文集 29. 603-604, 2005年9月9-11日(岐阜)
- (23) 佐々木康人, 山崎敏範, 富永浩之, 林敏浩 (2005). 文書・絵画閲覧時の視線情報計測. 日本教育工学会第21回全国大会講演論文集. 193-194, 2005年9月23-25日(徳島)
- (24) 衣笠裕, 山下直子, 林敏浩, 富永浩之, 山崎敏範 (2005). 操作式解答によるマルチメディア試験システム Drills M -漢字の字形学習に適した行為型と解答型の分類と実装-. 日本教育工学会第21回全国大会講演論文集. 485-486, 2005年9月23-25日(徳島)
- (25) 高志修, 林敏浩, 富永浩之, 山崎敏範 (2005). 個人適応の一問一答式クイズ AQUAs における学習者の解答傾向の推定. 日本教育工学会第21回全国大会講演論文集. 487-488, 2005年9月23-25日(徳島)
- (26) 花房佑馬, 井上貴史, 富永浩之, 林敏浩, 山崎敏範 (2005). ノンリアリティな視覚的効果を取り入れた高等初等力学のための VR シミュレーション教材 Interlude. 日本教育工学会第21回全国大会講演論文集. 549-550, 2005年9月23-25日(徳島)
- (27) 盧颯, 山下直子, 富永浩之, 林敏浩, 山崎敏範 (2005). 中国語母語話者のための漢字学習ポイントチェッカーの設計と開発. 電子情報通信学会技術研究報告 ET2005-44. 35-40, 2005年10月15日(広島)
- (28) 佐々木康人, 富永浩之, 林敏浩, 山崎敏範 (2005). 文書, 絵画閲覧時における視線情報の計測と分析. 電子情報通信学会技術研究報告 ET2005-47. 51-56, 2005年10月15日(広島)
- (29) 林敏浩, 中山迅 (2005). 反復再生可能型描画システム Polka のしおり機能と実装. 教育システム情報学会研究報告 20(4). 3-6, 2005年11月26日(大阪)
- (30) 中島亮一, 林敏浩, 渡辺健次, 林田行雄 (2005). Vote Based Evaluation を実装した分散共同評価システム DiCES の開発 ~ 情報教育に関する課題の明確化に向けて ~. 電子情報通信学会技術研究報告 ET2005-57. 1-6, 2005年12月17日(福岡)
- (31) 衣笠裕, 山下直子, 林敏浩, 富永浩之, 山崎敏範 (2005). 操作式解答によるマルチメディア試験システム Drills M ~ 領域解答と軌跡解答に対応する行為支援の実装と評価 ~. 電子情報通信学会技術研究報告 ET2005-58. 7-12, 2005年

12月17日(福岡)

- (32)高志修, 林敏浩, 富永浩之, 山崎敏範 (2005). 対話的な授業支援のための個人適応の一問一答式クイズ AQuAs ~ マルチメディア出題における GUI と解答傾向の推定方法の実装 ~. 電子情報通信学会技術研究報告 ET2005 -59. 12 -18, 2005年12月17日(福岡)
- (33)大木陽介, 富永浩之, 林敏浩, 山崎敏範 (2005). ステレオカメラを用いたビデオタブレットの試作と性能評価. 電子情報通信学会技術研究報告 ET2005 -72. 87 -92, 2005年12月17日(福岡)
- (34)杉上裕一, 山根大祐, 今井慈郎, 「Java Web アプリケーションサーバを用いた遠隔監視制御システム」平成17年度電気関係学会四国支部連合大会講演論文集, pp.134 ,2005
- (35)阿部淳也, 大賀 誠, 今井慈郎, 「Java Servlet による遠隔監視制御システムの実装と簡易評価」平成17年度電気関係学会四国支部連合大会講演論文集, pp.135 ,2005
- (36)出石大志, 今井慈郎, 堀 幸雄, 「ローカルネットワーク環境における自律的セキュリティ制御システム」平成17年度電気関係学会四国支部連合大会講演論文集, pp.251 ,2005
- (37)水田明宏, 今井慈郎, 「言語処理過程を可視化するコンパイルプロセスビジュアライザの設計」平成17年度電気関係学会四国支部連合大会講演論文集, pp.247 ,2005
- (38)安達章, 堀 幸雄, 今井慈郎, 「ネットワークカメラを用いた鉄道線路上における自動監視システム」平成17年度電気関係学会四国支部連合大会講演論文集, pp.137 ,2005
- (39)堀 幸雄, 今井 慈郎, 中山 堯, 自己組織化マップを用いたユーザからの興味パターンの抽出, 情報処理学会, 第68回全国大会, 6G -7, 2005.

4. その他(特許, 学会賞等)

- (1) Iwama, H., “Squeezing Genomes: Bioinformatics for Medical Researches” Hot Topics in Medical Science, 講演, 2005年12月8日(香川)
- (2) Iwama, H., H-Invitational (Human Genome Annotation Meeting) Disease Edition, 招待参加 2005年9月14~16日(東京)

反復再生可能型描画システム Polka の開発

総合情報基盤センター 林 敏浩

はじめに

理科において、学習者は自然の事物や現象を対象とした観察と実験を行いながら、それらについて学習する。このような学習の展開を行う方法の一つとして、学習者に観察対象や目に見えない現象を絵として描かせる描画法[1]がある。例えば、昆虫の学習ならば、観察した昆虫を絵に描きながら、昆虫の共通の性質を捉えるといった学習が可能である。

このような描画法に利用できるソフトウェアとして、刈宿が開発した描画のプロセスを再生表示できる再構成型ソフトウェア「脳の鏡」[2]がある。脳の鏡はペイント系ソフトウェアで学習者は自由に描画ができ、脳の鏡を利用した教育実践事例も報告されている。これに対して、我々は科学的モデルの学習に着目する。科学的モデルの学習では、学習者は対象や現象を抽象的な図形に変換して考え、さらにその抽象図形で説明する傾向にある。このような抽象的な図形を取り扱うことを想定すると、ドロー系描画ソフトが描画のために有効であることが期待できる。我々は脳の鏡と同様描画履歴再生機能を持つドロー系描画システム Polka の開発を進めている。

Polka の設計

本章では、理科教育における科学的モデルの学習の特徴を説明し、そのような学習を実現・支援する描画システムが持つべき機能に

ついて述べる。

科学的モデルの学習

絵を描いて学習を行う場合、学習者は対象を抽象的に捉えたり、モデルに基づき現象を予測・説明する。このように、(1)自然の事物・現象を抽象化したモデルで表現し、(2)そのモデルに基づいて予測・説明できるようになる学習を「科学的モデルの学習」と呼ぶ。また、我々は「学習者による描画」を科学的モデルの学習を実現する一手法として捉える。

科学的モデルの学習のための機能

我々は、科学的モデルの学習のための描画システムは以下の機能を持つべきと考える。

1. 円、四角形などの基本図形を描ける。
 2. 線画（自由曲線）をマウスで描ける。
 3. テキスト入力ができる。
 4. 同じ図形を連続して簡単に描画できる。
 5. 図形を移動、削除などの編集ができる。
 6. 描いたり消したりした過程をすべて記憶して、ワンステップずつ、連続的に再生したり逆再生したりできる。
- 1.~5.までは上述の(1)の教育・学習活動を支援するものであり、6.は(2)を支援する機能として、Polka では位置づけている。

Polka の開発

我々は、科学的モデルの学習に利用できる描画システムとして (1)ベクトルデータで図形を扱うドロー形式、(2)描画履歴を再生できる描画ソフト Polka を開発した。

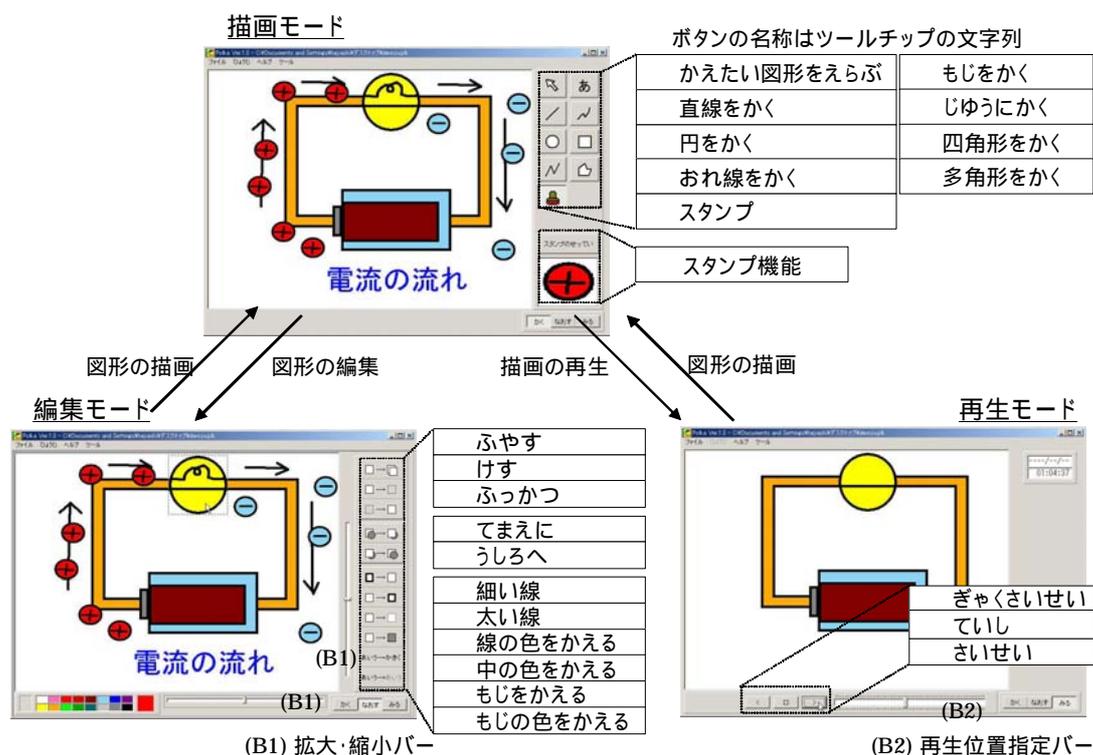


図 1 : Polka (<http://www.eng.kagawa-u.ac.jp/~hayashi/polka/polkaHP.html>)

我々は、Polka の操作を (1) 図形の描画、(2) 図形の編集、(3) 図形の描画履歴の再生の 3 モードに分類し、インタフェースを提供する。Polka のインタフェースを図 1 示す。各モードを「描画モード」、「編集モード」、「再生モード」と呼ぶ。学習者は描画・編集モードで図形を描画し、描画内容の描画手順を再生モードで確認することができる。

おわりに

本稿では、現在、開発・改良を進めている

反復再生可能型描画システム Polka について述べた。なお、本研究は、宮崎大学教育文化学部の中山迅教授との共同研究である。

参考文献

- [1] 中山迅, 稲垣成哲編著:"理科授業で使う思考と表現の道具: 概念地図法と描画法", 明治図書(1998).
- [2] 荻宿俊史:"「プロセスの作品化」による自己理解の深化支援", 日本教育工学雑誌, Vol.24(Suppl.), pp.203-206(2000).

著者寸描

氏名 : はやし としひろ
 所属・職 : 総合情報基盤センター・助教授
 教育研究分野 : 教育工学
 座右の銘 : 未定
 趣味 : 阿波踊り、柔道、酒を嗜むこと



クロストリジウム属のバクテリアの コラーゲン分解酵素の進化

総合情報基盤センター 竹崎直子

クロストリジウム属のバクテリアには、破傷風菌、ボツリヌス菌、ウェルシュ菌など非常に毒性の強いバクテリアがあり、動物の組織を破壊して食中毒、ガス壊疽などを起こします。コラーゲン分解酵素はこれらの菌が病原となる要因の一つとして考えられています。また、動物組織の大部分はコラーゲンで構成されているので、コラーゲン分解酵素はバクテリアの栄養摂取にも役立っていると思われる。

