報告：国際ワークショップ「札幌-IAEA核データ研究会2009」

Report on International Workshop

“Sapporo-IAEA Nuclear Data Meeting 2009”

北海道大学 大学院理学研究院
加藤 幾芳
村上 英樹
Kato Kiyoshi
Murakami Hideki
Graduate School and Faculty of Science, Hokkaido University

主催：北海道大学

趣旨：IAEAを中心にした国際核データベースの作成は，1969年に開発されその後多くの改良がなされてきたEXFOR（Exchange Format）を用いて行われてきている。そのデータベースEXFORは、Fortran言語で開発されたシステムをベースにしたもので、最近の様々なコンピュータ環境の進展からみるとかなり古く、利用しにくいものになってきている。とくに、核データの対象も、当初の原子炉での利用を中心とする中性子データから、医療、農業、工業など広い分野にわたる放射線利用に伴う核データや、宇宙における元素合成核反応データなど、扱うデータの種類が広がってきている。そのような進展に伴って、EXFORの改善では不十分になってきており、新たなデータベースを作成する必要性が議論されるようになってきた。

そこで、IAEAのEXFORを中心とした各種ソフトウェアの管理を担当している V. Zerkin 氏を招聘して、新しい核データベースの構築について、情報交換・問題点の整理検討を行なう国際ワークショップを開催することとした。本報告は、その様な目的で開催された「札幌-IAEA 核データ研究会」の概要を記録したものである。

開催日：2009年3月16日（月）－19日（木）

会場：北海道大学　ファカルティハウス「エンレイソウ」第２会議室
北海道大学　ベンチャー・ビジネス・ラボラトリー会議室（17日午前のみ）
Site A: Meeting Room 2, "ENREISOU", Faculty House
Site B: VBL Office in the Faculty of Engineering, Meme Media Lab.

参加者：

Viktor Zerkin (IAEA), Naohiko Otsuka (IAEA), Kiyoshi Kato (JCPRG), Masaki Chiba (JCPRG), Hiroshi Masui (JCPRG), Tomoaki Togashi (JCPRG), Toshiyuki Katayama (JCPRG), Yoshiharu Hirabayashi (JCPRG), Tomomasa Asano (JCPRG), Naoya Furutachi (JCPRG), Tooru Yoshida (JCPRG), Hideki Murakami (JCPRG), Takuma Matsumoto (Kyushu Univ.), Yuzuru Tanaka (Meme Media Lab.), Jun Fujima (Meme Media Lab.), Mikael Nicander Kuwahara (Meme Media Lab.),

PROGRAM（プログラム）：

March 16 (Mon.) ----------

■ AFTERNOON 1 [01:45 – 03:30 pm](Site A: Meeting Room 2, "ENREISOU", Faculty House)

* Welcome address (K.Kato)
* Confirmation of schedule (K.Kato)
* Introduction of JCPRG members (K.Kato)
* "Nuclear Data System developed at Kitami Inst. of Tech." (H.Masui) 資料１

■ AFTERNOON 2 [03:45 pm]
(JCPRG Steering Committee)

March 17 (Tue.) ----------

■ MORNING [10:00 – 12:30 am](Site B: VBL Office in the Faculty of Engineering)

* "Meme Media and Knowledge Federation " (Y.Tanaka) 資料２
* Technical details (J.Fujima, M.N.Kuwahara)

■ AFTERNOON [02:30 – 05:30 pm](Site B: VBL Office in the Faculty of Engineering)

* Technical details (J.Fujima, M.Kuwahara)
* Discussion on “Application of IntelligentPad technology to IAEA Web EXFOR retrieval system” （All members）

March 18 (Wed.) ----------

■ MORNING [10:00 - 12:00 am](Site A: Meeting Room 2, "ENREISOU", Faculty House)

* "EXFOR. Output for users. Proposal for standard output. Discussion about XML" (V.Zerkin) 資料３
* "Application of XML for future EXFOR software" (M.Chiba) 資料４
* Comments (N.Otsuka)

■ AFTERNOON [03:30 – 05:50 pm](Site A: Meeting Room 2, "ENREISOU", Faculty House)

* Demonstration of IAEA Nuclear Data Service: EXFOR + ENDF Web interface (V.Zerkin)

■ DINNER [06:00 pm](Site A: Meeting Room 2, "ENREISOU", Faculty House)

March 19 (Thu.) ----------

■ MORNING [10:00 – 13:00 pm](Site A: Meeting Room 2, "ENREISOU", Faculty House)

*“Interface with Evaluated Nuclear Data File and Covariance data”(T.Togashi)
* Discussion for Conclusion of the meeting.

■ AFTERNOON [02:45 pm -](Site B: VBL Office in the Faculty of Engineering)

* Technical Discussion (V.Zerkin, M.N.Kuwahara)

講演のうち４つのスライドを資料として添えてある。

資料１

Nuclear Data System Developed at Kitami Institute of Technology

Hiroshi Masui

Information Processing Center, Kitami Institute of Technology

Aim of this work

Charged particle reactions in Japan are complied by JCPRG

The complied data are formatted as NRDF

The charged reaction data are important, not only for nuclear physics, but also for enginnering, medication, and other research fields related to nuclear reactions

Therefore, in this work

We support to utilize NRDF in the view point of Information technology

We develop a system for data retrieving based on the SQL-server and client scheme

3

Outline of the system

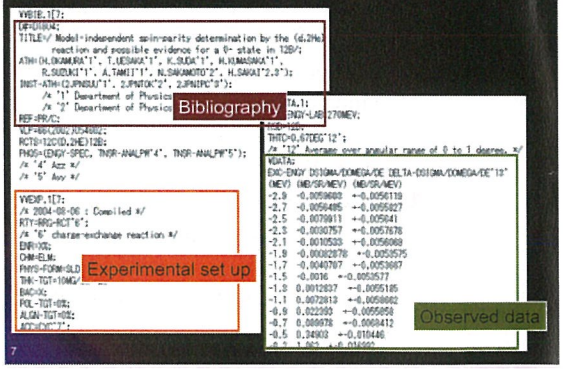
1. To provide data easily and widely
   Server and client scheme
2. To support for utilizing data
   Develop application for specific uses

Construct a PostgreSQL-database system for NRDF by using the Perl script

Data search and utilizing application driven by Perl/CGI

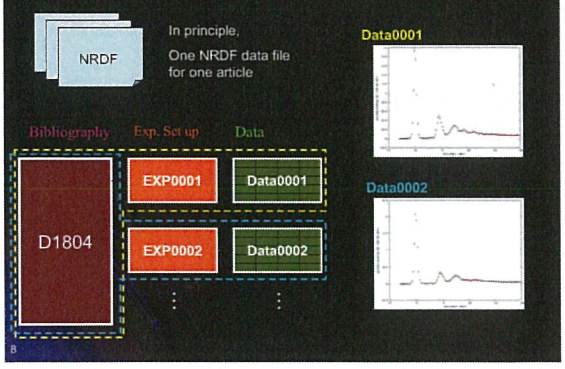
5

Data style of NRDF



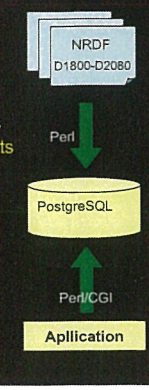
7

Data structure of NRDF



8

Why the SQL-database for NRDF is necessary?



