X5 - Enriched EXFOR in JSON

by Viktor Zerkin
~ independent software developer ~

EXFOR Workshop: Compilation of Experimental Nuclear Reaction Data
3 - 6 December 2024, IAEA Headquarters, Vienna, Austria

Part I.

Introduction. Purpose. Concept.

Before any development

Steps to go.

1. Understand/formulate problem and possibly task(s)
2. Three questions to answer:
   a) What already exists and can be used?
   b) What is wrong now?
   c) What do we want to achieve? (define ideal goal)
3. Preparations:
   a) discuss/define main ideas, concept and possible technologies
   b) study and test technologies, select technology
4. Plan: split problem to tasks/sub-tasks, define dates
5. Implementation: development loop

Problem and tasks

NRDC: exchange experimental data. Forms and methods of data distribution are not regulated by NRDC (no wide/deep cooperation).

1. Task: to provide EXFOR data in computational form to make available entire library in uniformed way
   - existing systems: XC4, C5, relational-DB: X4App/X4Lite/X4Pro/JANIS

2. Task: to make accessible full power of EXFOR hidden in meta data via providing information from EXFOR Dictionaries allowing various computations
   - existing systems: EXFOR + Dictionaries + Manual (needs a lot of expertise and programming)

3. Task: to provide related to EXFOR nuclear data: old monitor data and new standards, RIPL and decay data from ENSDF allowing data modernization and various data transformations
   - existing systems: EXFOR + Manual + RIPL + ENSDF + Literature (needs a lot of expertise and programming)

4. Task: to provide all above in a form convenient for programming in modern languages for the new generation new technologies (e.g. ML)
   - existing systems: partially solved in X4Pro and XML output from Web/X4Lite

What already exists and can be used? What is wrong now?

1. NDS Web database retrieval system providing data search, retrievals, plotting, comparison with evaluated data, data presentation in various forms, methods and formats: X4+, X4±, C4, C5, C5M, two JSON's, two XML's, StdOut, CompOut
2. Web data archives with full EXFOR in XC4, C5, EXFOR, Dictionaries: download from dedicated servers and GitHub
3. Standalone retrieval systems including (relational) databases: "EXFOR for Applications", X4Lite, X4Pro (NDS), JANIS (NEA)

1. Data provided by pieces in systems and data listed above, partially covering many tasks/needs but not consolidated: for example, (a) XML presents all keywords and information from Dictionaries, but not from external monitor files, RIPL and relevant ENSDF decay data, (b) C5 presents renormalized computational uniformed data, but no information from all keywords/codes, (c) EXFOR-Relational DB/X4Apps/X4Lite contain all keywords/codes in Tables, but also no external monitors and standards, and ENSDF decay data
2. A lot of needed data and information already generated and provided, and simply need to be consolidated in proper way in most popular form nowadays – JSON for building systems on new technologies suitable and familiar to new generation of users and new tasks (Python, R, NoSQL, ML).

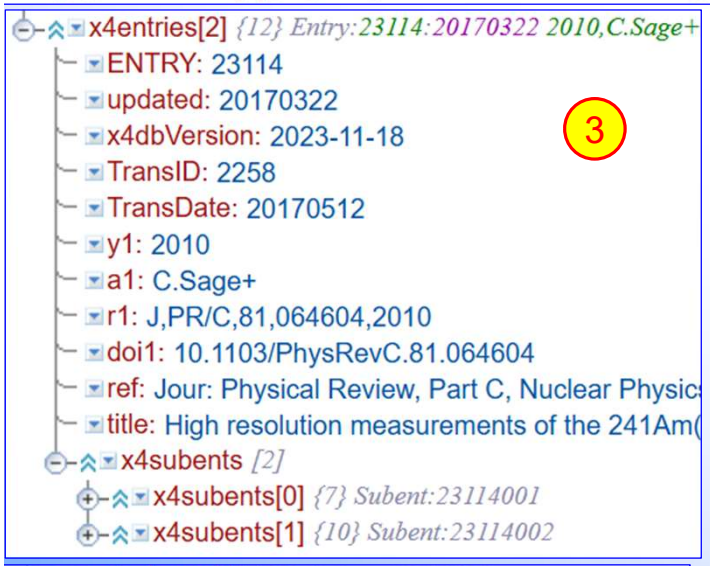
What is X5json design?