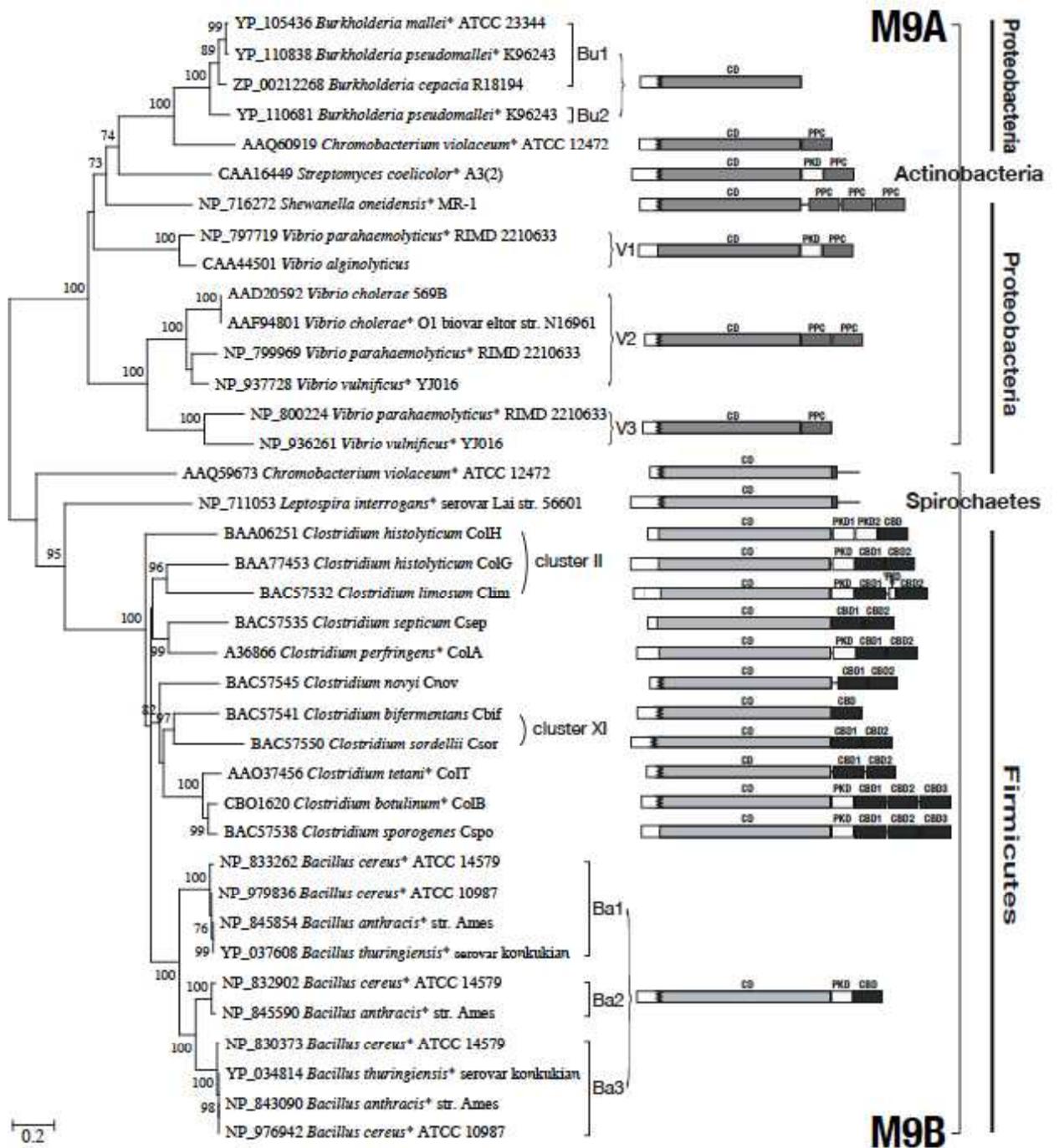
クロストリジウム属のコラーゲン分解酵素には触媒ドメイン(CD)、PKD ドメイン(PKD)、カルシウム結合ドメイン(CBD)の3つのドメインがあることが知られていました。*Clostridium histolyticum*の2種類のコラーゲン分解酵素 *ColG* と *ColH* はそれぞれ CD+PKD+2CBDs、CD+2PKDs+CBD という異なるドメイン構造をもっています。このドメイン構造変化、2種類のコラーゲン分解酵素がどのように出現したのかを知るために、いくつかのクロストリジウム属のバクテリアのコラーゲン分解酵素の DNA の配列データを得て、データベースに登録されている相同な配列データと共に分子系統学的解析を行ないました。

解析の結果は、クロストリジウム属のコラーゲン分解酵素の触媒ドメインに相同な配列データは M9 と呼ばれるタンパク質ファミリーに分類されていますが、M9 ファミリーの酵素が M9A と M9B の2つのサブファミリーに分

かれていることを示しています。M9A に属する酵素は主に proteobacteria のものですが、例外としてらせん菌 *Streptomyces coelicolor*にもあることがわかりました。M9B の酵素はクロストリジウム属とバチルス属のバクテリアがもち、これらの M9B から M9A 酵素からも遠いが、やや M9B の方に近い配列がスピロヘータ、proteobacteria から見つかりました。M9A 配列の3'末端には CBD ではなく PPC というドメインがありますが、これも CBD と同じようにドメインの数が異なる種や配列によって違います。

この研究で得たクロストリジウム属のコラーゲン分解酵素の配列、データベースから収集した配列を合わせて、PKD、CBD のドメインの数の増減の過程を推測しました。この結果、PKD、CBD 共に頻繁に重複による増加、欠失による減少が起きていることがわかりました。*C. histolyticum* の二つのコラーゲン分解酵素 *ColG* と *ColH* がいつ出現したのかははっきりとした結論を得ることはできませんでしたが、クロストリジウム属の進化過程の中では比較的最近である可能性が高いと思われます。

PKD および CBD のドメインの数の増減は、これらのドメインの数によって不溶性、溶解性のコラーゲンへの結合力が異なることが知られているので、それぞれのバクテリアの環境への適応の結果かもしれません。



随想 ～ 数学の研究というもの

総合情報基盤センター 上原正宏

数学の研究というのも、他の学問同様に、見かけほど簡単ではない。しかしながら、そのほとんどは平凡な事実であったり、まるで間違っただけであるのだが...、時として、何かに引かれるように、打たれるように、閃く事がある。それは何時到来するのか分からない。夢の中で、散歩中に、会議の席で、風呂場で、トイレで、予測無しに一時にやって来る。この年齢になると、平常時においてもそのようなのであるが、何故かしら、そのような時の記憶は実に定かでない。むしろ非常に忘れがちである事が多い。それを防ぐために、常に手元にメモ、ノートの類が必要である。もう何冊になるであろうか。題して「新発見問題ノート」。古いノートの部分は流石に懐かしい。若い時の、あの滾るような言葉が、数式が綴られている。年齢60をも過ぎると、自分自身のノートでありながら、時々何を考えていたのか分からない式も記録もある。残念ながら今では、頭が凝り固まっているのだろうか、当時の内容で理解出来ないものすら見受けられる。最近のページには白紙の部分が多い。見事に、長時間にわたり記録がないのである。幾つかの簡単な式の成立についての別

証明がある。新発見の部分もあるが、論文にするほどの内容でもない。困ったものである。とは云え、それらの証明は有史以来誰も気がつかなかった事には違いない(と思っている、否、思いたい)。歳を寄せるとはこういう事なのであろうか。ホンの10年前には分からなかった、気付かなかった事である。

まだまだ、自分自身の分野においてはその種の本来の数学的な(と信じられている)問題を思考し、少しでも完全で美しい関係式の発見に勤める事はもちろんである。しかし、比較的初頭的な内容で、誰もが知っている式や関係の簡単な別証明、ある表現の別の解釈などは、現行の教育の場で十分に役立つものもある。厳格にして崇高な数学といえども、世間のある人にとって、見るのも嫌なものであるかもしれない。その昔、全く別の方法で、非常に理解しやすく教えられていたとしたら、理解していたとしたら、彼の人々の数学観は今とは全く違っていただろうかもしれない。こんな風に考え、若干の新しい発見を、いくつかの別証明などを積極的に教育の場で生かす事を、利用する事を考えている昨今である。

平成 17 年度活動報告記

総合情報基盤センター 今井慈郎

1. はじめに

総合情報基盤センターに学内移籍し、平成 17 年度を向かえた。工学部併任として、就職委員を受諾していたので、工学部での就職支援教員と基盤センター教員との役割分担を取りながらの 1 年となった。本報告記では、主として、学外での発表を中心に、この 1 年間の活動を振り返る。

2. 国際会議

2001 年に、クラコフ工科大学教授(現、名誉教授)の J.Murzewski 先生のご紹介で、ポーランドの古都、クラコフで開催された欧州計算力学国際会議(European Conference on Computational Mechanics 2001)に出席・発表したのがきっかけで、少々強引な形で 5 年目も欧州を中心に学会発表を行うことになった。その間、研究テーマも材料強度データベースおよび余寿命予測から始まって、ハードウェアとソフトウェアを融合した情報処理教育教材の開発と評価、コンピュータの構造・動作の可視化を目的としたシミュレータの開発、遠隔監視制御システムと携帯電話を利用した GUI の開発などいくつかのテーマにわたった。今回、5 年を経過したこともあり、平成 17 年度分の発表および会議参加で経験した異国体験を纏める形で報告記を書くことを思いついた。

2. 1 シドニー(平成 17 年 7 月)

モバイルビジネスに関する国際会議への出席と発表は、2002 年のニュージーランドについて、シドニー(写真 1-1)が会場となり、2 度目の南半球への海外出張となった。どこの国際会議も参加を確保するために、関連するキーワードを基に種々の発表者リストから、言わば客呼びとでも言うべき電子メールによる国際会議参加(論文投稿)の勧誘が盛んである。特に、主催者側の努力は大変で、学生諸君も動員して、関係会議への過去の出席者を閲覧し、候補者へメールを送る作業が大変だと聞いたことがある。筆者自身もこの例に漏れず、主催者であるシドニー工科大学教授の教え子の学生さん(彼女は会議秘書の一人)から勧誘のメールを平成 16 年の末に受け取っていた。



(写真 1-1) シドニータワーからの景観

会議のテーマはモバイル技術および応用に関するものであり、これまでの発表をベースとして、主として、ユーザインターフェースに関するものとなった。7月の半ばと言えば、北半球は夏でも、南半球は言わば初冬であり、比較的温暖で治安の良いシドニーでも、やはりそれなりの寒さだった。出張旅費の出費を少しでも低減するため、ソウル経由でシドニーに向かったが、ソウル(実際は任川)国際空港で、ホストメンバーの一人 R.Steele 教授からセッション座長の打診メールが届いていたのには驚いた。初めての分野で、しかも英語の苦手な筆者には日程が厳しいとお断りの返信を送るのがようやくであった。滞在中には晴天もあって、半日程度の市内見学も行ったが、到着した当日は、冷たい風雨の朝となり、ソウル国際空港での時間待ちの疲れもあって、文字通り暗雲の立ち込めるシドニー到着となった。会議会場の下見もあって、会議開始の前日、会場ホテルまで徒歩で移動したが、傘もさせないほどの強風かつ、少雨ながら冷たい雨模様で、初対面の E.Lawrence 教授(彼女はホストメンバーの一人で筆者の発表セッションの座長)にも自己紹介すらしたいと思わないほどの悪条件だった。実は彼女の学生さんが筆者に勧誘のメールをくれたということを後で聞いたのだが、御礼すら言えなかったのは残念であった。

シドニーは風光明媚な港町で、海鮮料理の美味しいチャイナタウンなどもあって、まさに観光地といった風情だが、会議初日も風雨がやまず、それが幸いしてか、会場の各室には出席者が多かった。また、ホテルの昼食も美味しく、参加者は真面目に国際会議を満喫することとなった。オーストラリアという土地柄か、発表者は皆さんネイティブな英語を操り、出席者もお隣のニージーランドを含めて、オセアニア出身者が多く、治安の良さもあってアメリカ合衆国からの参加者も少なくなかった。どこでも、そうであるが、ここシドニーもアジアからの参加者が多い(日本人としては言いにくいですが、中国本土や韓国からの発表者も多い)。会場の外や談話室でもほとんどの場所で英語が話されており、セッション座長をお断りして良かったと胸を撫ぜた記憶がある。

会議の懇親会はハーバーサイドのイタリアン海鮮レストランで行われ、同席したのはフィンランドの若手研究者と在豪中国系オーストラリア人の研究者であった(写真 1-2)。中国系オーストラリア人の男性は彼の父母の代から在豪であり、なんと漢字を知らないとのことだった。奥さんが日本人であるため、筆者にも親近感があると言ってくれた。一方、フィンランド人の男性は経営学、特にモバイルを中心に据えたビジネスモデルに興味があると話していた。地球の反対側だねという今世界は狭くなったから問題ないと言っていた。NOKIA の話になると、結構力が入っているのが分かり、フィンランド人にとって言わば自信の源なのかなと想像してみた。



(写真 1-2) 懇親会



(写真 1-3) コアラが抱ける動物園

折角のシドニー出張なので、カンガルーとコアラは見ておくべきだと言う周囲の勧めもあって、コアラを抱けるという動物園まで、他の研究者の家族と一緒に、半日観光を行った。確かにコアラはいたが、寒さのためか、ドッジボールのように丸くなっているだけ、ひたすらユーカリを食べている感じで、抱きたいと思ったのは幼い子供だけだったように思えた(写真 1-3)。一方、カンガルーは言わば犬のような感じでユーモラスな体型からくる親近感とは裏腹に抱きしめたいと思う存在とは感じられなかった。オーストラリア博物館でのアボリジニ(豪州ネイティブの人達)の受難の歴史を公開しているが、白人による迫害の歴史を初めて知ってその重大を感じた良い体験だった。

2.2 バルセローナ(平成17年9月)

制御、オートメーションおよびロボティクスにおける情報科学に関する国際会議(以下、ICINCO)への出席と発表は、2004年のマドリードについて、2度目のスペインへの海外出張となった。実は、工学部の服部教授が前回発表された会議なので一緒に予定であったが、生憎の一人旅となった。バルセローナ自体は初めての街であり、スペインでありながら、南仏とも近い得意な地中海を中心とした海洋都市であるなど、一度は足を運びたい街の1つであったため、念願が叶ったことになる。市内には日本でも有名になったアントニオ・ガウディの建築が点在しバルセローナの文化的特徴を際立たせている。特に、傑作と言われるサグラダファミリア教会(写真 2-1、実は未だに完成しておらず、未完成品に訪問者はまたびっくりという印象を与える)やグエル公園(写真 2-2)など、独特な建築物が観光客の目を引く風景が新鮮に映った。

ICINCO は、昨年度ポルトガルのセトゥーバルというリスボン近郊の港町にあるセトゥーバル工科大学が主催する国際会議であり、今回は同じイベリア半島の反対側で隣国スペインに会場を移しての開催となった。筆者の発表は、遠隔監視制御システム、特に監視画像の解析によるネットワークカメラのホーミング制御(遠隔制御)に関するものだった。ICINCO というが会議の性格上、機械制御やプラント制御など、さまざまな内容が発表され、広範な研究者が集まっていた。

本来ならば、発表と同時に他の研究発表への質疑応答にも出向くべきであったが、バルセローナの気候は少し汗ばむほどの快晴続きで、弁解のようだが、筆者に限らず他の研究者も、会議の会場内での親交を深めるよりも、バルセローナ市民あるいは他の訪問者との親交を深める努力に励んだきらいがあった。もちろん、他にも個人的な理由があった。というのも ICINCO は初めての国際会議であるため、知人が少なかったことも一因であったし、後述する盗難に遭ったため、行動がすこぶる制約されたという精神的側面もあった。