Problem and difficulty

- We need to handle the files themselves
  ⇒ Data retrieving by "query" requests
- No-restrict format
  ⇒ Format are fixed restrict in DB

Solution for these problems

Construct the PostgreSQL-database by using the perl script

9

Table renormalization

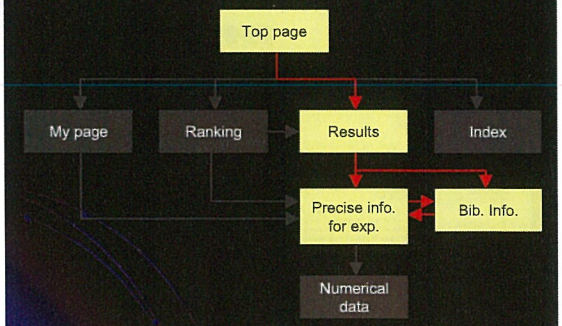
Principle key(s): D-number
And exp-number in NRDF

X(xx)	Y(yy)	E(ee)
x01	y01	e01
x02	y02	e02
x03	y03	e03
x04	y04	e04
x05	y05	e05
x06	y06	e06
x07	y07	e07
x08	y08	e08
x09	y09	e09

数値データ

11

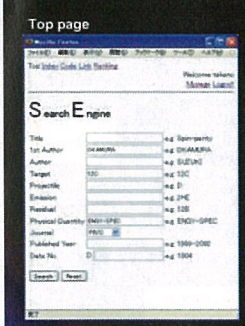
Flow-chart of the application system

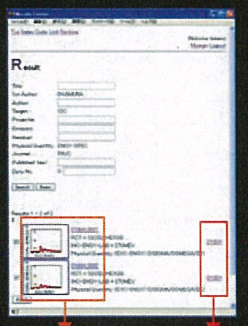


32

An ordinary key-word search form

Top page

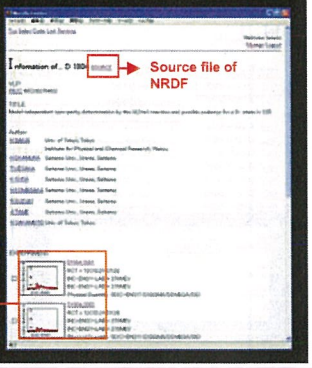




13

Bibliography

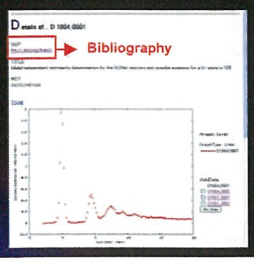
- Title of the article
- Journal
- Authors
- Link to NRDF file
- Link to numerical data

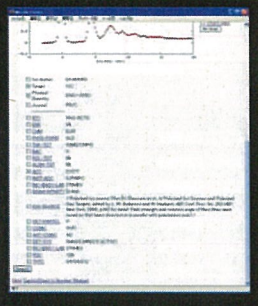


14

Precise information for the experiment

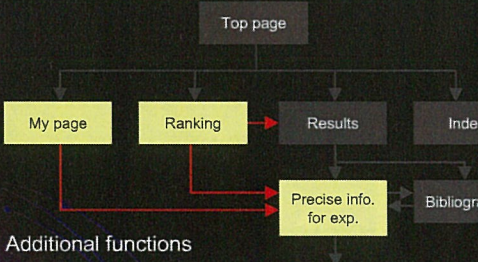
- Particles in the reaction
- Figures for experimental data
- Properties of experiemt
- Link to the bibliography





15

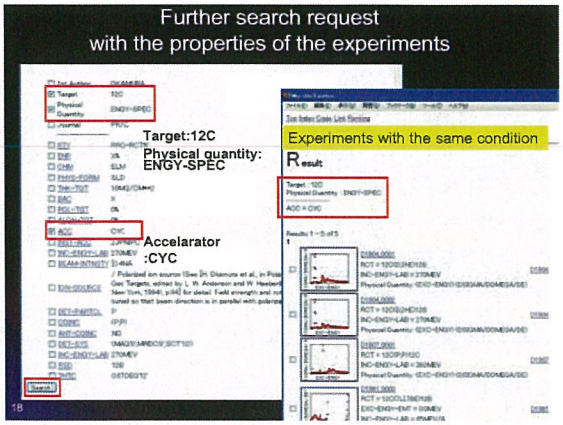
Flow-chart of the application system

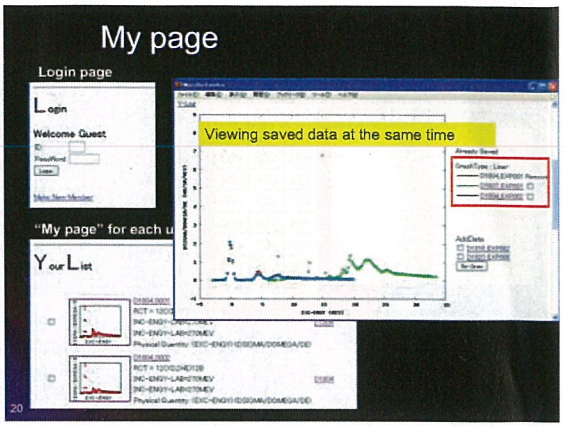


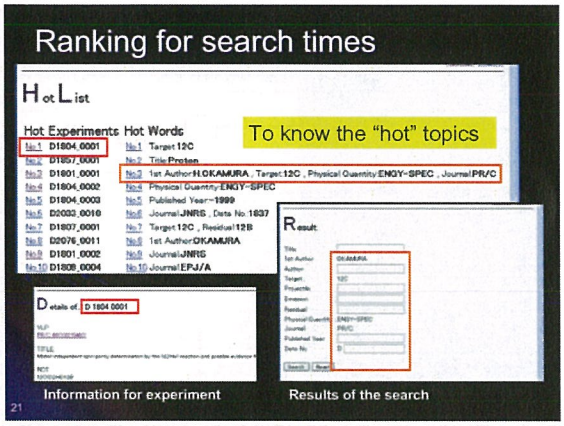
Additional functions

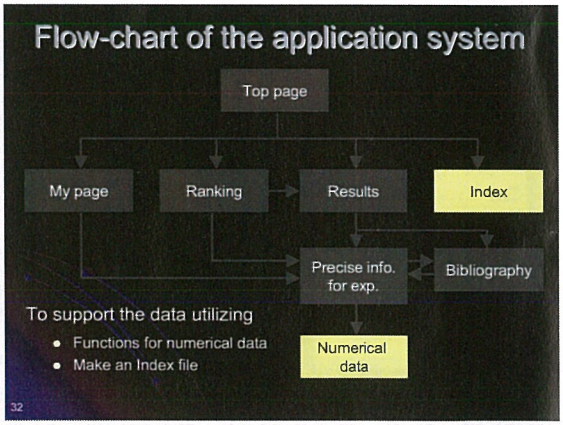
- Further search conditions
- My page
- Ranking

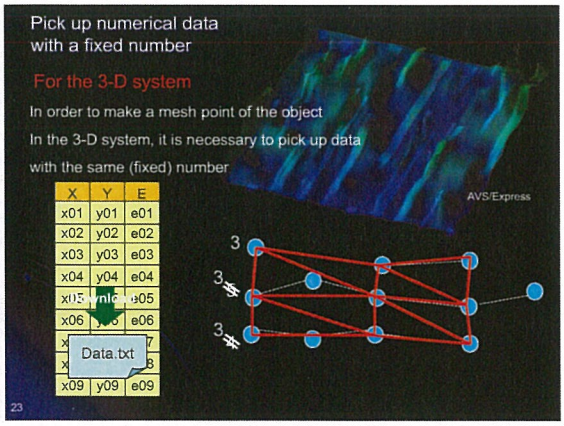
32

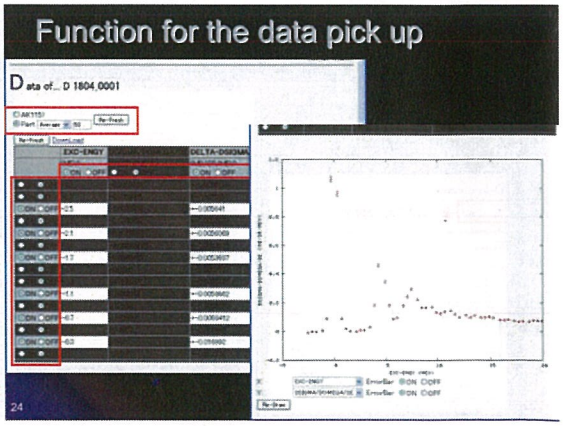

Further search request
with the properties of the experiments
Target:12C
Physical quantity:
ENGY-SPEC
Accelarator
:CYC
Experiments with the same condition
Result
18