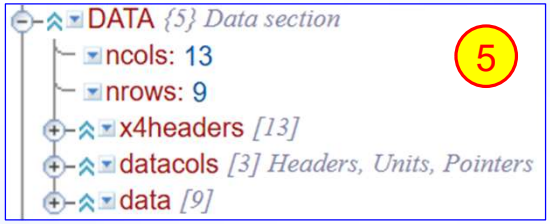
1. X5json file includes
   A. Part-I. Original EXFOR: ENTRY→SUBENT→(BIB/KW/Codes),COMMON,DATA
      1) structured meta-data (keywords/codes/free text) with dictionary-information
         + REACTION: split to Strings and further to SF1..SF9 with dictionary-info
         + REFERENCE: interpreted parameters (volume, report number, page, etc.)
         + COVARIANCE: data extracted from free-text to arrays
      2) original EXFOR data from COMMON and DATA sections: 2-D array
         + Header and Units given with interpretation and classification (form dictionaries)
   B. Part-II. Computational data: SUBENT → Datasets:<Subent+Pointer>
      1) computational EXFOR data values: 1-D arrays in (a) original and (b) basic units
         + only data relevant to Dataset (Pointer=‘ ‘ or Pointer=Dataset.Pointer)
         + sorted and classified according to dictionaries
      2) computational data ~C5: {y, Δy, Δy_{stat}, Δy_{sys}, x1, Δx1, x2, Δx2, etc.}
      3) MF, MT for data compatible with ENDF evaluated data
      4) data for renormalization: monitor cross sections and standards
      5) decay data for renormalization from ENSDF
2. X5json distributed within X4Pro, generated online on ENFOR Web retrieval system
3. Full EXFOR translated to X5json is available on GitHub by Entries; it includes also Python scripts creating data indices, import data to “pandas”, and producing plots with and without renormalization (compared with data from full EXFOR in C5)

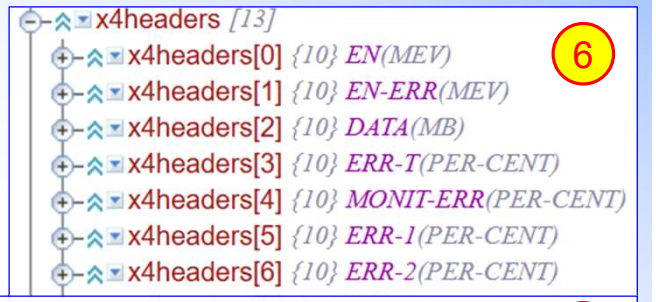
Part II.

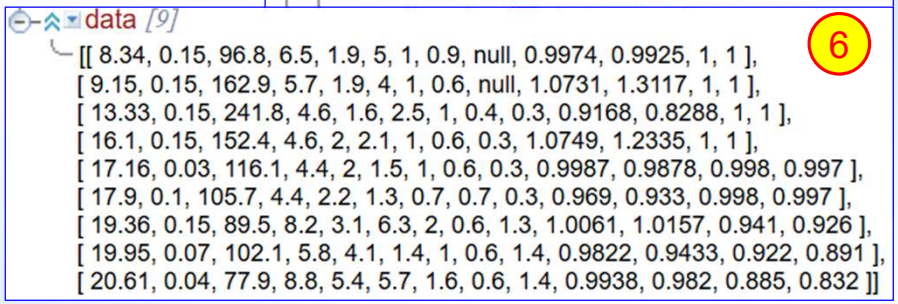
X5json. Structure. Data presentation.