(写真 2-1) サグラダファミリア教会

(写真 2-2) グエル公園

(写真 2-3) サン・パウ病院

懇親会には、必ず参加して少なくとも複数人の知人を海外に作るのも国際会議への参加の目的と吹聴してきた筆者にとって、参加費にも、事欠く状況はまさに想定外のことで、他にも思い出があったものの、大変印象の強い(しかし、あまり記憶に留めたいとは思えない)国際会議となった。最後にガウディばかりが注目されるバルセローナの建築であるが、ドメネク・イ・モンタネールというガウディのライバルとも目される建築家のサン・パウ病院(写真 2-3)は特に印象に残った。入院患者にとっては迷惑な話かもしれないが、世界遺産であるため、病院であっても、観光客が訪問することを前提に簡単なパンフレットなども用意されている(説明は英文もあり、他国からの訪問者へのサービスも忘れていない)。「芸術には人を癒す力がある」という信念のもと建築をモンタネールに依頼したという経緯があり、老後をここで過ごすのも悪くないかと思わせる効果もあるのかもしれない、言わば不思議な感覚を覚えたことも印象深かった。

2.3 リスボン(平成17年10月)

ポルトガルは筆者にとって思い出深い国である。というのも、初めて訪れたのは、平成15年(2003年)である。ちょうど、1543年種子島鉄砲伝来から460年後となった。実はもう1年前の2002年のウィーンに話は遡る。ポルトガルのコインブラ大学のA.Lopes先生とウィーンで開催された国際会議(World Congress on Computational Mechanics V)で出会い、友人関係となった。彼は筆者よりも若い研究者としても魅力的だが、とても気さくな人間で、会議開催中ずっと一緒に行動した。歓迎会も懇親会も発表会場もほとんど同じだった。ということで今後も付き合いを続けようということで、再開を期していた。

その翌年、初めてポルトガルに訪れたのも彼に会いたい気持ちも手伝ったからであった。彼の母校でもあるコインブラ大学で再会した時、相変わらずの人柄にとっても嬉しかったのを記憶している。その後、彼との関係もあって、ポルトガルには特別な親近感を持ち、メールでの遣り取り(無類のサッカー好きであるため、FIFA欧州大会での準優勝を喜んだり残念がったりしていたのも好印象の1つ)なども続けていた。ということで、リスボンへの出張は言わば、三度目のポルトガル訪問となった旅であった。ということで、後述するロストバケージに遭遇しようが、好印

象は変わらないというのが筆者の感想である。

もう1件、この会議(WWW および Internet に関する IADIS 国際会議)には、連続3回も出席しているの、正直言って会議会長の P.Isaias 教授を始め、スタッフにも知り合いが多く、会議主催の懇親会でも旧知に会える楽しみがある。その点、ICINCO と比較しても、格段とも言える居心地の良さである。同系統の国際会議に連続して出席することのメリットを説かれたことがあるが、この会議に出席するとその意味を体験することができる。



(写真 3-1) 世界遺産オビドス(城壁の町)

今回はリスボンからバスで2時間ほど離れたやはり世界遺産の街(写真 3-1、周囲を城壁で囲まれた)オビドスで懇親会が開催された。懇親会では恒例になっているが、優秀論文の顕彰があった。実は、懇親会上で向かいあったバーミンガム大学教授 M.Crawford 先生が受賞された(写真 3-2)。偶然、筆者がとった写真には会議会長である Isaias 教授も写真奥に立っている右側の人物として写っている。懇親会では、受賞が話題となり、大いに盛り上がった。ついでながら、Crawford 先生の奥様との親しくなり、自身が写った写真に筆者も写っていたとわざわざ電子メールで送ってもらった(写真 3-3)。



(写真 3-2) Crawford 先生と Isaias 教授



(写真 3-3) 筆者と向かいに Irene さん (Crawford 夫人)

会議も最後の日となり、筆者の行動パターンならば、そっと姿を消して、市内観光でも言いたいところだったが、真面目な Crawford 先生は朝食時に今日はどの発表を聞くから一緒に出ようなどと親切に声を掛けてもらったため、ついに最後まで発表を聞くこととなった。おまけに、Isaias 教授と朝食後で出くわした時、今日の予定はなどと聞かれ、先の話をしたところ、遠隔監視がテーマのセッション座長を見つけたと言われ、ポカンとしている筆者に君の発表テーマと類

似点もあるので、任せたとのことだった。英語は苦手だと断る暇も無く、Crawford 先生からも、是非聞きに行くから会場で会おうと駄目押しまでされて災難の最終日となった。観光地であるポルトガルでこんなに会議潰になったのは3回の訪問において初めての経験となった。

2.4 ウィーン(平成17年11月)

ウィーンは思い出の多い街である。初めて家族で欧州を訪れた時もウィーンが最初の訪問地となった。その後、クラコフに出張した時も、Lopes 先生と初めて会ったのも、ウィーンだった。しかし、いずれも初夏というベストシーズンだったので、今回の11月末という時期は初体験となった。

知的エージェント、WWW 技術およびインターネットコマースに関する国際会議(以下、IAWTIC'2005)への出席と発表は、4度目のウィーン、しかも11月末とは言え、クリスマスシーズン直前のウィーンへの海外出張となった。ウィーンでの国際会議ということで、期待があったのも事実である。ウィーンは特別の感情があり、年末の忙しい時期ながら、是非とも出席したい会議であった。当初、会議参加勧誘の電子メールが届けられた時、この会議が、ICINCO にも類似した、モデリング、制御およびオートメーションのための計算知能に関する国際会議(CIMCA'2005)と共同開催という案内だった。この分野は前述の服部教授の興味を持たれる研究分野だったため、セッション座長として立候補し、セッションごと発表件数を確保することができないか、その1部としての発表を担当できないかという打診から始まった。結果として、服部教授の予定が難しく、セッションを申請する形での会議参加は取りやめとなったが、締め切りにもかかわらず滑り込み久しぶりのウィーン滞在となった。

偶然なことに、CIMCA'2005 には、工学部客員教授のバチコフ教授も発表のため出席しておられ、奇しくも、ウィーンの地で再会することとなった。バチコフ教授はブルガリア出身のため、ウィーンは母国への短期帰省には好都合な場所であるとのことだった。おりしも、琴欧州関の活躍した九州場所の後であり、母国ブルガリアでの評判など、いつもは同じ工学部にいても聞けないような話題で旅先ながら盛り上がり、同教授の付き合いの広さから会議での友人関係もなお一層広がった感がある(写真 4-1)。筆者も発表までには何人か知人が増え、ニュージーランドから来たという研究者 Grey 先生とは一緒に写真を写すこととなった。Grey 先生の風貌とニュージーランド出身という話から、先生がロードオブザリングに出演したかどうかで、周囲が盛り上がり、筆者も記念写真を1枚ねだったことがその経緯である(写真 4-2)。実は、この会議の主催者はオーストリアならぬ、オーストラリアのキャンベラ大学教授陣で Grey 先生もその関係で地球の反対側にやってきたとのことであった。



(写真 4-1) パチコフ先生を中心とした昼食タイム



(写真 4-2) Grey 先生と筆者

会議は11月末までであり、その後、知人を訪ねてオーストリア西部に移動したが、11月末のため、ウィーン市内はクリスマスシーズン直前でいたるところにイルミネーションが飾り付けてあった。時々旅行パンフレットなどでも紹介のある、ウィーン市役所前のクリスマスマーケットを一目みたいと思い、会議での発表後、急いで駆けつけた。日本と違い、イルミネーションは点滅がほとんどなく、静かな雰囲気、観光客(日本人団体客も何件か出会った)も多かった(写真 4-3)。夕方から午後8時過ぎまで屋外に居たので気温は相当低かったと思われるが、不思議と暖かい印象がある。マーケットには様々は出店があったが、甘いチョコレートでできたお菓子(日本ならみたらし団子とでも言うべき存在)をかじりながら、これも甘いワイン?の入ったマグカップを持って回遊するウィーンっ子(と言ってもヨーロッパ人は夜ともなると男女の区別すらつかない状況だったので定かではないが)の真似をして、どちらも試食してみたが、これがウィーンの初冬の味かと思われる味覚だった。



(写真 4-3) ウィーン市庁舎前のクリスマスマーケット

3. トラブル対策

海外旅行にはトラブルはつき物であるが、ここでは、報告記ということで、非学術的内容ながら、読者(が居ればという前提ながら)にとって少しでも役立つ情報を提供し、反面教師としての勤めを果たしたい。

3.1 盗難

盗難はやはり油断と共にやってきた。バルセローナの中心はサンツという名の駅である。ここは多くの観光客とバルセローナ市民が行き交う場所で、スリなども多く、盗難に遭ったとスペイン警察に届けた時も、サンツ駅でそんな状況で財布を持っていては盗難も回避できないとかえって説諭されたほどだった。スペイン人は親切で、個人的には好意を感じる国民性である。しかし、スリに遭った(か、どうかも結果としては不明)から少し印象が悪くなったが、警察官との会話で再び好感を持つに至った。彼はとても筆者のつたない英語をどうにか理解し、客人として扱うことで、少なくとも一人の日本人観光者への適切な対応をしてくれた。説諭の内容で公開すべきは、財布など貴重品は、間違ってもズボンのポケットなど目に付かない場所に持たない、ということである。胸ポケットが一番であるというのが警察官としてのアドバイスだと言っていた。また、届出を出してもらっても本人に戻る保証は少ないが、盗難カード(実は筆者もクレジットカードを2枚、財布に入れていた)での被害を回避する上でも、日付のついた被害証明書が必要だとも教えてくれた。パスポートも持ち歩く必要は無く、警察への被害届出でもパスポートを見せるとは一度も要求されなかった。コピーを持っておいた方が良いこともあるので、できればカラーコピーでも持参した方が良いとも助言してくれた。

3.2 ロストバゲージ

ロストバゲージには、今回、リスボン空港とウィーン空港との2回、相次いで遭遇した感がある。どちらもパソコンは手荷物として機内持ち込みしており、プレゼンテーション自体には最小の影響で、大過なく済ませたことは幸運であった。しかし、リスボン空港でのロストバゲージは、翌日の発表に服装の面ではわずかな影響が出た。そこで、ジャケットと共に、せめてネクタイだけでも手荷物にしよばせることをお勧めしたい。女性の服装には詳しくないので、的確な助言はできないが、男性であれば、ジャケットとワイシャツおよびネクタイがあれば一応形にはなると思われる。もちろん、ライフスタイルとして短パンとティーシャツというアメリカンスタイルで通されるなら、別の意味で問題ないと言える。また、空港のバゲージ紛失カウンタでは、冷静にできればにこやかに対応することをお勧めする。バゲージを紛失したのは多くの場合、航空会社のスタッフであり、空港スタッフ(それも出発ではなく到着の空港のスタッフ)は全く責任はない。その人間に不満をぶつけても事態は改善しないし、むしろ無意味であろう。ここは、相手の責任感に訴えて、予約しているホテルまで(滞在先に)できるだけ早くバゲージを届けてもらう交渉をして、身軽な状況を活用して、早めに空港を後にすることをお勧めしたい。その際大切なことは、

予約したホテルの名前と住所を確実に相手に伝えるため、メモを手持ち荷物に携行しておくことである。できれば、活字で(プリントアウトしたものを)示すことが誤解を避けられるという意味でも重要だろう。実は、シドニー出張の際に、同じ名前のホテル(チェーン)があることを知らず、どちらのホテルか確認するため、慌てて案内所を探した経験がある。そのような時にロストバゲージにでも遭遇したら、まさにアウトであった。くれぐれも滞在先を明確にしておくこと(筆者のようなアバウトな人間は少ないとは思うが)をお勧めしたい。

3.3 フライトキャンセル

ウィーンは4回目の訪問と先にも述べたが、4回にもなれば、いろいろな経験をするようになる。パリ空港からウィーン空港に飛ぶべきフライトが雪のためキャンセルされていたのも初めての経験だった。当然、他のフライトもキャンセルが続出しており、トランスファーロビーはお客で一杯となり、カウンタ前には長蛇の列ができていた。いつもなら、すいすいと移動できる空港もフライトキャンセルとなれば、全く話は別である。どうにかしてパリからウィーンへ移動できないかと殺気だってくるので、余計イライラすることになる。これに追い討ちを掛けたのが、パリまで同行した航空会社のキャビンアテンダントの方からの次の、「うち(エールフランス)は、気象条件が理由でフライトキャンセルになった場合、保証してくれないかもしれません」という一言だった。親切なキャビンアテンダントの方ただただに心配は募るばかりだった。実は以前、1回だけ日本発の時刻が台風を予想して遅れ、結果としてパリ乗り継ぎが叶わず、パリ空港近郊のホテルに臨時宿泊したという経験があった。しかし、この時には、エールフランスのサービスとしてホテル宿泊費用を出してもらった(ホテルクーポンをいただいた)という経緯があった。諦めきれない筆者は、長蛇の列を待ち続け、ようやく自身の順番にこぎつけ、せつせつとウィーンへの乗り継ぎフライトを探してくれるよう要望した。その答えは大変短かった。すなわち、本日のフライトは既に最終便だったので、確保できない。明日の朝なら可能だが、それはトランスファーデスクの扱いではなく、パスポートコントロールを経て外に出て、チェックインカウンタで交渉して欲しいとのことだった。それが分かっていたらこんなに長蛇の列を待つ必要もなかったのにと悔やむが、しかし、まだ問題が解決された訳ではない。慌ててチェックインカウンタに移動したが、ここでも結構長い列ができていた。ようやく、交渉を終え、翌日一番のフライトチケットを入手し、ホテルクーポンを手にした時には、午後10時を過ぎていた。それからホテルへどのようにして移動したかも本来ならば説明すべきかもしれないが、ここでは割愛する。