My page
Login page
Login
Welcome Guest
Viewing saved data at the same time
"My page" for each u
Your List
20


Ranking for search times
Hot List
Hot Experiments
Hot Words
To know the "hot" topics
No.1 D1804_0001
No.1 Target:12C
No.2 Title:Proton
No.3 1st Author:H.OKAMURA , Target:12C , Physical Quantity:ENGY-SPEC , Journal:PR/C
Details of D 1804 0001
Result
Information for experiment
Results of the search
21


Flow-chart of the application system
Top page
My page
Ranking
Results
Index
Precise info.
for exp.
Bibliography
To support the data utilizing
Functions for numerical data
Make an Index file
Numerical
data
32

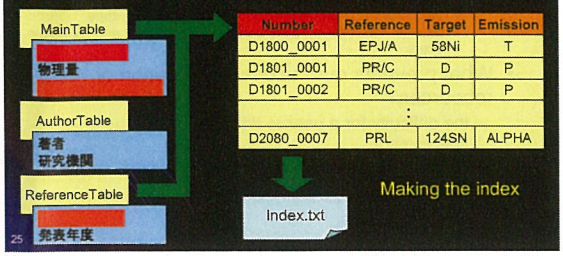

Pick up numerical data
with a fixed number
For the 3-D system
In order to make a mesh point of the object
In the 3-D system, it is necessary to pick up data
with the same (fixed) number
X Y E
x01 y01 e01
x02 y02 e02
x03 y03 e03
x04 y04 e04
x06 e06
x09 y09 e09
Data.txt
AVS/Express
23


Function for the data pick up
Data of... D 1804_0001
24

Making an index file

To utilize the data for the "Data mining"

We need to know the trend of each variable in the database as the whole



25

Summary and Discussion

- The PostgreSQL-besed server and client system for NRDF is developed
- The system is suitable for utilizing data much more than the ordinary text-based search system.
- For Hi-performance data retrieving and using useful functions, e.g. making index, one need more computer power (faster, larger, and multi-pipe)

28

資料２

Meme Media and Knowledge Federation

Yuzuru Tanaka, Jun Fujima,
and Micke Kuwahara
Meme Media Laboratory, Hokkaido University
N13 W8, Sapporo 0608628, Japan
{tanaka, fujima, mkwahara}
@meme.hokudai.ac.jp

The Web as a repository of a huge variety of knowledge resources

- The Web is becoming a huge open repository of knowledge resources that cover almost all areas
  - knowledge resources
    - What can be dealt with by computers, including data, documents, tools, and services, for the sharing and reuse by users.
- What prevents our maximum utilization of Web resources?
  - the difficulty for a user to find what he or she is interested in
  - the difficulty for a user to make mutually related Web resources work together

R&D efforts

- the difficulty for a user to what he or she is interested in
  - Big projects in the US, Europe, and Japan, such as Digital Library Initiative (DLI) Projects in US, and Information Grand Voyage Project in Japan
- the difficulty for a user to make mutually related Web resources work together
  - no major projects except those on GRID computing,
    - which basically focuses on the interoperability of distributed resources to perform routine or frequently requested jobs.

Knowledge federation as a key technology to develop

- "Federation" denotes ...
  - the process of combining multiple technology elements into a single virtual entity
- 2 origins in IT
  - a federated database architecture (Dennis Heimbigner,1985):
    - A collection of independent database systems are united into a loosely coupled federation in order to share and exchange information.
  - federation of services (Bill Joy, late 90s) :
    - A federation is a set of services that can work together to perform a task
- Knowledge federation (Yuzuru Tanaka, 2004)
  - flexible, *ad hoc*, and instantaneous selection, customization, and combination of knowledge resources by users
  - The 1st International Conference on Knowledge Federation, Oct. 20-22, 2008, Duvrovnik, Croatia

Extending the Web

Web 2.0

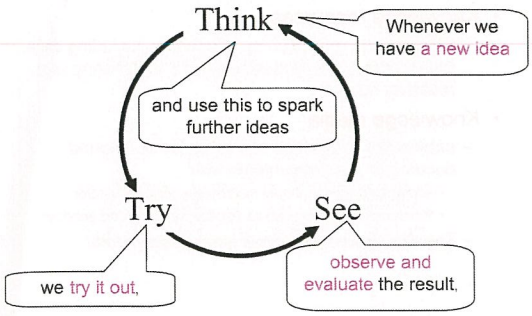
Collaborative and/or semantic reorganization of Web resources to structurally enrich relations among Web resources

Collaborative reediting of Web resources for composing new ones to structurally and functionally enrich Web resources

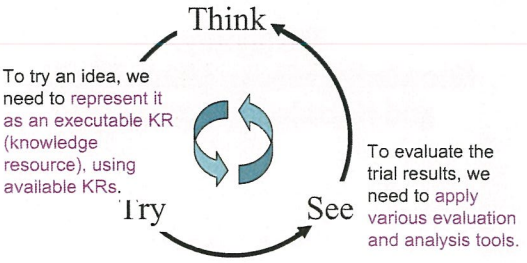
Federation is different from integration

- Federation
  - assumes an open networked environment of diverse, heterogeneous, autonomous, and distributed resources, and
  - deals with open scenarios of information processing.
- Integration
  - targets local and centralized management, and interoperation of resources in a closed environment, and
  - deals with closed scenarios of information processing.

Why ad hoc knowledge federation?



Acceleration of Creative Thinking

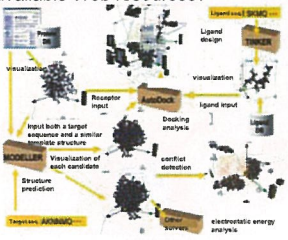


Ad hoc knowledge federation and the reuse of federation results

- Immediate trial of a new idea requires
  - rapid composition of a new tool from available resources,
  - its application to a data set extracted from available resources.
  - repetition of such processes.
- Further reuse by other people requires
  - republication of these composed tools and extracted data sets as web resources

A grand challenge of knowledge federation technologies

- Could one compose such a complex application as a drug design system just through ad hoc federation of available Web resources?



Components exist as Web resources.

Not possible now,

But yes in a near future.

Science Infrastructure

- new research methodology: 'data-based science', or 'data-centric science'
  - systematic data acquisition first,
  - then data analysis for hypothesis validation
    - genomics, proteomics, brain science, clinical trials, nuclear physics, astrophysics, material science, meteorology, and seismology
- Leading to
  - a large number of huge, independent accumulations of data, and
  - a great variety of data analysis tools.
- Requiring knowledge federation operations
  - extraction of appropriate data sets from multiple seemingly unrelated sources,
  - their compilation or customization, and
  - the application of data analysis and/or data visualization tools to these data sets.
- Such operations are inherently performed in an *ad hoc* manner, and should be executable rapidly enough to follow up on fleeting ideas.

Science infrastructure vs. conventional e-science

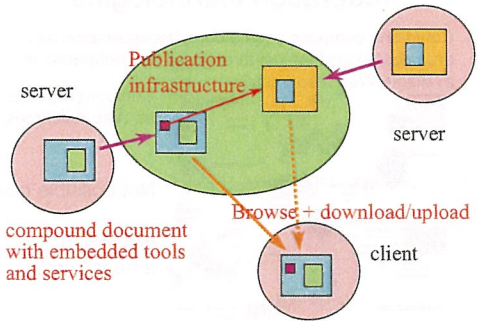
- Science infrastructure
  - requires a new generic technology for 'knowledge federation'
    - as well as the development and extension of databases, simulators, and analysis tools in each area.
    - to meet the dynamically changing demands of creative activities.
- Conventional e-science
  - For routine or frequently requested jobs
    - work-flow and/or resource-orchestration technologies
    - based on GRID computing.