X5json structure





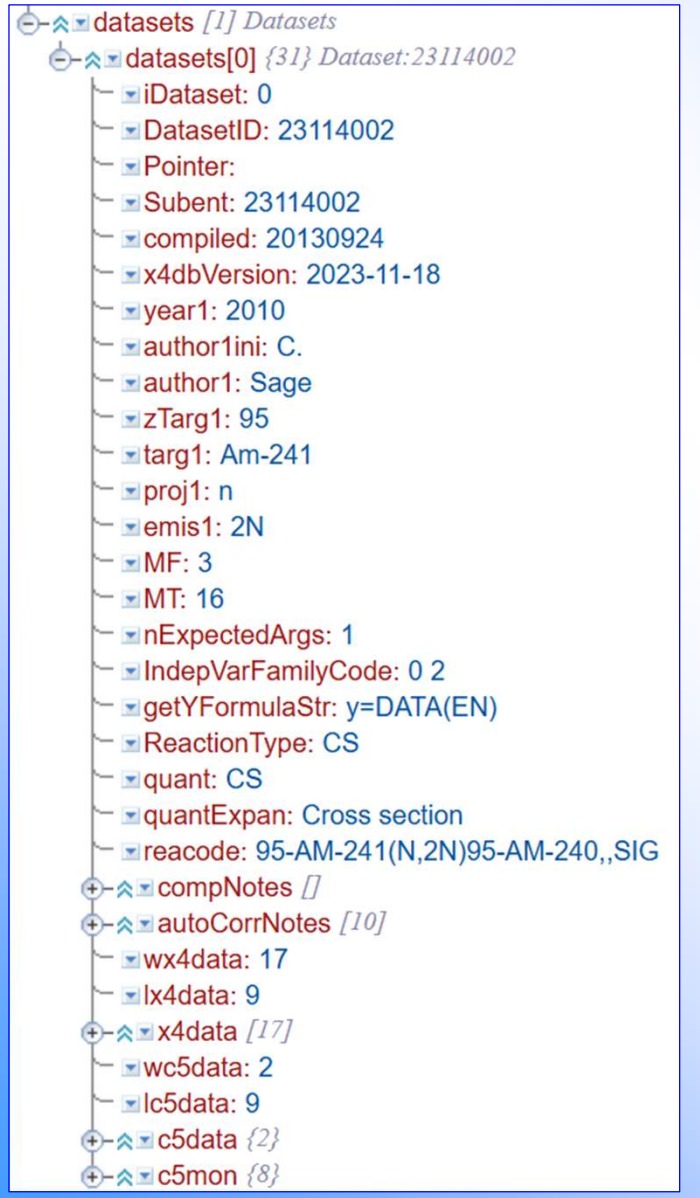


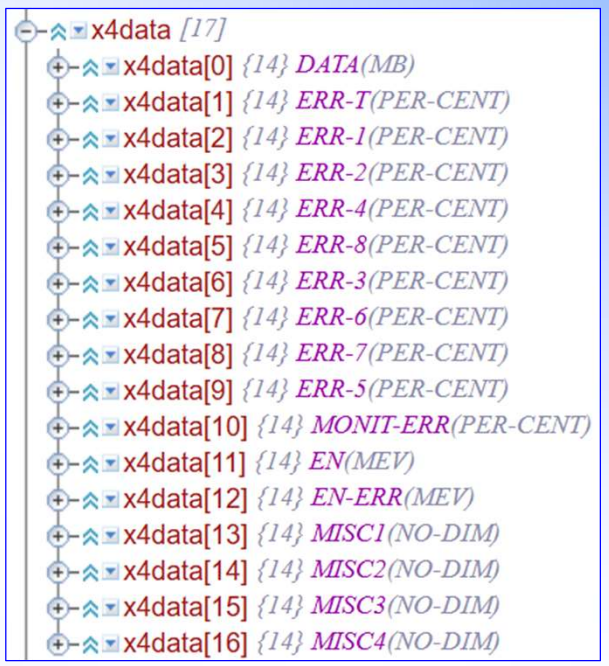


X5json hierarchy reproduces structure of an EXFOR file with extensions for ease processing.
Part-1: exactly EXFOR + information from Dictionaries
Part-2: Datasets for computational purposes