3.4 警察官による職務尋問

まさかと思うが、人相の問題か、現地警察官から職務尋問を既に2回も受けている。いずれも共通するのは、イタリア人に道を聞かれ、欧州人でもないのに、教えてあげようと妙な義侠心を持ったことが発端である。警察手帳が、二人の間に割って入り、一応、英語で、職務尋問も唐突に始まった。紙幣を見せる、パスポートを見せる、外国人ならホテルはどこか、などと言った話である。イタリア人はどうやら、偽ユーロを東洋人である筆者に売りつけようとしたらしいとい

うのが、今になっての想像である。当然ながら、その時はパニックで、『邦人、欧州で警察の尋問に遭い投獄へ』まどの妙な記事が新聞に掲載されたら罷免されるかもしれないと恐れおののいたのも事実だった。ところがそんな時に限って、パスポートは持っていない(前述したように本来ならコピーでも持参していれば良かったが)。しかも、財布の中には10ユーロ札が2、3枚という少々情けない状態だった。しかも、ホテルの名前を正確に発音できない(これはパニックだからという逃げは卑怯で本来の英語下手が原因)など、見苦しい事態の連続だった。要するに、イタリア人に道を聞かれて、私は英語がわかりません、と正直に答えて、その場を立ち去るのが一番と助言したい。多分、警察もどこか怪しいと踏んで尋問をされると思われるので、人相の点で筆者ほど不利益を受けている方も多くないと思われるので、これが一般論であるとは言えないが。

4. おわりに

本来ならば、これで報告を終わりとすべきであるが、昨年度は研究室の学生さんにも発表の機会をもってもらい、研究室総出で、電気関係学会四国支部連合大会に参加し発表してもらった。写真5-1は大会での発表を終えて集合写真を撮っていただいた時のものである。平成17年度の活動報告という観点から明記させていただいた。文末に付録として、発表内容を記載した。



(写真 5-1) 四国支部連合大会で発表したゼミ学生と筆者

最後に、平成18年度を直前にして、平成17年度を総括するとまだまだ不満な点もあって改善すべき課題が山積している。しかし、総合情報基盤センターの堀幸雄先生には特別の感謝の念を持っている。一番の理由は学生さんへの影響である。指導教員として役目が十分でなかった点を堀先生のご支援とご助言でどうにか乗り越えることができた。この点を明記して深謝の意を示したい。株式会社シーマイクロ代表取締役社長の増田眞一氏には、研究面と共に社会的活動においても、また、筆者個人のメンタルな面でも大変お世話になっている。氏には個人的にも私淑しており、研究室の教え子の採用に際しても過分なご配慮をいただいている。今後もご指導・ご鞭撻を願うばかりである。

センタースタッフのその他の研究会等参加報告

国際医学交流講演会

開催場所 香川大学医学部

開催日時 6月4日(土)

総合情報基盤センター 上原正宏

香川大学とブルネイ国との国際交流を目指し、ブルネイ・ダルサラーム大学講師を招き、第一回目の国際会議が6月4日(土)香川大学医学部大学院棟Fにて開催された。講演期間中は全て英語での会であり、医学部長をはじめ医学部の若手研究者の講演があった。上原は講演会に参加するのみならず本講演会の企画、実施にあたり、医学部教授を補佐し十分にその役割を果たした。

国際医学交流講演会

開催場所 ブルネイ・ダルサラーム大学

開催日時 9月26日(月)

総合情報基盤センター 上原正宏

国際医学交流講演会が9月26日(月)ブルネイ・ダルサラーム大学 Chancellor Hall で開催された。上原はこの講演会に参加し、両大学医学部の今後の国際交流の可能性に寄与した。



国際医学交流講演会

State-of-Art Medical Issues at Kagawa

開催場所 香川大学医学部

開催日時 12月4日(土)

総合情報基盤センター 上原正宏

国際医学交流講演会 State of Art Medical Issues at Kagawaが12月3日(土) 香川大学医学部臨床講義棟 2F で開催された。講演者は、駐日ブルネイ大使、ブルネイ国厚生省総務長官、ブルネイ・ダルサラーム大学医学部長、同講師、香川大学医学部教授数名、助教授数名などであった。上原はこの講演会に参加する事は勿論、講演会の開催に際し、企画構成、実施面で総合的に統括し、中心的な役割を果たした。結果として、両大学医学部のみならず両大学間の今後の国際交流について多大なる貢献をし、大きな成果を上げた。今後の発展が期待されている。



香川大学総合情報基盤センター規則

(趣旨)

第1条 この規則は、国立大学法人香川大学組織規則第18条第2項の規定に基づき香川大学総合情報基盤センター(以下「センター」という。)に関し必要な事項を定める。

(目的)

第2条 センターは、香川大学(以下「本学」という。)における学内共同教育研究施設として情報通信技術(マルチメディア技術及びネットワーク技術を含む。以下同じ。)を活用した基盤の管理運用及びその支援を行い、教育研究及び大学運営に供するとともに、情報通信技術及びその応用に関して高度な研究開発を行い、その発展に寄与すると同時に本学における情報環境の高度化を推進し、また、学内外への情報発信基盤として学内及び地域の連携強化を進めることを目的とする。

(業務)

第3条 センターは、次の各号に掲げる業務を行う。

- (1) キャンパス情報ネットワーク、センターのサーバ類及び情報処理教育システムの構成、維持管理及び利用に関すること。
- (2) 学術研究のための情報通信技術の研究に関すること。
- (3) 学術研究の情報通信技術の活用に関すること。
- (4) キャンパス情報ネットワークを活用した分散システムの構築及び連携に関すること。
- (5) 学内外への情報通信技術を用いた情報システムの構築、提供及び利用に関する

ること。

- (6) 情報通信技術の教育に関すること。
- (7) 情報通信技術を活用した教材作成に関すること。
- (8) 国立情報学研究所、全国共同利用の大型計算機センター及びその他学外の情報ネットワークとの連携に関すること。
- (9) その他大学の情報化推進に関すること。

(部門)

第4条 センターに情報基盤システム部門、情報応用システム部門及び情報教育システム部門を置く。

- 2 情報基盤システム部門は、キャンパス情報ネットワーク及びセンターのサーバ類の構成及び維持管理並びに情報通信技術の基盤となる技術の研究
- 3 情報応用システム部門は、情報通信技術を応用した各種データベースの構築、利用及び支援並びに情報通信技術の応用に関する研究
- 4 情報教育システム部門は、教育用計算機システムの構築及び運用、教育用教材の開発並びに情報通信技術の教育に関する研究

(分室)

第5条 センターは、幸町キャンパス、林町キャンパス及び三木町農学部キャンパスに、それぞれ分室を置く。

- 2 分室に関し必要な事項は、別に定める。

(職員)

第6条 センターに、次の各号に掲げる職員を置く。

- (1) センター長
- (2) 部門長

(3) 専任教員（以下「センター教員」という。）

(4) その他必要な職員
（センター長）

第7条 センター長の任命は、本学専任教授の中から、学長が行う。

2 センター長は、センターの業務を掌理する。

3 センター長の任期は2年とし、再任を妨げない。ただし、任期の末日は、当該センター長を任命する学長の任期の末日以前でなければならない。

4 前項の規定にかかわらず、センター長が辞任をした場合又は欠員となった場合の後任者の任期は、前任者の残任期間とする。
（センター長の選考時期）

第8条 センター長の選考は、次の各号の一に該当する場合に行う。

- (1) 任期が満了するとき。
- (2) 辞任を申し出たとき。
- (3) 欠員となったとき。

2 センター長の選考は、前項第1号の場合には、任期満了の1月以前に、同項第2号又は第3号の場合には、速やかに行うものとする。

（部門長）

第9条 部門長の任命は、センター教員の中からセンター長の推薦に基づき、学長が行う。

2 部門長は、部門の業務を掌理する。

（センター教員）

第10条 センター教員の任命は、センター長の申出に基づき、学長が行う。

2 前項の申出は、第11条に規定する運営

委員会が選考したセンター教員候補者を推薦することにより行う。

（運営委員会）

第11条 センターの管理及び運営に関する重要事項を審議するため、運営委員会を置く。

2 運営委員会に関し必要な事項は、別に定める。

（事務）

第12条 センターの事務は、部門が関係する学部事務部の協力を得て、学術室研究協力グループにおいて処理する。

（雑則）

第13条 この規則に定めるもののほか、センターに関し必要な事項は、別に定める。

附 則

1 この規則は、平成16年4月1日から施行する。

2 センター教員の任命に係る申出は、第10条第2項の規定にかかわらず、香川大学・香川医科大学統合協議会（平成15年4月22日）の合意に基づき行うものとする。

3 初代のセンター長は、第7条第1項の規定にかかわらず、平成16年3月31日に香川大学総合情報基盤センター長であった者とする。

4 初代のセンター長の任期は、第7条第3項の規定にかかわらず、平成17年3月31日までとする。

附 則

この規則は、平成17年6月23日から施行し、平成17年6月1日から適用する。

香川大学総合情報基盤センター運営委員会規則

(趣旨)

第1条 この規則は、香川大学総合情報基盤センター規則第11条第2項の規定に基づき、香川大学総合情報基盤センター運営委員会(以下「運営委員会」という。)に関し必要な事項を定める。

(任務)

第2条 運営委員会は、総合情報基盤センター(以下「センター」という。)の円滑な運営を図るため、次の各号に掲げる事項を審議する。

- (1) 中期目標・中期計画に関する事項
- (2) 重要な規則の制定又は改廃に関する事項
- (3) 重要な組織の設置又は廃止に関する事項
- (4) 教員の選考に関する事。
- (5) 教育研究活動等の状況について自ら行う評価に関する事。
- (6) その他センター長が管理運営及び教育研究に関して必要とする事項

(組織)

第3条 運営委員会は、次の各号に掲げる委員をもって組織する。

- (1) センター長
- (2) センターの部門長
- (3) 各学内共同教育研究施設(センターを除く。)から選出された教員各1人
- (4) 各学部から選出された教員各1人
- (5) 地域マネジメント研究科及び香川大学・愛媛大学連合法務研究科から選出された教員各1人
- (6) 保健管理センターから選出された教員1人
- (7) 医学部附属病院から選出された教員

1人

- (8) 総務・企画部長
- (9) 学術部長
- (10) 教育・学生支援部長

2 前項第3号から第7号までの委員は、学長が任命する。

3 第1項第3号から第7号までの委員の任期は、2年とし、再任を妨げない。

4 第1項第3号から第7号までの委員に欠員が生じた場合の補欠の委員の任期は、前任者の残任期間とする。

(委員長)

第4条 運営委員会に、委員長を置き、センター長をもって充てる。

2 委員長は、運営委員会を招集し、その議長となる。

3 委員長に事故があるときは、あらかじめ委員長が指名した委員が、その職務を代行する。

(議事)

第5条 運営委員会は、委員の過半数の出席がなければ議事を開くことができない。

2 議事は、出席委員の過半数をもって決し、可否同数のときは、議長の決するところによる。

(委員以外の者の出席)

第6条 運営委員会は、必要があるときは、委員以外の者の出席を求め説明又は意見を聴取することができる。

(専門委員会)

第7条 運営委員会に、専門の事項を調査検討するため、専門委員会を置く。

2 専門委員会に関し必要な事項は、別に定める。

(事務)

第8条 委員会の事務は、学術室研究協力グループにおいて処理する。

(雑則)

第9条 この規則に定めるもののほか、運営委員会に関し必要な事項は、運営委員会が別に定める。

附 則

- 1 この規則は、平成16年4月1日から施行する。
- 2 センター教員の選考については、第2条第4号の規定にかかわらず、香川大学・香川医科大学統合協議会(平成15年4月2

2日)の合意に基づき行うものとする。

附 則

この規則は、平成17年6月23日から施行し、平成17年6月1日から適用する。

附 則

- 1 この規則は、平成17年8月10日から施行し、平成17年6月16日から適用する。
- 2 この規則の施行後最初に任命されるアドミッションセンターから選出された教員の任期は、第3条第3項の規定に関わらず、平成18年3月31日までとする。

香川大学情報基盤運営専門委員会規程

(趣旨)

第1条 この規程は、香川大学総合情報基盤センター運営委員会規則第7条第2項の規定に基づき、香川大学情報基盤運営専門委員会(以下「専門委員会」という。)に関し、必要な事項を定めるものとする。

(審議事項)

第2条 専門委員会は、香川大学総合情報基盤センター(以下「センター」という。)に関し、次の各号に掲げる事項を審議する。

- (1) センターの電子計算機システムの運用に関すること。
- (2) キャンパス情報ネットワークの運用に関すること。
- (3) その他センターが行うサービス業務に関すること。

(組織)

第3条 専門委員会は、次の各号に掲げる委員をもって組織する。

- (1) センター長
- (2) 香川大学総合情報基盤センター規則第4条第1項に規定する各部門から選出された教員各1人
- (3) センターの技術職員
- (4) 各学部から選出された教員各1人
- (5) 地域マネジメント研究科及び香川大学・愛媛大学連合法務研究科から選出された教員各1人
- (6) 医学部附属病院医療情報部から選出された教員1人
- (7) 経営管理室企画情報グループから選出された者1人
- (8) 学術室情報図書グループから選出された者1人

(9) 教育・学生支援室学務グループから選出された者1人

(10) その他センター長が必要と認めた者

2 前項第2号、第4号から第6号及び第10号の委員は、それぞれの部局の長の推薦に基づき、学長が任命する。

3 第1項第7号から第9号までの委員は、それぞれの所属課長の推薦に基づき、学長が任命する。

4 第1項第2号及び第4号から第10号までの委員の任期は、2年とし、再任を妨げない。ただし、委員に欠員が生じたときの補欠の委員の任期は、前任者の残任期間とする。

(委員長)

第4条 専門委員会に委員長を置き、センター長をもって充てる。

2 委員長は、委員会を招集し、その議長となる。

3 委員長に事故があるときは、あらかじめ委員長が指名した委員がその職務を代行する。

(委員以外の者の出席)