Taxonomy: Knowledge Media, Meme Media, and Knowledge Federation

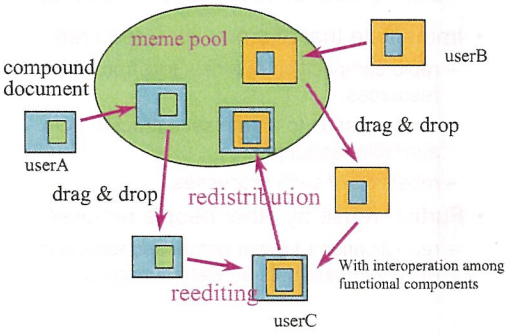
Knowledge resources, and knowledge media

- Knowledge resources
  - What can be dealt with by computers, including data, documents, tools, and services, for the sharing and reuse by users
- Knowledge media
  - externalize knowledge resources as compound documents, i.e., documents with
    - embedded nonfunctional multimedia contents and/or
    - functional contents such as application tools and services
  - The Web is an example of knowledge media.

The Web as knowledge media



Meme media



Knowledge federation to make the Web work as meme media

- Knowledge federation introduces memetic evolution to the Web
- Biological evolution is based on mechanisms
  - Replication / Recombination / Mutation / Natural selection
- Same is true with the evolution of knowledge resources (Web resources)
  - Replication / Reediting / Evaluation by a society or a community
  - Reediting
    - Easy for multimedia documents
    - Difficult for compound documents with functional components
  - Knowledge federation accelerates collaborative reediting of Web resources
- Web resource as a meme

Meme Media as Basic Enabling Technologies

Meme Media Architectures: IntelligentPad and IntelligentBox

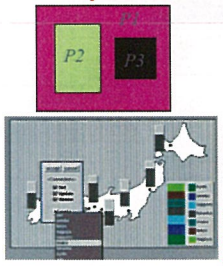
- 1987-1992:
  - Component-based media architecture IntelligentPad (1989)
    - pads as media components
- 1993-2001:
  - IntelligentPad as a Meme media architecture for users to reedit and to redistribute intellectual resources (1993)
  - 3D meme media architecture IntelligentBox (1995-)
    - boxes as media components

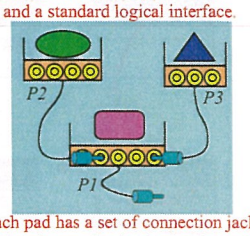


IntelligentPad as Meme Media

Pads are pasted together to define a Compound Document.

Each pad wraps an object with a standard display representation and a standard logical interface.

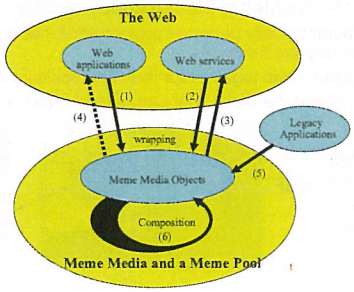




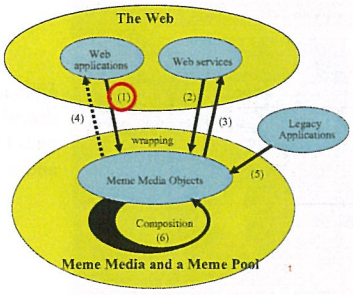
Each pad has a set of connection jacks called slots and a single pin plug to be inserted to one of its parent's slots

How to apply meme media to Web resources?

Wrapping web resources as meme media objects

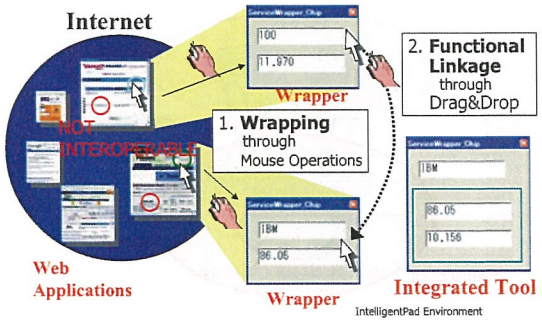


How to wrap different types of knowledge resources?

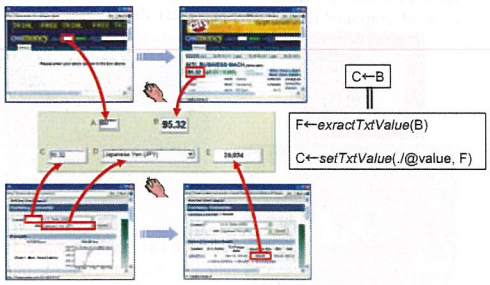


Wrapping and Linking of Web Applications

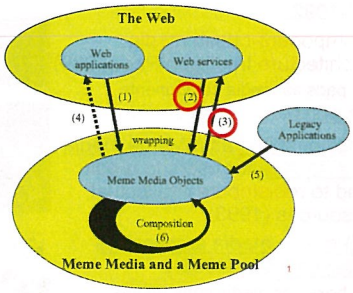
CHIP (2003) for Dynamic Linkage among Web Applications



Linkage between Web clips from different navigations (C3W, 2004)

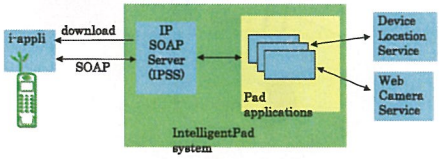


How to wrap different types of knowledge resources?

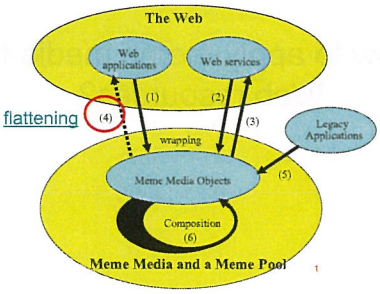


Proxy pads and server-side meme media

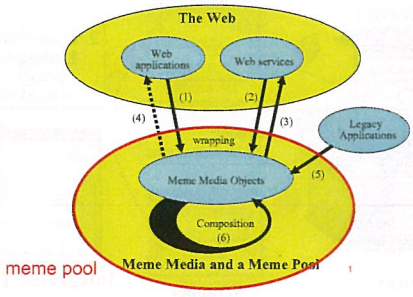
- Server-side Meme Media (2005) enabled users
  - to automatically generate proxy pads for accessing arbitrary Web services,
  - to paste them together for their composition, and
  - to convert the composite pad to a new Web service.



How to wrap different types of knowledge resources?



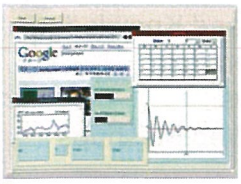
How to wrap different types of knowledge resources?



Piazza as a meme pool system (≠ the Web)

Upload and download of pads by drag-and-drop (1998)

Wiki Piazza (2004)



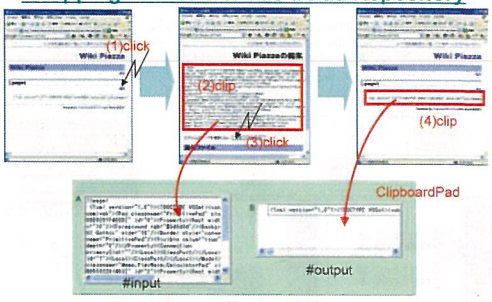
- Wiki piazza system works both
  - for resource providers to register new computing resources, and
  - for resource consumers or users to look up and to use available resources.

Repository service
to register computing resources wrapped as pads

Lookup service
also composed by Wrapping of Wiki Search and Google Search: Similarity search

Piazza based on Wiki

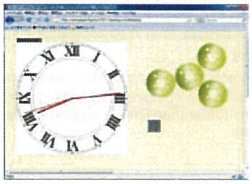
Wrapping Wiki to work as a Pad Repository



After defining slots to the ClipboardPad, we use only this pad.

How to make the Web work as a meme pool?