第5条 専門委員会は、必要があるときは、委員以外の者の出席を求めて説明又は意見を聴くことができる。

(事務)

第6条 専門委員会の事務は、学術室研究協力グループにおいて処理する。

(雑則)

第7条 この規程に定めるもののほか、専門委員会に関し必要な事項は、専門委員会が別に定める。

附 則

この規則は、平成17年6月23日から施行し、平成17年6月1日から適用する。

香川大学総合情報基盤センター利用規程

(趣旨)

第1条 この規程は、香川大学総合情報基盤センター規則第13条の規定に基づき、香川大学総合情報基盤センター(以下「センター」という。)の利用に関し、必要な事項を定めるものとする。

(利用者の資格)

第2条 センターを利用することのできる者は、次の各号に掲げるとおりとする。

- (1) 香川大学(以下「本学」という。)の職員
 - (2) 指導教官が必要と認める本学の学生
 - (3) その他センター長が適当と認める者
- (利用の申込)

第3条 センターを利用しようとする者は、利用課題ごとに所定の事項を記入したセンター利用申請書をセンター長に提出し、その承認を受けなければならない。

2 情報処理教育のためにセンターを利用しようとする場合は、センター教育利用申請書をセンター長に提出し、その承認を受けなければならない。

3 センター長は、前各項の承認をしたときは、課題番号を付して申請者に通知するものとする。

4 前項の課題番号の有効期限は1年以内とし、当該年度を超えることができない。

第4条 センターの利用を承認された者(以下「利用者」という。)は、登録申請書に記載した事項について変更が生じた場合は、改めて前条の規定に基づき、承認を受けなければならない。

(利用時間等)

第5条 センターを利用することのできない日は、次の各号に掲げる日とする。

(1) 土曜日及び日曜日

(2) 国民の祝日に関する法律(昭和23年法律第178号)に規定する休日

(3) 12月28日から翌年の1月4日まで

(4) その他センター長が必要と認めた日

2 センターの利用時間は、原則として平日の午前9時から午後5時までとする。

3 前項に規定する時間外にセンターを利用しようとする者は、あらかじめ所定のセンター時間外利用申請書をセンター長に提出し、その承認を受けなければならない。ただし、センター長が別に定める時間外の利用については、この限りではない。

(不正使用の禁止)

第6条 利用者は、自己の課題番号を他の者に使用させてはならない。

(経費の負担)

第7条 利用者は、当該利用に係る必要な経費を、別に定めるところにより、負担しなければならない。

2 前項の規定にかかわらず、センター長が特に必要と認めたときは、利用に係る経費を免除することができる。

(成果の公表)

第8条 利用者は、センターを利用して行った研究の成果を、論文等により公表するときは、センターを利用した旨を明記し、所定の用紙に論文名及び誌名を記載の上センター長に提出するものとする。

(損害賠償)

第9条 利用者は、故意又は重大な過失により、その使用に係る物品を損傷したときは、その損害を弁償する責めを負わなければならない。

(利用の取り消し等)

第10条 利用者がこの規程に違反し、又はセンターの運営に重大な支障を生じせしめたときは、センター長はその利用の承認を取り消し、又はその利用を停止することができる。

(雑則)

第11条 この規程に定めるもののほか、センターの利用に関し必要な事項は、センター長が別に定める。

附 則

この規程は、平成16年4月1日から施行する。

香川大学キャンパス情報ネットワーク運営規則

(趣旨)

第1条 この規則は、香川大学(以下「本学」という。)キャンパス情報ネットワーク(以下「学内ネットワーク」という。)の管理運営について、必要な事項を定めるものとする。

(定義)

第2条 この規則における用語の意義は、次の各号に掲げるとおりとする。

- (1) 「部局」とは、教育学部(附属教育研究施設及び附属学校を含む。)、法学部、経済学部、医学部、工学部、農学部(附属教育研究施設を含む。)、附属図書館、医学部附属病院、総合情報基盤センター(以下「センター」という。)及び事務局をいう。この場合において、センター以外の学内共同教育研究施設等(以下「施設等」という。)については、施設等の設置場所が幸町キャンパスにあっては事務局、三木町医学部キャンパスにあっては医学部、林町キャンパスにあっては工学部、三木町農学部キャンパスにあっては農学部の部局に含める。
- (2) 「部局長」とは、前号に規定する部局の長をいう。
- (3) 「ネットワーク」とは、コンピュータ、端末装置等を相互に接続するための通信ケーブル及び接続用機器をいう。
- (4) 「学内ネットワーク」とは、基幹ネットワークと支線ネットワークで構成されたものをいう。
- (5) 「基幹ネットワーク」とは、キャンパス間並びにキャンパス内における部局間及び建物間を結ぶネットワークをいう。

(6) 「支線ネットワーク」とは、基幹ネットワークの接続機器に接続される建物内のネットワークをいう。

(区分)

第3条 学内ネットワークは機能上、教育研究用ネットワーク、診療用ネットワーク及び事務用ネットワークに区分するものとする。

(管理責任等)

第4条 学内ネットワークを管理運営するため、総括責任者を置き、総合情報基盤センター長(以下「センター長」という。)をもって充てる。

2 基幹ネットワークは、センターにおいて管理運用し、その管理者はセンター長をもって充てる。

3 支線ネットワークのうち、教育研究用ネットワークは、該当部局において管理運用し、その管理者は当該部局長(以下「部局管理者」という。)をもって充てる。

4 支線ネットワークのうち、診療用ネットワークは、医学部附属病院において管理運用し、部局管理者は医学部附属病院長をもって充てる。

5 支線ネットワークのうち、事務用ネットワークは、経営管理室において管理運用し、その管理者は総務・企画部長をもって充てる。

6 複数部局により一体として運用する支線ネットワークにおいては、関係部局の協議により、当該支線ネットワーク共通の担当者を置くことができる。

(利用者の資格)

第5条 学内ネットワークに接続することができる者は、次の各号の一に掲げる者と

する。

- (1) 本学職員
- (2) 部局管理者が許可した者
- (3) その他センター長が適当と認めた者
(利用の範囲)

第6条 学内ネットワークは、教育研究用、診療用及び事務用以外の目的で利用してはならない。

(雑則)

第7条 この規則に定めるもののほか、学内ネットワークの運営に関し必要な事項は、別に定める。

附 則

この規則は、平成16年4月1日から施行する。

附 則

この規則は、平成17年6月23日から施行し、平成17年6月1日から適用する。

香川大学キャンパス情報ネットワーク運営細則

(趣旨)

第1条 この細則は、香川大学キャンパス情報ネットワーク運営規則第7条の規定に基づき、学内ネットワークの運営に関し、必要な事項を定めるものとする。

(学内ネットワークの管理運用)

第2条 総合情報基盤センター(以下「センター」という。)は、香川大学総合情報基盤センター運営委員会(以下「運営委員会」という。)の策定する運営方針に従い、学内ネットワークの管理運用に関し、次の各号に掲げる業務を行う。

- (1) 基幹ネットワークの良好な動作状態の維持
- (2) 基幹ネットワークと支線ネットワークの接続に関する技術的指導及び支援
- (3) 学内ネットワークに接続する機器のドメイン名及びIPアドレス空間の管理
- (4) 学外ネットワークとの接続に関する業務
- (5) ネットワーク機器及びその接続形態に関しての各部局に対する技術的支援
- (6) その他運営委員会から付託された業務

2 センターは、前項第3号に掲げる機器のドメイン名及びIPアドレスを割り当てる場合、当該部局と協議するものとする。

3 基幹ネットワークを構成する機器は、センター長の許可なく改修等を行ってはならない。

4 教育研究用ネットワークにおける支線ネットワーク上のコンピュータ等を他部局の管理する支線ネットワークに接続する場合は、当該部局間で協議するものとする。

(教育研究用ネットワークの管理運用)

第3条 支線ネットワークのうち、教育研究用ネットワークの管理運用は、運営委員会の策定する運営方針に従い、当該部局において次の各号に掲げる業務を行う。

- (1) 教育研究用ネットワーク利用諸手続
- (2) 教育研究用ネットワークの良好な動作状態の維持及び管理
- (3) その他教育研究用ネットワークの管理運用に関し必要な事項

2 教育研究用ネットワークの管理運用業務を補佐するため、部局担当者(技術・事務)を置き、部局管理者が指名するものとする。

3 この細則に定めるもののほか、教育研究用ネットワークの取扱いについては、別に定める。

(診療用ネットワークの管理運用)

第4条 支線ネットワークのうち、診療用ネットワークの管理運用は、運営委員会の策定する運営方針に従い、医学部附属病院において次の各号に掲げる業務を行う。

- (1) 診療用ネットワーク利用諸手続
- (2) 診療用ネットワークの良好な動作状態の維持及び管理
- (3) その他診療用ネットワークの管理運用に関し必要な事項

2 診療用ネットワークの管理運用業務を補佐するため、部局担当者(技術・事務)を置き、部局管理者(医学部附属病院長)が指名するものとする。

3 この細則に定めるもののほか、診療用ネットワークの取扱いについては、別に定める。

(事務用ネットワークの管理運用)

第5条 支線ネットワークのうち、事務用ネットワークの管理運用は、運営委員会の策定する運営方針に従い、経営管理室において次の各号に掲げる業務を行う。

- (1) 事務用ネットワーク利用諸手続
- (2) 事務用ネットワークの良好な動作状態の維持及び管理
- (3) その他事務用ネットワークの管理運用に関し必要な事項

2 この細則に定めるもののほか、事務用ネットワークの取扱いについては、別に定める。

(基幹ネットワークへの接続)

第6条 基幹ネットワークに支線ネットワークを接続しようとする者は、当該部局管理者に申し出るものとする。

2 部局管理者は、前項の申し出を受け、これを必要と認める場合は、センター長に申請し、相互接続の承認を得て必要な事項の割当を受けるものとする。

3 接続に係わる経費については、支線ネットワーク接続者が負担するものとする。

(教育研究用ネットワークへの接続)

第7条 支線ネットワークのうち、教育研究用ネットワークに機器を接続しようとする者(以下「申請者」という。)は、当該部局管理者に所定の申請書を提出するものとする。

2 所属部局と異なる部局の支線ネットワークに機器を接続しようとするときは、所属部局管理者を経て、接続しようとする支線ネットワークの部局管理者に所定の申請書を提出するものとする。

3 部局管理者は、第1項及び前項に基づく申請が適当であると認めたときは、アドレス番号を付して許可するものとする。

4 部局管理者は、前項の許可をしたときは、センター長に速やかに報告するものとする。

る。

5 教育研究用ネットワークに機器を接続した者が、その利用を取り止める場合は、所属部局の部局管理者又は接続しているネットワークの部局管理者に届け出るものとする。

6 部局管理者は、前項の届け出があったときは、センター長に速やかに報告するものとする。

7 接続及び廃止に係わる経費については、申請者が負担するものとする。

(学内ネットワークの変更)

第8条 基幹ネットワークの変更、支線ネットワークと基幹ネットワーク接続機器との接続形態の重要な変更及び新しいプロトコルの使用は、運営委員会で審議する。

2 部局において教育研究用ネットワーク、診療用ネットワーク又は事務用ネットワークを変更しようとするときは、あらかじめセンター長と協議しなければならない。

(支線ネットワーク内の変更)

第9条 支線ネットワーク内部において、基幹ネットワークの運用に影響を及ぼすような変更を行おうとする者は、事前に部局管理者に申し出るものとする。

2 部局管理者は、前項の申し出を受けた場合、センター長と連絡協議の上、必要と認めたときは、これを承認する。

(学内ネットワークの運用を担当する者の遵守事項)

第10条 センター長、センター職員、部局管理者及び部局担当者は、学内ネットワークを利用する通信の秘密を侵してはならない。

(ネットワーク利用者の責任)

第11条 ネットワーク利用者(以下「利用者」という。)は、部局管理者に許可されたIPアドレス以外のアドレスを用いたネ

ネットワーク機器を接続してはならない。

2 利用者は、学内ネットワークの円滑な運営を妨げないよう、良識をもって利用しなければならない。

3 利用者の故意又は重大な過失により、ネットワークに障害が生じた場合は、責任を負わなければならない。

(接続許可の取消し及び利用の制限)

第12条 センター長又は部局管理者は、利用者がこの細則その他関連する規則等に違反したと判断したときは、そのコンピュータ等の接続許可の取消し又は利用を制限することができる。

2 センター長、部局管理者及び部局担当者は、協議の上、必要に応じて利用者の利用状況を調査することができる。

(維持に関する経費)

第13条 基幹ネットワークの維持に要する経費については、運営委員会の定めるところによる。

2 支線ネットワークの維持に要する経費は、原則として当該部局で負担するものとする。

(損害の補償)

第14条 利用者が学内ネットワークの利用により被った損害は、その原因にかかわらず補償されない。

(雑則)

第15条 この細則に定めるもののほか、学内ネットワークの運営に関し必要な事項は、運営委員会が定める。

附 則

この規則は、平成17年6月23日から施行し、平成17年6月1日から適用する。

2005年開催の会議および行事

2005年(平成17年)	23(土)	公開講座「エイズとHIV」
1月	28(木)	技術交流会(三木町医学部キャンパス)(第1回)
13(木)		ネットワーク勉強会(第2回)
18(火)		平成16年度第3回総合情報基盤センター教員会議
28(金)		第2回情報基盤運営専門委員会第12回NEC連絡会議
2月	5月	
7(月)	23(月)	平成17年度第2回総合情報基盤センター教員会議
15(火)	27(金)	第4回情報基盤運営専門委員会第15回NEC連絡会議
	30(月)	技術交流会(幸町キャンパス)(第2回)
	6月	
	7(火)	平成17年度第3回総合情報基盤センター教員会議
3月	13(月)	理事視察(総合情報基盤センター幸町地区)
1(火)	17(金)	第16回NEC連絡会議
7(月)	24(金)	第2回国立大学法人情報系センター協議会総会(一橋大学)
15(火)	30(木)	平成17年度第4回総合情報基盤センター教員会議
18(金)	7月	
22(火)	15(金)	第5回情報基盤運営専門委員会第17回NEC連絡会議
	26(金)	平成17年度第1回総合情報基盤センター運営委員会
4月	8月	
14(木)	1(月)	平成17年度第5回総合情報基盤センター教員会議
15(金)	3(水)	キャンパスウォーク(第1回)
18(月)		