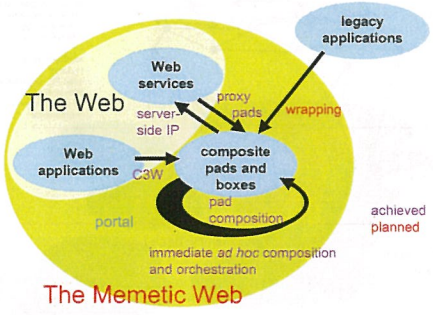
- The 2008 version of IntelligentPad: a web-top system
  - fully exploits Microsoft Silverlight technology, and runs on Internet Explorer 6 and 7, Mozilla Firefox, or Safari browser empowered by Silverlight plug-in.
  - no need to install any IntelligentPad kernel on clients



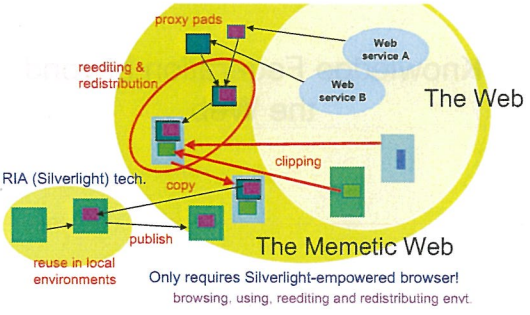
New features of the new version

- no IntelligentPad kernel running on clients
  - Users do not recognize any difference between web resources and composite pads.
  - Composite pads can be also registered in HTTP servers as extended web resources.
- canvas-free pads
  - Any graphical object including a line segment can be treated as a pad.
    - SVG (Scalable Vector Graphics) to describe graphical objects.
    - A child pad on a canvas-free pad is bound by the local coordinate system spanned by the latter pad,

The Memetic Web

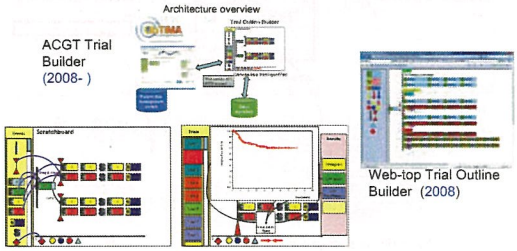


The Memetic Web



EU's FP6 Integrated Project ACGT (Advancing Clinico-Genomic Trials on Cancer)

- Trial Outline Builder: a subsystem of ObTiMa
  - Joint development with the Department for Pediatric Hematology and Oncology at the University Hospital of the Saarland, and Fraunhofer IAIS



Meme Media and Knowledge Federation vs. SOAs and Web Mashups

Related Technologies (1) Service-Oriented Architecture (SOA)

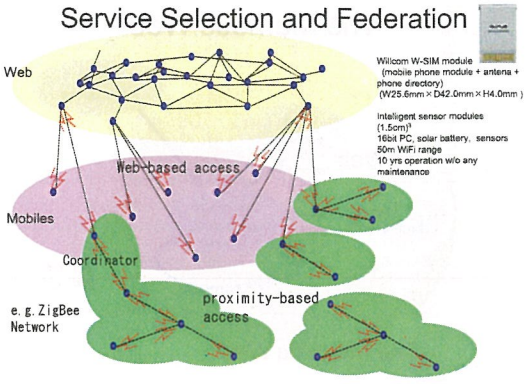
- Two similar technology trends:
  - Science / Business Cloud-based SOA
    - Provides processing power, services, middleware, developers' toolkits through the Web
  - Science / Business GRID-based SOA
    - Provides HPC power, services, and service-composition workflow definition / execution / management systems.
- Both emphasize:
  - definition, execution, and management of routine or frequently requested jobs
  - **not** user-centric, immediate, unforeseen *ad hoc* composition
- Therefore both provide:
  - tools for editing internal representations of service compositions,
  - **not** tools for directly reediting compound documents to compose new knowledge resources

Related Technologies (2) Mashup Technologies

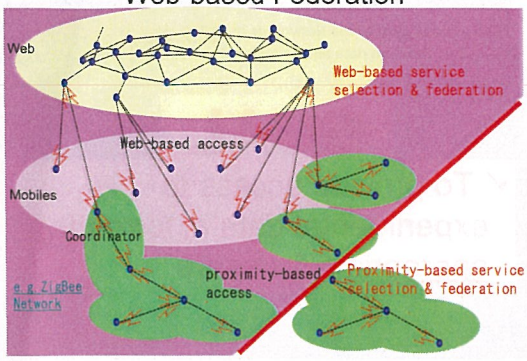
- Mashup: user-driven micro-integration of Web data
  - i.e., data merging / data feeding / data joining / data filtering / data annotating
  - Traditional mashup tools are based on workflow composition of Web services to construct a new Web application or service
- Strengths
  - They are starting to address RIA (Rich Internet Applications).
  - Some tools (e.g., Mash Maker and Lotus Mashup) allow users to clip widgets directly from Web applications.
- Weaknesses
  - Widget extraction and composition use different environments.
  - Widgets cannot be combined with local tools to compose new applications (other than limited support for *de facto* standard office tools).

Knowledge Federation beyond the Web

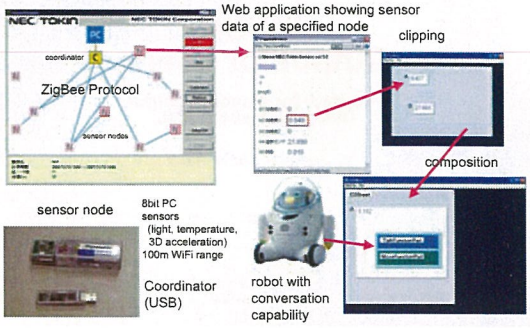
Service Selection and Federation



Web-based Federation



Knowledge Federation between a Sensor Network and a Robot



Concluding Remarks (1)

- The current Web
  - a huge repository of knowledge resources of almost all areas.
- The problem is how to make it easy
  - to select some set of resources including web resources and local ones,
  - to customize them, and
  - to make them work together.

Concluding Remarks (2)

- Knowledge federation as a key technology for people
  - to freely and immediately reedit some of the Web resources,
  - to compose a new knowledge resource for his or her own reuse, and
  - to publish it again as Web resources..
- The 2008 version of IntelligentPad extends the Web to the memetic Web.
  - Users can
    - freely publish both web resources and composite pads into the memetic Web,
    - extract resource fragments from some resources in the memetic Web,
    - make them work together to compose a composite pad, and
    - publish it into the memetic Web for its further reuse by others.

For further information on meme media architectures and applications

Tanaka, Y.

Meme Media and Meme Market Architectures: Knowledge Media for Editing, Distributing, and Managing Intellectual Resources.

IEEE Press & Wiley-Interscience, NJ (2003)

資料３

EXFOR.
Output for users.

Proposal for standard output.
Discussion about XML.

V. Zerkin, IAEA-NDS

Sapporo, Japan
March-2009

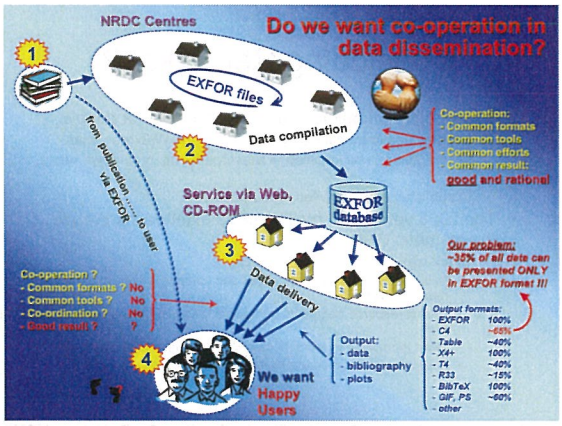
Objectives and Tasks of NRDC.

"The primary goal of the Network is the dissemination of nuclear reaction data and associated documentation to users"

GOAL

✓ To provide access to experimental data in format(s) appropriate for users

---



NRDC-2008

Accepted

WP2008-38
NDS Proposal for a Common NRDC EXFOR Web Service:

- Common output formats
- Co-operation in data dissemination systems

EXFOR: EXchange FORmat

i.e. exchange between Nuclear Data Centers

Now EXFOR - System including

- Data files in EXFOR format
- Dictionaries
- Rules and exchange procedures
- Database (s)
- Retrieval systems
- . . . . . Utilities and Applications . . . . .

60s – 80s main task: exchange data.

Now EXFOR: ~18500 Entries, +500 Entries / year

1980: 3000+500 (+16.7%)
1990: 8000+500 (+6%)
2000: 12000+500 (+4%)
2008: 18000+500 (+2.8%)

Database extension: stable

Access to data: growing

1997. Web access
1998. CD-ROM
2000. Migration to relational databases
2004. Advanced plotting
2006. New output: X4+, R33, X4±
2007. SG-30; full database in C4 format
2008. New output: X4±, XML
2009. Exchange: well established

2009. New tasks in EXFOR:

- X4-evaluated
- new EXFOR checking-code
- Web-service on PC
- . . . . . .
- develop output formats:
  - Standard output for users
  - XML (?)
  - New computational format

Now EXFOR is de-facto main output format. But it is not convenient for most of users!!!

- Who knows MS-Word file format (DOC)?
- You normally look at Web pages using Web browser, although you can also see html-source of the page…
- The same with EXFOR: it is convenient for experienced compilers to exchange data between themselves, but why data should be given in the same form for users?...

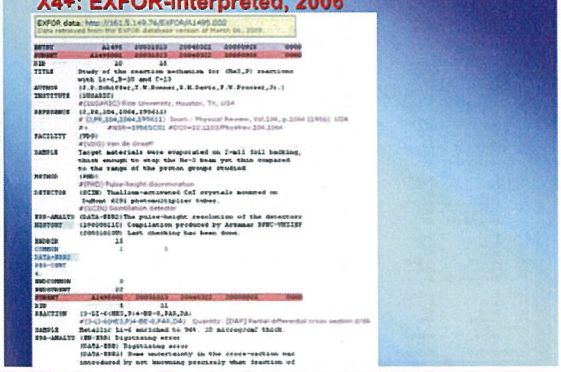
What do we offer to users beyond EXFOR format?

Output of NDS EXFOR System:

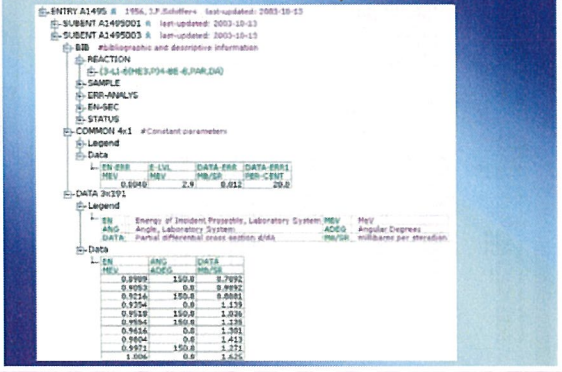
- EXFOR	100%
- C4	~65%
- Table	~40%
- X4+	100%
- T4	~40%
- R33	~15%
- BibTeX	100%
- GIF, PS	~60%
- other	

NDS Web EXFOR: Output

X4+: EXFOR-interpreted, 2006



X4±: EXFOR-interactive Tree, 2008



EXFOR STANDARD OUTPUT

/under development/

Problems with EXFOR (as output format for users):

- Difficult to interpret - requires additional knowledge (structure and info)
- Data are spread through the file (Common-1, Common-N, Data-N) and not sorted (order of columns is not fixed)
- As result: difficult to write software reading EXFOR
- Existing output formats (C4, Plots, Tab,…) do not cover all EXFOR data

To write program reading EXFOR you should know:

- Structure of the file
- Dictionaries (structure and relations)
- Pointers
- Common sections (Subent-N and Subent-1)

EXFOR STANDARD OUTPUT

Goal:

To define comprehensive output format of EXFOR system: equivalent to EXFOR, but much easier to read and interpret

Mini-goal:

To deliver data to users from EXFOR in format simple for software development and easier for understanding

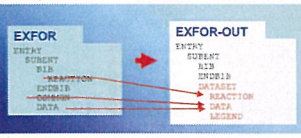
Audience:

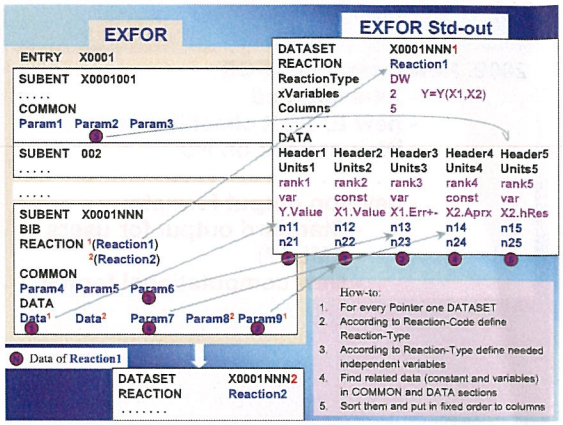
- professional evaluators, developers of model codes,… (programming)
- data centres developing retrieval/processing systems (programmers)
- regular users (?)
- all EXFOR users (?)

EXFOR STANDARD OUTPUT

Main Features:

1. Formatted (fixed columns) text
2. Keywords, codes, free-text from EXFOR, but
   + interpreting information from dictionaries
3. No pointers
4. New level: Dataset – includes all information related to one reaction: reaction-code, interpretation, data-section and legend
5. Data-section contains data from EXFOR DATA and COMMON sections
6. Data columns are sorted (fixed order)
7. Legend contains codes and interpretation of data (includes, e.g. basic-units and conversion factors)





Example of DATASET

```
#DATASET        13970003
#NOW            2009/02/24:14:25:08
#SUBENT         13970003   20050225
#ENTRY          13970      20050225
#AUTHOR1        S.A.Long+
#YEAR           1998
#X4REF1         J,PR/C,57,3191,1998
#REFERENCE1     Jour: Physical Review, Part C, Nuclear Physics, Vol.57, p.3191 (1998)
#REACTION       73-TA-181(N,P)72-HF-181,,DA/DE
#D4REAC         RDF
#C4Reaction     (N,P),DA/DE
#ReactionType   DAP
#Quantity       Double diff.cross section d2/dA/dE
#xVariables     3
#4              Y = Y(X1,X2,X3)
#COLUMNS        6
#HEADER         7          6          12
!DATA       -DATA-ERR   EN      EN-RSL     E-EXC      ANG
!MB/SR/MEV  MB/SR/MEV   MEV     MEV        MEV        ADEG
!Y.Value    Y.Err-      X1.Value X1.Resl+- X2.Value   X3.Value
!0.1        0.901       1.1     1.922      2.1        3.1
!variable   variable    const   const      variable   variable
!1.0E-9     1.0E-9      1000000.0 1000000.0 1000000.0  1.0
!B/SR/EV    B/SR/EV     EV      EV         EV         ADEG
#DATA       162         6          12
 0.03934    0.08472     300.    3.         0.2394     1.2
 0.02584    0.1355      300.    3.         1.703      1.2
 0.06289    0.08472     300.    3.         3.043      1.2
 0.236      0.1016      300.    3.         4.624      1.2
 0.5784     0.08472     300.    3.         6.077      1.2
 0.9543     0.1016      300.    3.         7.408      1.2
 1.449      0.08472     300.    3.         9.101      1.2
 1.69       0.1016      300.    3.         10.67      1.2
 2.033      0.1355      300.    3.         12.13      1.2
 1.968      0.1186      300.    3.         13.59      1.2
 2.107      0.08472     300.    3.         15.05      1.2
 2.331      0.1694      300.    3.         16.39      1.2
 . . . . . .
 . . . . . .
```