10 (水)	キャンパスウォーク (第2回)	31 (月)	平成17年第2回度学術情報ネットワークノード担当者会議 (学術総合センター)
14 (日)	夏期休業 (16日まで)		
17 (水)	キャンパスウォーク (第3回)		
24 (水)	キャンパスウォーク (第4回)		
25 (木)	総合情報基盤センターコンピュータシステム仕様策定委員会 (第1回)	11月	
31 (水)	キャンパスウォーク (第5回)	2 (水)	県立丸亀高等学校生徒見学 (総合情報基盤センター幸町地区施設)
9月		4 (金)	総合情報基盤センターコンピュータシステム仕様策定委員会 (第3回)
9 (金)	第6回情報基盤運営専門委員会第18回NEC連絡会議	11 (金)	平成17年度第8回総合情報基盤センター教員会議
10月		25 (金)	第8回情報基盤運営専門委員会第20回NEC連絡会議
7 (金)	総合情報基盤センターコンピュータシステム仕様策定委員会 (第2回) 学術及び総合情報処理センター長会議 (浜松市)	12月	
14 (金)	大型計算機利用大阪地区協議会総会	2 (金)	総合情報基盤センターコンピュータシステム仕様策定委員会 (第4回)
21 (金)	第7回情報基盤運営専門委員会第19回NEC連絡会議	9 (金)	平成17年度第9回総合情報基盤センター教員会議
28 (月)	平成17年度第7回総合情報基盤センター教員会議	16 (金)	第9回情報基盤運営専門委員会第21回NEC連絡会議
		28 (水)	年未年始休業 (1月4日まで)

平成16年度利用申請一覧（研究・事務）

	所 属	氏 名	研 究 課 題	備 考
1	教育学部	青 木 昌 三	物理学に於けるいくつかの問題のシミュレーション	
2	教育学部	佐 藤 明 宏	国語科教育に関する研究	
3	教育学部	安 西 一 夫	情報交換とフーリエ解析の応用	
4	教育学部	松 下 文 夫	マルチメディアの教育効果	
5	教育学部	平 篤 志	多民族社会における少数派コミュニティの特性に関する研究	
6	教育学部	安 東 恭一郎	研究	
7	教育学部	山 本 茂 喜	国語科単元学習における評価法の研究	
8	教育学部	松 島 欣 哉	New England Transcendentalists, 特に Henry David Thoreau の研究	
9	教育学部	加 藤 みゆき	茶の風味に関する研究	
10	教育学部	小 川 育 子	親水性ポリマーの膨潤	
11	教育学部	宮 崎 英 一	計算機による光学パターンの解析	
12	教育学部	内 藤 浩 忠	整数論	
13	教育学部	深 石 博 夫	数学の研究及び通信	
14	教育学部	松 村 雅 文	光散乱の計算と、その星間現象への応用	
15	教育学部	高 井 忠 昌	立方格子における分散曲線	
16	教育学部	黒 田 勉	研究	
17	教育学部	高 橋 尚 志	Photoemission spectrum data の解析と理論計算	
18	教育学部	小 池 和 男	素粒子の世代構造の研究	
19	教育学部	秋 山 智	グラフィックデザインにおけるコンピュータ処理	
20	教育学部	日 野 陽 子	PPP ダイアルアップ	
21	教育学部	森 征 洋	宇宙・地球科学データベース	
22	教育学部	森 征 洋	風の局地性に関する研究	
23	教育学部	西 岡 圭 子	教育方法に関する研究	
24	教育学部	Paul Batten	Ethnography of Compensatory Communicative Strategies	
25	教育学部	阪 根 健 二	生徒指導研究	
26	教育学部	村 山 聡	経済分析	
27	教育学部	松 本 康	教科教育に関する調査研究	
28	教育学部	山 下 直 子	日本語・日本事情教材の作成の資料収集・連絡のため	
29	教育学部	櫻 井 佳 樹	講義室等でのパソコンの利用	
30	教育学部	高 倉 良 一	法学教育に関する情報収集	
31	教育学部	永 尾 智	英語助動詞の歴史的研究	
32	教育学部	青 山 夕 夏	情報交換	
33	附属幼稚園	塩 田 知 子	インターネットと幼稚園教育	
34	附属幼稚園	高 井 忠 昌	幼児教育におけるインターネットの利用	
35	附属坂出小学校	西 浦 雅 弘	情報教育におけるインターネットの研究	
36	教育学部	藤 元 恭 子	幼児の身体表現に関する研究	
37	教育学部	山 田 貴 志	研究	
38	教育学部	小 方 朋 子	障害児教育における地域ネットワーク作り	
39	教育学部	小 椋 博	社会化の理論と余暇研究	
40	教育学部	服 部 兼 敏	心理統計の授業で使用する	
41	教育学部	時 岡 晴 美	生活様式の継承と変化に関する実証的研究	

	所 属	氏 名	研 究 課 題	備 考
42	教育学部	佐々木 信 行	化学データベース作成	
43	教育学部	山 下 智恵子	家庭科授業ファイルの講義室での活用	
44	教育学部	石 川 雄 一	学内 LAN への PPP 接続	
45	教育学部	岡 田 順 直	パソコンを用いた数学教育の開発	
46	教育学部	大 浦 みゆき	物理学実験におけるコンピュータ利用	
47	法学部	松 井 康 浩	地域国際化に関する動向	
48	法学部	池 端 忠 司	インターネットをはじめとするマルチメディアの法制度の研究	
49	法学部	肥 塚 肇 雄	金融取引の適正化の法理	
50	法学部	湯 山 智 之	国連国際法委員会の国家責任条文草案の起草作業に関する情報・研究の検索の為のインターネット閲覧	
51	法学部	尾 崎 壽 子	データ処理	
52	法学部	神 江 伸 介	選挙研究	
53	法学部	松 尾 邦 之	法律関係情報の収集	
54	法学部	石 井 一 也	開発論・平和研究	
55	法学部	堤 英 敬	選挙関連データの収集と分析	
56	法学部	植 松 真 生	国際取引法における I T 関連問題の研究のため	
57	連合法務研究科	松 本 タ ミ	法学における情報処理	
58	連合法務研究科	草 鹿 晋 一	法学教育研究と I T	
59	連合法務研究科	田 淵 浩 二	法学に関する教育・研究	
60	連合法務研究科	吉 成 務	法律実務教育と情報	
61	連合法務研究科	宇都宮 純 一	法科大学院教育用の情報収集	
62	連合法務研究科	横 山 信 二	法科大学院教育用の情報収集	
63	連合法務研究科	小 林 敬 和	学生指導、研究	
64	経済学部	久 松 博 之	計量経済モデルの推定と検定	
65	経済学部	小 宮 一 高	研究にともなう資料収集のため	
66	経済学部	藤 村 和 宏	研究(サービス・マーケティングの研究)	
67	経済学部	木 村 住 枝	経済分析	
68	経済学部	西 尾 富美子	地域社会分析	
69	経済学部	岡 田 ツヤ江	経営分析	
70	経済学部	井 上 貴 照	国際経済分析	
71	経済学部	本 田 道 夫	知的情報処理の研究	
72	経済学部	三 谷 宗 子	U N I X システム管理	
73	経済学部	安 藤 博 子	U N I X の利用	
74	経済学部	長 山 貴 之	日本の予算に関する実証分析	
75	経済学部	曾 道 智	コンフリクトのゲーム理論解析	
76	経済学部	増 井 久 之	情報リテラシー、経営情報システム、電子政府、電子商取引、ビジネスモデル	
77	経済学部	佐 藤 公 一	西洋思想関係の情報検索	
78	経済学部	安 井 敏 晃	米国の保険学研究誌の論文、記事入手	
79	経済学部	佐 藤 幹 子	日本経済分析	
80	経済学部	佐 藤 保 久	研究及び情報収集・検索	
81	経済学部	岡 田 徹太郎	経済政策	
82	経済学部	金 澤 理恵子	教育のサポート	
83	経済学部	細 川 滋	経済分析	
84	経済学部	尾 崎 万 枝	経営分析	

	所 属	氏 名	研 究 課 題	備 考
85	経済学部	堀 啓 造	マーケティングの心理的研究	
86	経済学部	平 井 健 之	Eメールの利用	
87	経済学部	板 倉 宏 昭	研究	
88	経済学部	山 下 隆 資	情報収集	
89	経済学部	瀧 川 一 幸	ドイツ語教材,ドイツ語教育法のための研究	
90	経済学部	松 岡 久 美	経営関連情報の検索のため	
91	経済学部	中 村 邦 彦	情報処理技法の研究	
92	経済学部	喜 田 恵津子	財務分析	
93	経済学部	穴 戸 榮 徳	オペレーションズ・リサーチの応用	
94	経済学部	梶 脇 裕 二	研究	
95	経済学部	高 塚 創	地域成長の数値解析	
96	経済学部	守 矢 信 明	フランス文化圏の資料収集	
97	経済学部	藤 原 敦 志	金融論の検証	
98	経済学部	原 真 志	産業集積に関する研究	
99	経済学部	佐 藤 忍	国際労働力移動の研究	
100	経済学部	原 直 行	経済研究	
101	経済学部	植 木 英 治	Eメール、掲示板	
102	経済学部	稲 田 道 彦	地域文化研究	
103	経済学部	高 木 文 夫	研究	
104	経済学部	斉 藤 和 也	西洋古典学関係の情報収集	
105	経済学部	姚 峰	経済モデル解析	
106	経済学部	星 野 良 明	経済理論の研究	
107	経済学部	宮 脇 秀 貴	現代企業の会計情報システム分析	
108	経済学部	崔 康 植	研究	
109	経済学部	朴 鏡 杓	管理会計研究	
110	経済学部	ラナデ . R . R	数理経済学	
111	経済学部	桑 原 正 行	会計情報検索のため	
112	経済学部	大 野 拓 行	日本経済の分析	
113	経済学部	渡 辺 己	データ処理	
114	経済学部	清 水 真 志	政治経済学(経済原論)の研究	
115	経済学部	関 義 雄	設計品質に関する研究	
116	経済学部	井 上 信 一	日系企業に於ける管理会計の実態	
117	経済学部	LRONG. LIM	経営・統計分析	
118	経済学部	藤 井 宏 史	マクロ経済のシミュレーション	
119	経済学部	犬 飼 知 徳	戦略分析	
120	経済学部	吉 田 誠	人事管理の展開	
121	経済学部	横 山 佳 充	情報処理の教育	
122	経済学部	安 井 修 二	ゴーイングシラバスの構築	
123	地域マネジメント 研究科	三 原 麗 珠	人文・社会・自然科学の研究	
124	地域マネジメント 研究科	三 好 勝 則	地域政策に関する情報の収集	
125	地域マネジメント 研究科	伊 坂 善 明	まちづくり研究のため	
126	地域マネジメント 研究科	西 山 良 明	マーケティング・マネジメント研究	
127	地域マネジメント 研究科	柴 田 友 厚	テクノロジーマネジメントに関する研究	

	所 属	氏 名	研 究 課 題	備 考
128	工学部	石 川 浩	信頼性工学に関する研究	
129	工学部	澤 田 秀 之	研究	
130	工学部	角 田 直 人	高分解能電子顕微鏡像の計算機シミュレーション	
131	工学部	郭 書 祥	マイクロ能動ガイドワイヤカテーテルシステム	
132	工学部	仲 谷 英 夫	地球科学データベース構築	
133	工学部	中 西 俊 介	非線型光学効果の数値計算	
134	工学部	秦 清 治	学外からメールサーバへのアクセスのため	
135	工学部	吉 野 文 雄	研究	
136	工学部	寺 林 優	地球科学データベース構築	
137	工学部	香 川 考 司	研究	
138	工学部	生 越 重 章	モバイルマルチメディアの研究	
139	工学部	加 藤 大志朗	画像処理・ネットワークソフトに関する研究	
140	工学部	堀 川 洋	パターン認識に関する研究	
141	工学部	守 田 秀 則	土地利用モデルの開発	
142	工学部	高 津 義 典	研究	
143	工学部	古 川 善 吾	ソフトウェアの分散開発方式の研究	
144	工学部	山 崎 敏 範	パターン情報処理	
145	工学部	山 崎 敏 範	感性情報処理の研究	
146	工学部	和 田 隆 広	研究	
147	工学部	服 部 哲 郎	パターン情報の処理と認識に関する研究	
148	工学部	呉 景 龍	研究	
149	工学部	吉 田 秀 典	大規模地下空洞掘前解析とその評価	
150	工学部	石 井 知 彦	スピン分極相対論電子状態の解析	
151	工学部	今 井 慈 郎	計算機ネットワーク環境の実証的研究	
152	工学部	今 井 慈 郎	ネットワークセキュリティに関する基礎的研究	
153	工学部	FINN,William Daniel	Foundation Geodyuamics	
154	工学部	井 面 仁 志	学外からのメールサーバへの接続	
155	工学部	井 面 仁 志	学外からのメールサーバへの接続	
156	工学部	白 木 渡	学外からのメールサーバへの接続	
157	工学部	河 原 能 久	研究	
158	工学部	神 崎 正	研究	
159	工学部	井 筒 勝 彦	研究	
160	工学部	角 道 弘 文	研究	
161	工学部	野々村 敦 子	研究	
162	工学部	青 柳 省 吾	研究	
163	工学部	堺 孝 司	研究	
164	工学部	野 田 茂	研究	
165	工学部	松 島 学	研究	
166	工学部	長谷川 修 一	研究	
167	工学部	山 中 稔	研究	
168	工学部	増 田 拓 朗	研究	
169	工学部	土 井 健 司	研究	
170	工学部	森 下 一 男	研究	
171	工学部	守 屋 均	研究	