Example of DATASET (continued)

```
 2.393      0.2622      300.    3.         33.6       13.7
 2.508      0.295       300.    3.         35.1       13.7
#ENDDATA        162         6
#Legend          6          12
#
#COL-1      : Data       : variable
#+  Rank    : 0.1        : Y.Value
#+  Type    : 21         : 0.1
#+  Header  : DATA       : [Data: data]
#+  Units   : MB/SR/MEV  : [millibarns per steradian per MeV]
#+  BasicUnits: B/SR/EV  : Conv.Factor=1.e-09
#
#COL-2      : Error      : variable
#+  Rank    : 0.901      : Y.Err-
#+  Type    : 21         : 0.901
#+  Header  : -DATA-ERR  : [Uncertainty: -error]
#+  Units   : MB/SR/MEV  : [millibarns per steradian per MeV]
#+  BasicUnits: B/SR/EV  : Conv.Factor=1.e-09
#
#COL-3      : Data       : constant
#+  Rank    : 1.1        : X1.Value
#+  Type    : 41         : 2.1
#+  Header  : EN         : [Incident energy: energy]
#+  Units   : MEV        : [MeV]
#+  BasicUnits: EV       : Conv.Factor=1.e+06
#
#COL-4      : Error      : constant
#+  Rank    : 1.922      : X1.Resl+-
#+  Type    : 41         : 2.922
#+  Header  : EN-RSL     : [Uncertainty: +-resolution]
#+  Units   : MEV        : [MeV]
#+  BasicUnits: EV       : Conv.Factor=1.e+06
#
#COL-5      : Data       : variable
#+  Rank    : 2.1        : X2.Value
#+  Type    : 53         : 3.1
#+  Header  : E-EXC      : [Secondary energy: excitation energy]
#+  Units   : MEV        : [MeV]
#+  BasicUnits: EV       : Conv.Factor=1.e+06
#
#COL-6      : Data       : variable
#+  Rank    : 3.1        : X3.Value
#+  Type    : 61         : 4.1
#+  Header  : ANG        : [Angle: angle] Angle of outgoing
particle
#+  Units   : ADEG       : [Angular Degrees]
#
#ENDDATASET     13970003
```

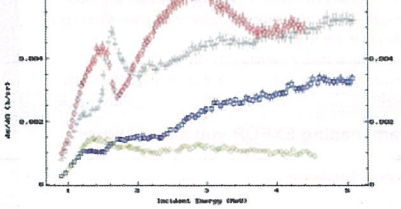
Example-2 of DATASET

```
REQUEST         62001   20061104        3     150853          0  0  0
#DATASET        103200023
#NOW            2009/02/24:19:23:49
#SUBENT         10320002   19991005
#ENTRY          10320      20020823
#AUTHOR1        G.R.Norman+
#YEAR           1972
#X4REF1         J,CJP,50,2385,197210
#REFERENCE1     Jour: Canadian Journal of Physics, Vol.50, p.2385 (1972)
#REACTION       13-AL-27(N,TOT),,SIG,,RES
#D4REAC         RDF
#C4Reaction     (N,TOT),SIG,,RES
#ReactionType   CSN
#Quantity       Cross section at resonance
#xVariables     1
#+              Y = Y(X1)
#COLUMNS        2
#HEADER         7          2          12
!DATA       EN-RES
!B          MEV
!Y.Value    X1.Value
!0.1        1.1
!variable   variable
!1.0        1000000.0
!B          EV
#DATA       20         2          12
 2.4        0.79
 1.2        0.85
 . . . . . .
 0.55       3.62
#ENDDATA        20         2
#Legend          2          12
#COL-1      : Data       : variable
#+  Rank    : 0.1        : Y.Value
#+  Type    : 21         : 0.1
#+  Header  : DATA       : [Data: data]
#+  Units   : B          : [Barns]
#COL-2      : Data       : variable
#+  Rank    : 1.1        : X1.Value
#+  Type    : 32         : 1.1
#+  Header  : EN-RES     : [Resonance parameter: energy] Resonance energy
#+  Units   : MEV        : [MeV]
#+  BasicUnits: EV       : Conv.Factor=1.e+06
#ENDDATASET     103200023
```

Contra-example, when separation to DATASET's is not reasonable

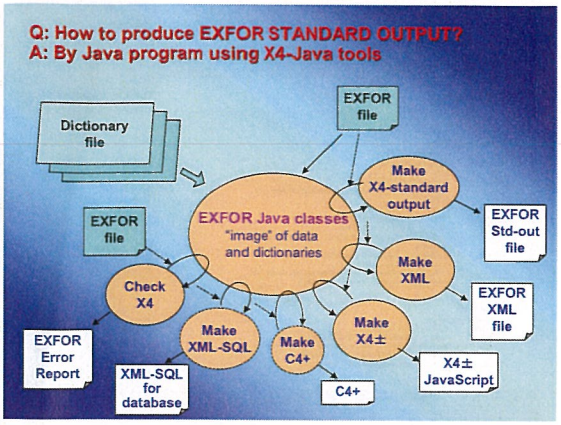
A possible solution is to have one dataset, but with several Data headers, e.g.
DATA-EN
DATA-WID
DATA-SIGR

Example-3 of DATASET



Another contra-example: several Subentries can be joined…

We can have one Dataset, two or four



EXFOR-XML

What is wrong with EXFOR format?

- Difficult to extend (e.g. +DOI, NSR to REFERENCE) →software-problem
- Difficult to interpret Codes (partially solved in X4+ and EXFOR-Editor)
- Not always clear how to interpret Data (to be solved in X4Std-out)
- Difficult to interpret Common parameters

Why XML?

- Easy to extend (e.g. DOI, NSR can appear as Attributes)
- Easy to write new programs (Java)
- Nowadays "Standard" for exchange information between Applications
- Standard software can be used

NOT XML?

- A lot of development work at the beginning (need experience, time for studies)
- Legacy codes working with EXFOR - rewrite?
- Problems for compilers and Fortran-users
- A lot of unnecessary text in the files, need viewer and editor

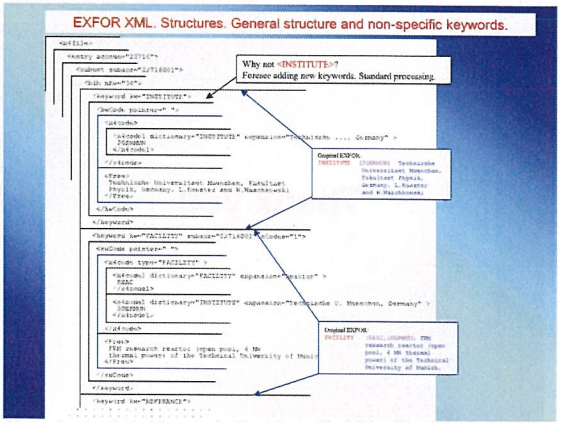
Main Question:

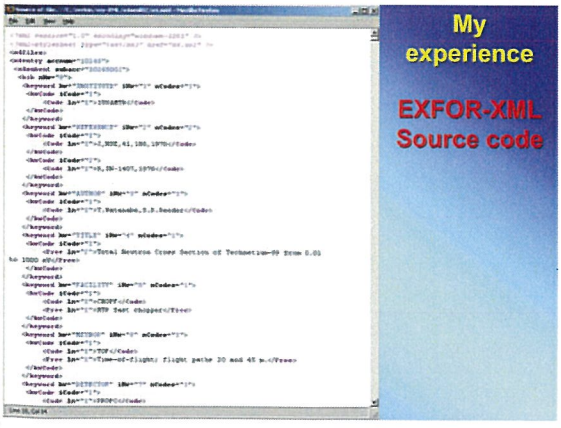
What is the purpose of EXFOR-XML?

Output format (now)? Or Exchange (future)?

What we are going to achieve?