	所 属	氏 名	研 究 課 題	備 考
172	工学部	藤 田 晃	研究	
173	工学部	垂 水 浩 幸	研究	
174	工学部	北 島 博 之	研究	
175	工学部	安 藤 一 秋	研究	
176	工学部	清 水 秀 明	研究	
177	工学部	岡 本 研 正	研究	
178	工学部	最 所 圭 三	研究	
179	工学部	中 川 清	研究	
180	工学部	丹 治 裕 一	研究	
181	工学部	筑 瀬 靖 子	研究	
182	工学部	尹 禮 分	研究	
183	工学部	Vatchkov Gantcho	研究	
184	工学部	石 井 明	研究	
185	工学部	山 口 順 一	研究	
186	工学部	石 丸 伊知郎	研究	
187	工学部	高 橋 悟	研究	
188	工学部	林 純一郎	研究	
189	工学部	高 本 喜 一	研究	
190	工学部	石 原 秀 則	研究	
191	工学部	能 見 公 博	研究	
192	工学部	三 原 豊	研究	
193	工学部	大 平 文 和	研究	
194	工学部	平 田 英 之	研究	
195	工学部	橋 口 原	研究	
196	工学部	大 上 祐 司	研究	
197	工学部	細 木 真 保	研究	
198	工学部	江 原 隆一郎	研究	
199	工学部	小 川 一 文	研究	
200	工学部	若 林 利 明	研究	
201	工学部	掛 川 寿 夫	研究	
202	工学部	国 重 和 俊	研究	
203	工学部	馮 旗	研究	
204	工学部	小 柴 俊	研究	
205	工学部	品 川 一 成	研究	
206	工学部	田 中 克 志	研究	
207	工学部	江 島 正 毅	研究	
208	工学部	神 垣 良 昭	研究	
209	工学部	伊 藤 寛	研究	
210	工学部	須 崎 嘉 文	研究	
211	工学部	清 田 雄 彦	研究	
212	工学部	小 松 孝 二	研究	
213	工学部	塚 本 一 義	研究	
214	工学部	吉 村 英 徳	学外からメールサーバへのアクセスのため	
215	工学部	富 永 浩 之	情報科学の教育研究	
216	工学部	荒 川 雅 生	研究	

	所 属	氏 名	研 究 課 題	備 考
217	工学部	末 永 慶 寛	海域環境変化の数値シミュレーション	
218	工学部	中 山 千 晶	研究	
219	工学部	岩 部 達 雄	研究	
220	工学部	鶴 町 徳 昭	研究	
221	工学部	河内山 隆 紀	計測データ解析	
222	工学部	岩 本 直 樹	研究	
223	工学部	俵 原 基 吉	事務処理支援業務	
224	工学部	立和名 慎 一	実験実習補助のため	
225	工学部	吉 田 俊 一	実験実習補助のため	
226	工学部	松 本 直 通	実験実習補助のため	
227	工学部	川 口 政 秀	実験実習補助のため	
228	工学部	絹 田 志 穂	実験実習補助のため	
229	工学部	上 路 林 太郎	研究	
230	工学部	横 島 哲	複雑環境乱流場のラージ・エディ・シミュレーション	
231	農学部	早 川 茂	食品蛋白質の生理的・加工的特性の解析	
232	農学部	竹 川 薫	分裂酵母の塩基配列及び遺伝子構造の解析	
233	農学部	鈴 木 晴 雄	マルチの温度効果	
234	農学部	片 岡 郁 雄	果樹生産分野における研究打ち合わせと資料収集	
235	農学部	豊 田 正 範	作物学における統計解析	
236	農学部	永 瀬 雅 啓	生体防御機構に関する研究	
237	農学部	彌 永 孝 一	園芸学実験実習・パソコン入門	
238	農学部	亀 山 宏	環境政策シミュレーション C G E モデルの構築	
239	農学部	麻 田 恭 彦	担子菌類の分子生物学	
240	農学部	山 内 高 圓	鶏腸管の研究	
241	農学部	佐 藤 正 資	生理活性試験結果のデータ処理	
242	農学部	古 本 敏 夫	情報検索等	
243	農学部	別 府 賢 治	果樹生産分野における資料収集	
244	農学部	秋 光 和 也	e-mail	
245	農学部	多 田 邦 尚	E-mail によるパソコン通信, 海洋環境データ処理	
246	農学部	高 村 武二郎	園芸植物の種苗生産に関する研究	
247	農学部	合 谷 祥 一	ゲルのレオロジー的性質の解明	
248	農学部	松 井 年 行	A C C 合成酵素の遺伝子解析について	
249	農学部	深 田 和 宏	D H C P 学内 L A N 接続利用	
250	農学部	渡 邊 彰	生命機能科学実習	
251	農学部	東 江 美 加	情報科学実習の為	
252	農学部	田 中 直 孝	研究成果のデータ処理	
253	農学部	小 林 剛	植物多様性学、情報科学の研究のため	
254	農学部	蓑 輪 雅 好	農業施設環境調節	
255	農学部	末 吉 紀 行	文献検索	
256	農学部	加 藤 尚	研究	
257	農学部	小 杉 祐 介	園芸植物の老化生理に関する研究	
258	附属図書館	片 山 恒 信	メディアプラザ P C 利用・管理	
259	附属図書館	岩 澤 尚 子	図書館システムのメンテナンス	
260	附属図書館	山 本 博 子	メディアプラザ P C 利用・管理	
261	附属図書館	櫛 橋 一 雅	図書館システム管理 メディアプラザ管理	

	所 属	氏 名	研 究 課 題	備 考
262	附属図書館	藤 谷 栄	学生用P Cの管理	
263	附属図書館	斉 藤 純 代	図書受入目録業務	
264	附属図書館	竹 島 雅 美	メディアプラザのメンテナンス	
265	附属図書館	鷓 川 多歌子	他機関のデータベース等利用・メディアプラザ管理	
266	附属図書館	鈴 木 美智子	メディアプラザP C利用・管理	
267	附属図書館	池 田 喜美代	メディアコーナーのメンテナンス	
268	保健管理センター	鎌 野 寛	学外データベース利用、研究データ処理	
269	留学生センター	塩 井 実 香	日本語学・日本語教育に関する調査・研究	
270	希少糖研究センター	高 田 悟 郎	香川大学農学部における情報科学に関する講義及び実習	
271	生涯学習教育研究センター	清 國 祐 二	教養科目の教材研究	
272	総合生命科学実験センター	清 未 知 宏	生命機能科学実験・実習	
273	総合生命科学実験センター	宮 川 勇 人	研究	
274	国際環境法遵守調査研究センター	石 橋 可奈美	国際環境法の遵守調査研究	
275	学務部	近 藤 まゆみ	w e bシラバス入力等共通利用目的	

平成16年度総合情報基盤センターPC利用状況

PCルーム1, 2, 3

		月	火	水	木	金
8:50	前					計量経済学
	後		情報法			
10:20	前		計量経済学		情報管理論	
	後					情報法 経済統計
12:00	前	情報管理論			(特)時系列分析入	授業 調査データ分析
	後					
14:30	前		地域科学概論 演習		法学部3年演習	
	後		演習		法学部3年演習	消費者調査法
16:10	前	情報処理基礎	情報処理基礎	情報処理基礎	情報処理基礎 演習	情報処理基礎 個別演習
	後				演習 プロゼミ	個別演習
17:50	前	統計学				
	後		(特)計量経済学		教養ゼミ	
19:30	前	統計学				授業
	後		(特)計量経済学		教養ゼミ	
21:10	前	統計学				授業
	後		(特)計量経済学		教養ゼミ	
備考		情報処理基礎				

注：この表に記載の授業名は、総合情報基盤センターに通知にあったものである。

平成16年度総合情報基盤センターPC利用状況

教育学部情報処理教室1, 2

		月	火	水	木	金
8:50	前			英語コミュニケーションLR 演習 LJ[6]		
	後			マルチメディア演習(II)		
10:20	前	数学科教育論		マルチメディア演習(Ⅷ)	ゼミ	
	後			総合英語 JE[9]		
12:00	前	シミュレーション物理	数理情報論			
	後					
14:30	前	教育総合セミナー				総合英語 L[1]
	後		マルチメディア演習(Ⅱ)		マルチメディア演習(Ⅲ)	総合英語 L[1]
16:10	前	計算機基礎	マルチメディア演習(Ⅳ)		法とコンピュータ入門	教育統計学
	後	マルチメディア演習(Ⅰ)				英語コミュニケーション検定 演習 E
17:50	前					
	後					
19:30	前					
	後					
21:10	前					
	後					
備考						

注：この表に記載の授業名は、総合情報基盤センターに通知にあったものである。

平成16年度総合情報基盤センターPC利用状況

農学部情報処理室1, 2

		月	火	水	木	金
8:50	前			科学英語B		
	後					
10:20						
10:30	前				情報科学	
	後					
12:00						
13:00	前		生物生産学実験・演習	生物生産学実験・演習		
	後					
14:30						
14:40	前		生物生産学実験・演習	生物生産学実験・演習		
	後					
16:10						
16:20	前					
	後					
17:50						
18:00	前					
	後					
19:30						
19:40	前					
	後					
21:10						
備考		学生実験				

注：この表に記載の授業名は、総合情報基盤センターに通知にあったものである。

平成16年度総合情報基盤センターPC利用状況

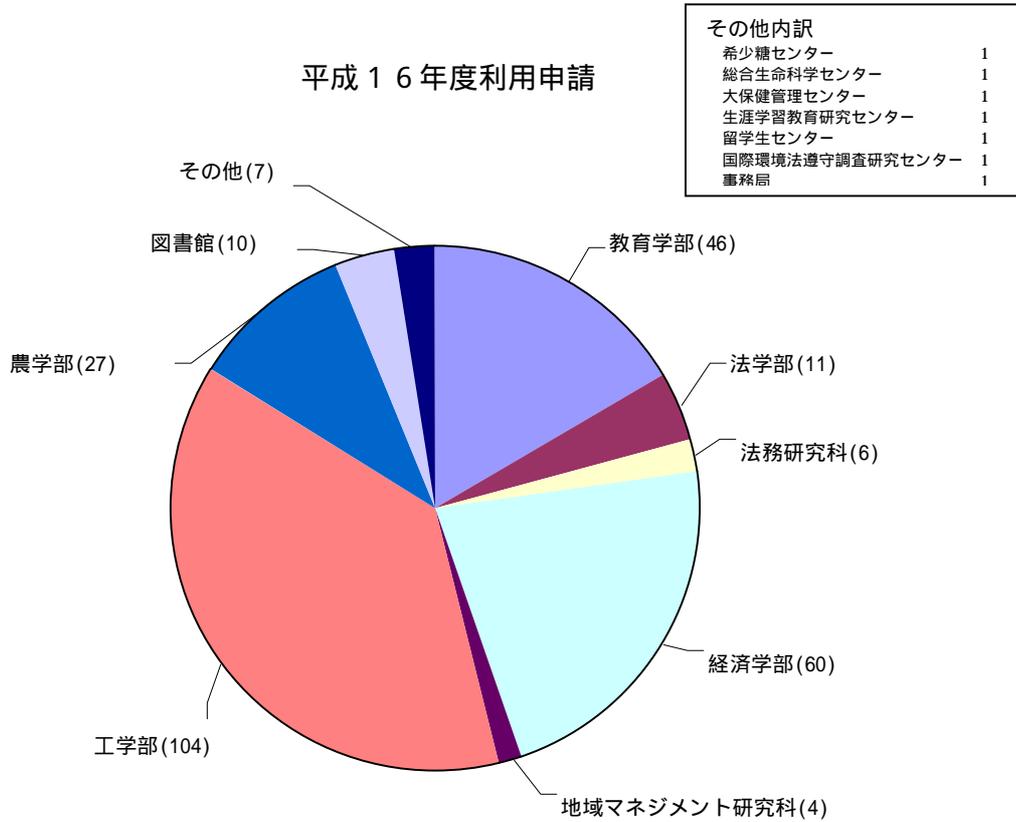
工学部演習室(4301)

		月	火	水	木	金
8:50	前			プログラミング		
	後			プログラミング		
10:20						
10:30	前		計算機材料設計	プログラミング		
	後		知能工学特論	プログラミング		
12:00						
13:00	前		メカトロニクス論	教養ゼミナール		
	後			プログラミング		
14:30						
14:40	前			安全システム建設 工学概論		
	後			プログラミング		
16:10						
16:20	前					
	後					
17:50						
18:00	前					
	後					
19:30						
19:40	前					
	後					
21:10						
備考						