XML: Plus	XML: Minus	Alternative
Self explanatory	Need Viewer	X4+, X4±, X4Std-
May allow to use standard validation	Create Schema (experience…)	out Java-X4Check
Allow to use standard Editors (?)	Existing Editors are good enough?	(develop…) EXFOR-Editor
Easy to build modern programs (Java)	What about Fortran programs?	(Sarov) X4Std-out: easy to build programs in any language





My experience

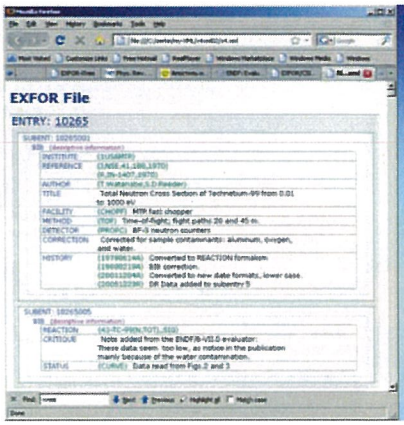
EXFOR-XML Source code

```
<?xml version = "1.0" ?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/exfiles">
<xsl:comment >This is a comment</xsl:comment>
<HTML>
<head>
<link rel="stylesheet" type="text/css" href="x4.css"/>
</head>
<BODY>
<H1>EXFOR File</H1>
<table border="0" width="100%"><tr><td>
<xsl:for-each select="x4entry">
  <div class="x4entry">
      ENTRY: <b><xsl:value-of select="@accnum"/></b>
  <xsl:for-each select="x4subent">
    <div class="x4sub">
        SUBENT: <xsl:value-of select="@subacc"/>
    <br/>
    <xsl:if test="bib !='' ">
    <div class="x4bib"><font color="blue">BIB</font> <span
class="x4hlp">(descriptive information)</span><br/>
      <div>
      <table border="0" width="100%">
          <font color="blue">
          <xsl:for-each select="bib/keyword">
        <td>
        <td valign="top" class="x4biblnv">
            <xsl:value-of select="$kw"/>
        </td>
        <td class="x4biblwcode">
            <xsl:for-each select="kwCode">
          <div class="x4code">
              <xsl:if test="Pointer !='' ">
                  <span class="x4pointer"><xsl:value-of select="Pointer"/></span>
                  <xsl:text > </xsl:text>
              </xsl:if>
              <xsl:if test="Code !='' ">
                  <span class="x4code1"><xsl:value-of select="Code"/></span>
                  <xsl:text > </xsl:text>
              </xsl:if>
              <xsl:if test="Free !='' ">
                  <span class="x4free"><xsl:value-of select="Free"/></span>
              </xsl:if>
          </div>
            </xsl:for-each>
        </td>
        </tr>
            </xsl:for-each>
            </font>
        <span class="end1">END BIB</span>
        </table>
        </div>
      </div>
      </xsl:if>
  </xsl:for-each>
  </div>
</xsl:for-each>
</td></tr></table>
</BODY>
</HTML>
</xsl:template>
</xsl:stylesheet>
```

XSL: XML → HTML

```
div {  padding-left: 10px;  margin-left:  5px;  padding-right: 10px;}
div.x6ib1 {  border:1px dashed #aaa;  padding-bottom: 10px;}
div.x6mtmy {
  border-bottom: 40px solid #fff;  padding:0;  margin:0;
  font-size:14pt;  background-color:#eee;  page-break-after: always;}
div.x6ode {margin:0;  padding:0;}
.x6ndel {
  color:green;  padding:1px;}
.x6fvel {
  color: black;  padding:1px;  padding-left:2px;  margin-left:4px;
  white-space: pre-wrap;}
div.x6sub {
  padding-bottom: 10px;  margin: 10px;  border:1px dotted green;
  font-size:9pt;  line-height:12pt;  background-color:#fff;
}
.endl {  display: none;}
x6pointer {
  background-color:  red;
  color: white;
  padding-left:2px;
  padding-right:2px;}
body {  font-family: verdana,geneva,helvetica,sans-serif;  font-size:9pt;
  color: blue;
}
.x6hlp {
  font-family: verdana,geneva,helvetica,sans-serif;
  font-size:8pt;
  color:#c0c;
  padding-left:10px;
}
table, td {
  font-size:9pt;
  margin:0;
  padding:0;
  border-collapse: collapse;
}
td {
  border:1px solid #aaa;
  border-right: none;
  border-left: none;
  padding-left:3px;
  padding-right:3px;
  border-collapse: collapse;
}
td.x6hlhkv {
  border-left:1px solid #aaa;
  width:70pt;
}
td.x6hlhkvcode {
border-right:1px solid #aaa;
  padding-left:3px;
  padding-right:3px;
  width:250pt;
}
```

CSS: Stiles



EXFOR-XML View in Web-Browser

Thank you.

資料4

Application of XML for future EXFOR software

M. Chiba

Nuclear Data Exchange Format of Next generation

- I suppose: Nuclear Data Exchange Format of Next Generation should be defined based on XML
- Current EXFOR or NRDF Format is specific to itself
- If the Exchange Formats be redefined based on XML, we might utilize various XML supports available in developing applications

XML

- XML document: text content marked up with text tags
- Well-formed XML document: XML parsers can read it and understand it
- Tag Syntax:
  - start tag: <name of element>
  - End tag: </name of element>
- XML tree
- Attribute

DTDs and schema

- XML documents can be validated by XML parsers with DTDs or schema
- XML documents containing Well-formedness mistakes can be checked before publishing it

Validating parser

- Validating parser compares a document to its DTD and lists any places where the document differs from the constraints specified in the DTD
- Validating a Document:
  - Use the parser's API to validate documents
  - Use one of the online validators
  - Use a local program to validate the document

Namespaces

- Purposes
  - To distinguish between elements and attributes from different vocabularies with different meanings that happen to share the same name
  - To group all related elements and attributes from a single XML application together so that software can easily recognize them

Internationalization

- Character Set
- Encoding
- Text declation

XML as Document Format

- XML: first and foremost a document format
  - Human beings would read
  - A syntax for computer data in applications
  - Rigid structures: tree
  - Narrative Document Structures

XML on the Web

- XSLT stylesheet

XSL Transformations(XSLT)

- xsl:stylesheet and xsl:transform

XPath

- XPath: a non-XML language for identifying particular parts of XML documents
  - XSLT uses Xpath expressions to match and select particular elements in the input document for copying into the output document or further processing
  - Xpointer uses XPath expressions to identify the particular point in or part of an XML document to which an XLink links
  - W3CXML Schema Language uses XPath expressions to define uniqueness and identify constraints
  - XForms relies on XPath

XLinks

- Xlink: an attribute-based syntax for attaching links to XML documents

XPointers

- XPointer: a non-XML syntax for identifying locations inside XML documents

XInclude

- Xinclude: a new technology for combining multiple well formed and optionally valid documents and fragments thereof into a single document

Transformation from a Semantic markup into a presentational markup

- Cascading Style Sheets(CSS)
- XSL Formatting Objects(XSL-FO)

XSL Formatting Objects

- XSL-FO:

XML as a Data Format

- Inventors intention: XML as a format for narrative documents to be read by people
- Most common applications of XML today: storage and transmission of information
- The structures appropriate for applications
  - Rigid
  - Strongly typed element
  - Elements tend to look more likely database records
  - Make data portable
  - Proliferation of free XML tools

Strengths of using XML

- Strengths of using XML as a software data format:
  - Simple syntax: easy to generate and parse
  - Support for nesting: Nested elements allow programs to represent complex structures easily
  - Easy to debug: human-readable data format is easy to explore and create with a basic text editor
  - Language- and platform-independent: XML and Unicode guarantee that your data files will be portable across virtually every popular computer architecture and language combination in use today

XML Schemas

- Many applications need a powerful and expressive validation method:
  - W3C developed the XML Schema Recommendation
- Schemas can describe complex restrictions on elements and attribute
  - An XML Schema is an XML document containing a formal description of what comprises a valid XML document

Schemas Versus DTD

- DTD provides the capability to do basic validation of the following items in XML documents:
  - Element nesting
  - Element occurrence constraints
  - Permitted attributes
  - Attribute types and default values
- Not provide fine control over the format and data types of element and attribute values

Schemas Versus DTD

- W3C XML Schema standard includes the following features:
  - Simple and complex data types
  - Type derivation and inheritance
  - Element occurrence constraints
  - Namespace-aware element and attribute declarations

Programming Models

- XML support is available for virtually every programming platform in use today
  - Text-based XML processing
  - Event-driven XML processing
  - Tree-based XML processing
  - Pull-based XML processing

Document Object Model(DOM)

- DOM: an API for accessing and manipulating XML documents as tree structures

Simple API for XML(SAX)

- SAX: event-based API for reading XML documents