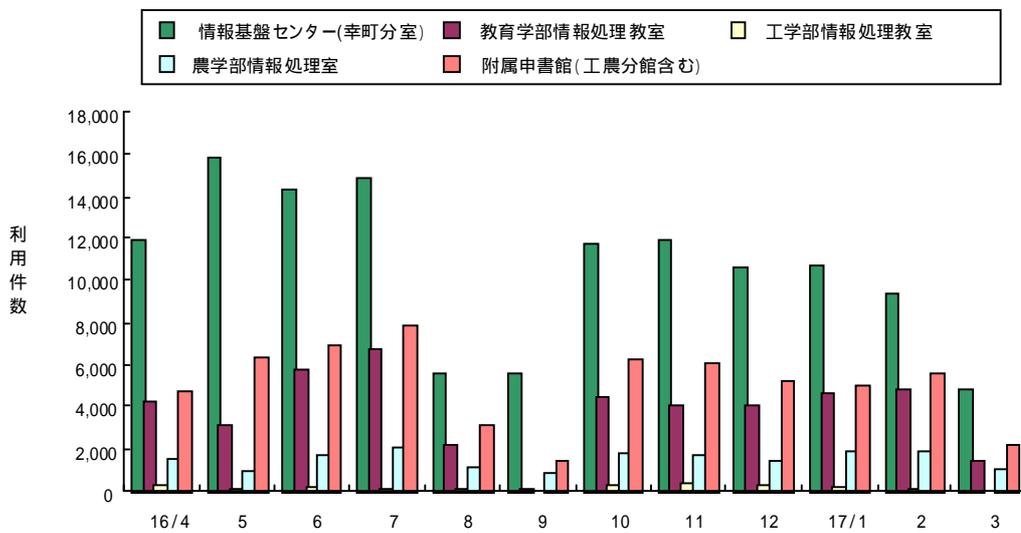
注：この表に記載の授業名は、総合情報基盤センターに通知にあったものである。

計算機利用統計

平成16年度利用申請



平成16年度パソコン室利用状況



平成16年度パソコン室利用統計

総合情報基盤センター幸町分室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	272	362	231	384	489	1,086	345	334	258	421	436	209	4,827
法学部	2,901	2,949	3,138	2,905	1,454	1,241	3,136	2,805	2,524	2,578	2,158	1,435	29,224
経済学部	7,644	11,979	10,277	10,618	2,697	2,345	7,493	7,932	7,262	7,219	6,124	2,762	84,352
工学部	126	85	166	266	155	167	169	167	141	110	199	159	1,910
農学部	335	140	309	385	522	317	322	258	179	153	184	195	3,299
その他	671	290	229	296	286	353	303	421	233	244	236	108	3,670
計	11,949	15,805	14,350	14,854	5,603	5,509	11,768	11,917	10,597	10,725	9,337	4,868	127,282

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	12,309	22,961	13,904	30,052	39,964	85,977	18,572	23,154	18,573	25,973	33,075	16,406	340,920
法学部	135,466	175,333	192,791	194,277	110,998	98,160	161,960	186,316	167,268	201,910	180,496	107,932	1,912,907
経済学部	366,367	902,891	715,009	755,341	196,419	175,762	395,971	513,252	509,981	538,629	444,458	200,499	5,714,579
工学部	6,688	4,234	9,931	19,121	11,903	14,300	10,596	11,138	9,340	8,422	17,249	11,369	134,291
農学部	18,252	6,740	23,071	32,118	55,526	24,827	20,367	21,475	17,008	13,046	16,073	14,679	263,182
その他	11,473	5,557	13,274	20,576	12,536	12,076	18,584	19,076	18,128	17,185	20,367	10,022	178,854
計	550,555	1,117,716	967,980	1,051,485	427,346	411,102	626,050	774,411	740,298	805,165	711,718	360,907	8,544,733

教育学部情報処理教室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	3,323	2,158	4,388	4,827	1,569	37	3,422	3,088	2,895	3,481	3,622	1,055	33,865
法学部	21	19	351	540	76	0	320	346	389	371	335	84	2,852
経済学部	163	859	536	681	181	0	378	419	488	442	528	129	4,804
工学部	373	46	245	278	86	0	84	94	79	96	189	101	1,671
農学部	119	37	107	177	116	0	56	41	37	50	66	36	842
その他	66	13	131	215	124	65	138	136	220	227	142	19	1,496
計	4,065	3,132	5,758	6,718	2,152	102	4,398	4,124	4,108	4,667	4,882	1,424	45,530

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	154,889	146,674	321,951	364,703	135,379	1,687	208,947	209,009	211,464	280,162	310,320	78,528	2,423,713
法学部	11,510	1,423	27,252	42,864	7,788	0	25,468	26,398	33,441	34,275	34,343	7,518	252,280
経済学部	16,694	70,442	40,963	63,643	16,029	0	22,534	34,849	44,090	44,451	55,932	10,469	420,096
工学部	7,850	2,002	15,956	19,122	5,752	0	3,800	6,529	5,484	6,901	16,308	6,023	95,727
農学部	4,004	1,770	6,679	12,195	13,202	0	3,175	1,963	3,449	5,278	6,602	2,968	61,285
その他	4,231	874	8,353	20,989	4,260	529	11,640	10,387	21,183	20,122	12,590	1,845	117,003
計	199,178	223,185	421,154	523,516	182,410	2,216	275,564	289,135	319,111	391,189	436,095	107,351	3,370,104

工学部情報処理教室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	0	0	0	0	0	0	0	0	0	0	0	0	0
法学部	0	0	0	0	0	0	0	0	0	0	0	0	0
経済学部	0	0	1	0	0	0	0	0	0	0	0	0	1
工学部	85	67	141	82	14	15	212	259	311	134	86	1	1,407
農学部	0	0	0	0	0	0	0	0	0	0	0	0	0
その他	0	162	2	1	78	0	91	88	13	7	1	17	460
計	85	229	144	83	92	15	303	347	324	141	87	18	1,868

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	0	0	0	0	0	0	0	0	0	0	0	0	0
法学部	0	0	0	0	0	0	0	0	0	0	0	0	0
経済学部	0	0	27	0	0	0	0	0	0	0	0	0	27
工学部	7,088	1,341	7,564	1,226	759	826	28,282	25,651	29,244	12,938	3,743	1	118,663
農学部	0	0	0	0	0	0	0	0	0	0	0	0	0
その他	0	143	195	51	73	0	1,297	983	870	704	58	2,847	7,221
計	7,088	1,484	7,786	1,277	832	826	29,579	26,634	30,114	13,642	3,801	2,848	125,911

農学部情報処理室

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	3	0	0	0	1	1	5	7	7	9	4	3	40
法学部	6	0	9	0	3	4	7	4	3	1	8	12	57
経済学部	0	4	1	2	0	0	4	12	8	11	2	2	46
工学部	1	1	5	1	2	0	1	0	2	2	0	0	15
農学部	1,364	828	1,608	1,958	1,044	830	1,637	1,532	1,410	1,825	1,790	947	16,773
その他	135	65	58	68	67	35	70	97	26	64	57	35	777
計	1,509	898	1,681	2,029	1,117	870	1,724	1,652	1,456	1,912	1,861	999	17,708

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	112	0	0	0	159	129	531	434	283	596	406	35	2,685
法学部	262	0	221	0	349	111	276	121	266	82	1,588	470	3,746
経済学部	0	568	117	94	0	0	58	700	680	783	59	268	3,327
工学部	108	108	396	63	106	0	3	0	7	30	0	0	821
農学部	63,200	54,661	91,379	138,537	100,788	49,120	82,562	98,455	99,701	143,683	193,768	74,974	1,190,828
その他	1,566	942	1,455	7,140	1,539	587	2,360	2,426	1,475	6,150	4,739	1,758	32,137
計	65,248	56,279	93,568	145,834	102,941	49,947	85,790	102,136	102,412	151,324	200,560	77,505	1,233,544

附属図書館(工学部及び農学部分館含む)

利用件数

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	505	674	807	954	445	198	711	718	510	657	657	163	6,999
法学部	587	858	976	1,004	378	201	874	806	705	802	1,007	647	8,845
経済学部	1,289	2,483	1,889	2,137	633	310	1,541	1,746	1,410	1,431	1,450	383	16,702
工学部	844	736	1,201	1,521	393	113	1,371	1,365	1,218	897	965	190	10,814
農学部	1,078	1,421	1,536	1,683	912	443	1,237	1,039	869	850	1,122	642	12,832
その他	375	217	499	564	373	121	502	406	418	375	357	168	4,375
計	4,678	6,389	6,908	7,863	3,134	1,386	6,236	6,080	5,130	5,012	5,558	2,193	60,567

延べ利用時間(分)

	16/4	5	6	7	8	9	10	11	12	17/1	2	3	計
教育学部	15,748	31,816	40,435	49,340	27,766	12,760	33,603	37,468	27,587	36,360	37,284	8,053	358,220
法学部	17,762	40,573	43,860	42,283	18,523	9,270	34,731	40,042	34,130	44,197	49,198	29,889	404,458
経済学部	45,999	129,155	96,878	106,443	31,087	17,442	71,348	89,403	82,122	88,468	68,295	16,394	843,034
工学部	25,905	26,969	44,226	56,863	16,753	5,357	41,175	40,267	36,663	28,202	35,842	7,754	365,976
農学部	44,924	60,065	69,040	76,414	47,178	21,138	48,593	44,741	38,237	41,574	51,234	37,614	580,752
その他	17,041	15,076	30,797	34,566	16,665	10,245	24,533	28,004	31,137	25,313	23,562	13,192	270,131
計	167,379	303,654	325,236	365,909	157,972	76,212	253,983	279,925	249,876	264,114	265,415	112,896	2,822,571

センタースタッフ，運営会議，運営専門委員会メンバー一覧表

(平成18年2月1日現在)

総合情報基盤センター運営委員会

中村邦彦	総合情報基盤センター長(併)
林敏浩	助教授(総合情報基盤センター情報基盤システム部門)
神鳥成弘	教授(総合情報基盤センター情報応用システム部門)
竹崎直子	教授(総合情報基盤センター情報教育システム部門)
清國祐二	助教授(生涯学習教育研究センター)
武重雅文	教授(大学教育開発センター)
LIM LRONG YEW	助教授(留学生センター)
武藤英一	教授(地域開発共同研究センター)
宮下信泉	助教授(総合生命科学実験センター)
高田悟郎	助教授(希少糖研究センター)
青木昌三	教授(教育学部)
堤英敬	助教授(法学部)
中村邦彦	教授(経済学部)
木村正司	助教授(医学部)
古川善吾	教授(工学部)
蓑輪雅好	教授(農学部)
穴戸榮徳	教授(地域マネジメント研究科)
鹿子嶋仁	助教授(香川大学・愛媛大学連合法務研究科)
久郷敏明	助教授(保健管理センター)
横井英人	講師(医学部附属病院)
真鍋芳樹	教授(アドミッションセンター)
和田喜樹	総務・企画部長
渋澤知祥	学術部長
酒井和博	教育・学生支援部長

情報基盤運営専門委員会

中村邦彦	総合情報基盤センター長(併)
林敏浩	助教授(総合情報基盤センター情報基盤システム部門)
神鳥成弘	教授(総合情報基盤センター情報応用システム部門)
竹崎直子	教授(総合情報基盤センター情報教育システム部門)
河内一芳	技術専門職員(併)(総合情報基盤センター)

瀬野 芳孝	技術専門職員（総合情報基盤センター）
曾根 計俊	技術専門職員（総合情報基盤センター）
青木 昌三	教授（教育学部）
黒田 勉	助教授（教育学部）
堤 英敬	助教授（法学部）
鹿子嶋 仁	助教授（香川大学・愛媛大学連合法務研究科）
中村 邦彦	教授（経済学部）
穴戸 榮徳	教授（地域マネジメント研究科）
木村 正司	助教授（医学部）
古川 善吾	教授（工学部）
最所 圭三	教授（工学部）
香川 考司	講師（工学部）
蓑輪 雅好	教授（農学部）
原 量宏	教授（医学部附属病院医療情報部）
名尾 憲次	チーフ（経営管理室企画情報グループ）
片山 恒信	サブリーダー（学術室情報図書グループ）
大塚 一行	チーフ（教育・学生支援部学務グループ）

総合情報基盤センタースタッフ

中村 邦彦	センター長（併）
林 敏浩	助教授（情報基盤システム部門長）
今井 慈郎	助教授（情報基盤システム部門）
岩間 久和	助教授（情報基盤システム部門）
神鳥 成弘	教授（情報応用システム部門長）
上原 正宏	助教授（情報応用システム部門）
吉田 裕美	助教授（情報応用システム部門）
竹崎 直子	教授（情報教育システム部門長）
堀 幸雄	助手（情報教育システム部門）
河内 一芳	技術専門職員（併）
瀬野 芳孝	技術専門職員
曾根 計俊	技術専門職員
池田 裕子	時間職員（事務）
川田 延枝	時間職員（技能）
塩崎 洋子	時間職員（事務）

編集後記

今冬、日本は、記録的な寒波と大雪で数々の被害がもたらされました。被災地の方々にはお見舞い申し上げます。ここ讃岐の地にあっても、何度か雪が降り、「温暖な瀬戸内気候」とはかけはなれた冬となりましたが、皆様いかがお過ごしだったでしょうか。

さて、皆様のおかげをもちまして、総合情報基盤センター年報第3号を発行できることとなりました。特に、執筆者の皆様には、年度末の多忙な時期、本稿への執筆に時間を割いていただき、誠にありがとうございました。

本号から、CD-ROM版、WEB版を主体として、従来からの印刷媒体は最小限にとどめました。また、印刷物はダイジェスト版となっておりますので、印刷物をお読みになった方は、ぜひ、CD-ROM版、WEB版（当センターホームページ）もお読みになってください。より多くの、より詳細な記事があります。来年度以降、どうするかは未定ですが、本号の評判を参考に決めていきたいと考えています。本号をお読みになられた方は、どうぞ、忌憚のない感想・意見を当センターまでお寄せいただければ幸いです。

本号にも執筆いただきました青木昌三先生（教育学部・教授）におかれましては、今春、定年をおむかえになられます。青木先生は、旧香川大学情報処理センターから長年にわたり当センターの発展にご尽力いただきました。ここに厚く御礼申し上げます。また、少し前になりますが、初代総合情報基盤センター長の乗松尋道先生（医学部・教授）におかれましては、平成17年3月に定年をおむかえになられました。乗松先生は、旧香川医科大学情報メディアセンター時代から当センターの発展にご尽力いただきました。ここに厚く御礼申し上げます。今後の、両先生のご活躍を祈念いたします。

平成 18年 早春

編集者一同

香川大学総合情報基盤センター
<http://www.itc.kagawa-u.ac.jp/>

年報 第3号

2006年2月発行

編集 香川大学総合情報基盤センター

発行 香川大学総合情報基盤センター

〒761-0793 木田郡三木町池戸1750-1

087-891-2266 (ダイヤルイン)

〒760-8523 高松市幸町2-1

087-832-1292 (ダイヤルイン)