Part-1. REFENCE, REACTION, COVARIANCE

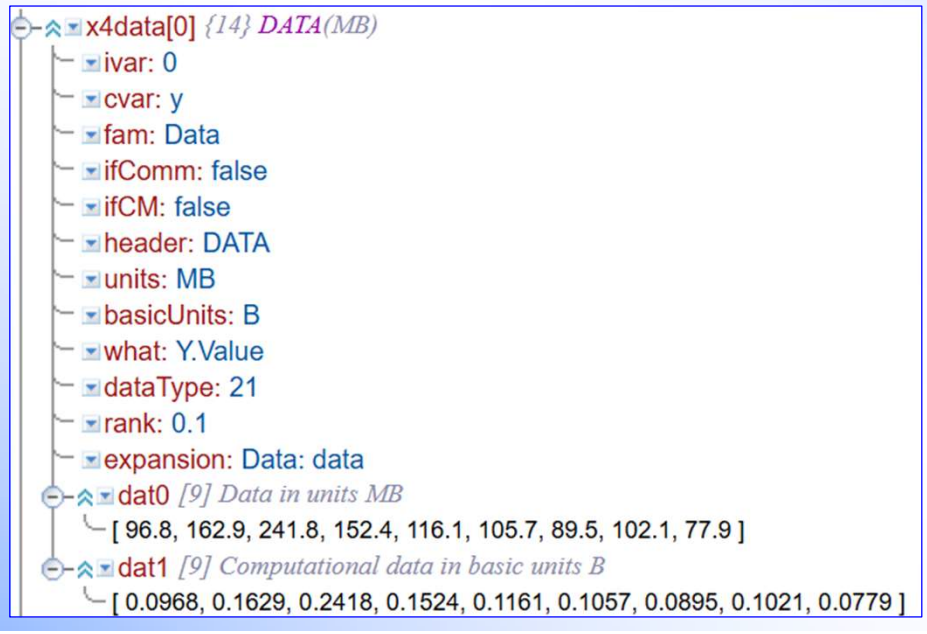
Part-2. Dataset: <Subent, Pointer>



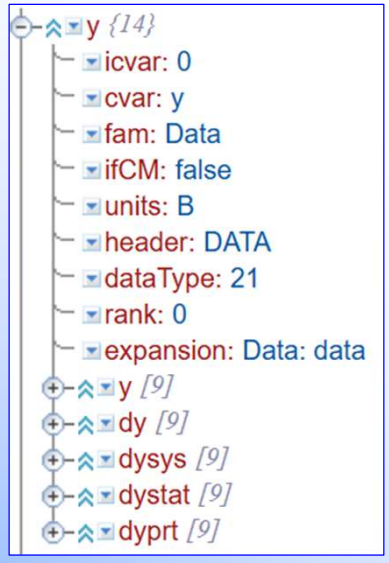


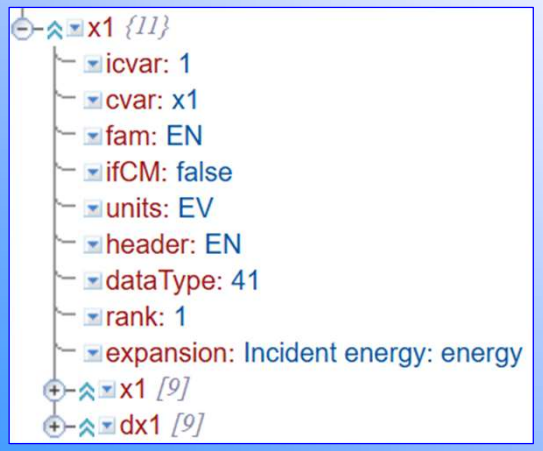
"dataset[]": Dataset information
+ Bib-info
+ info from Dictionaries
+ MF.MT

"x4data": data from DATA and COMMON relevant to the Dataset by columns.
Data columns are sorted.
Data are presented in 1-D array:
"dat0" – in original units,
"dat1" – in basic units

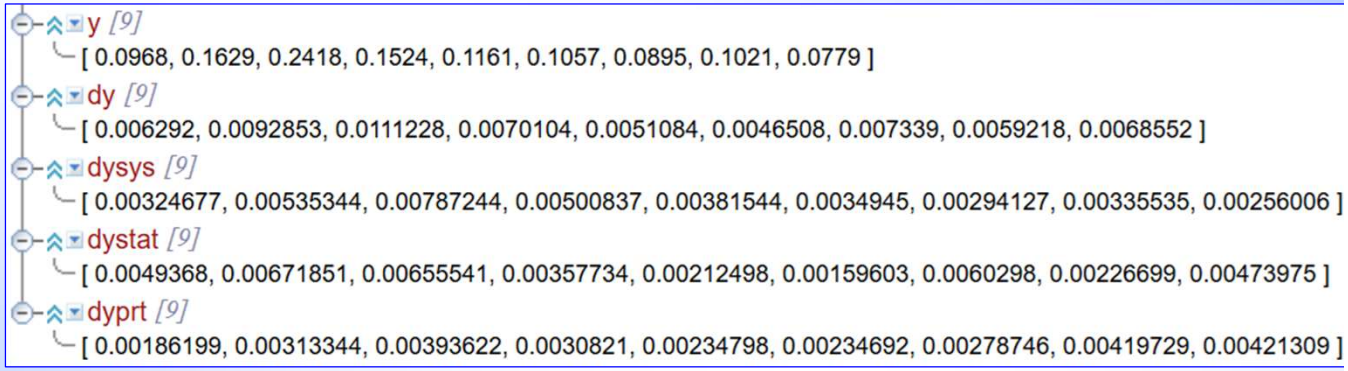


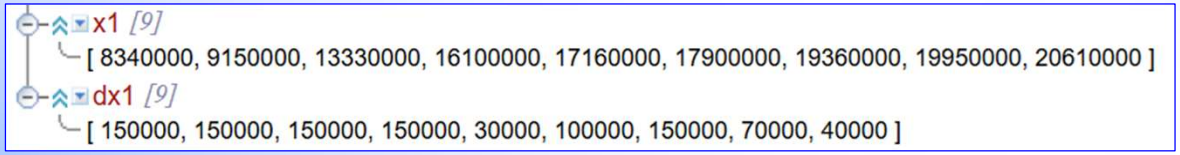
Part-2. Dataset: c5data





"c5data": all values given in basic units (like in C5)
Number of independent variables (x1) is defined in
"nExpectedArgs" of dataset[0]
"dysys": generalized fully correlated uncertainties
"dystat": generalized uncorrelated uncertainties
"dyprt": generalized partially correlated uncertainties





Part-2. Data for automatic renormalization

"compNotes": show whether Monit-Err recalculated
"DECAY-DATA" present old and new decay data if any
"autoCorrNotes": show how corrections can be done
"c5mon" present monitor data: normalization energy,
"m0", "dm0" – old monitor data and uncertainties
"m1", "dm1" – new monitor data
"Fc0" – factor to be applied for data value

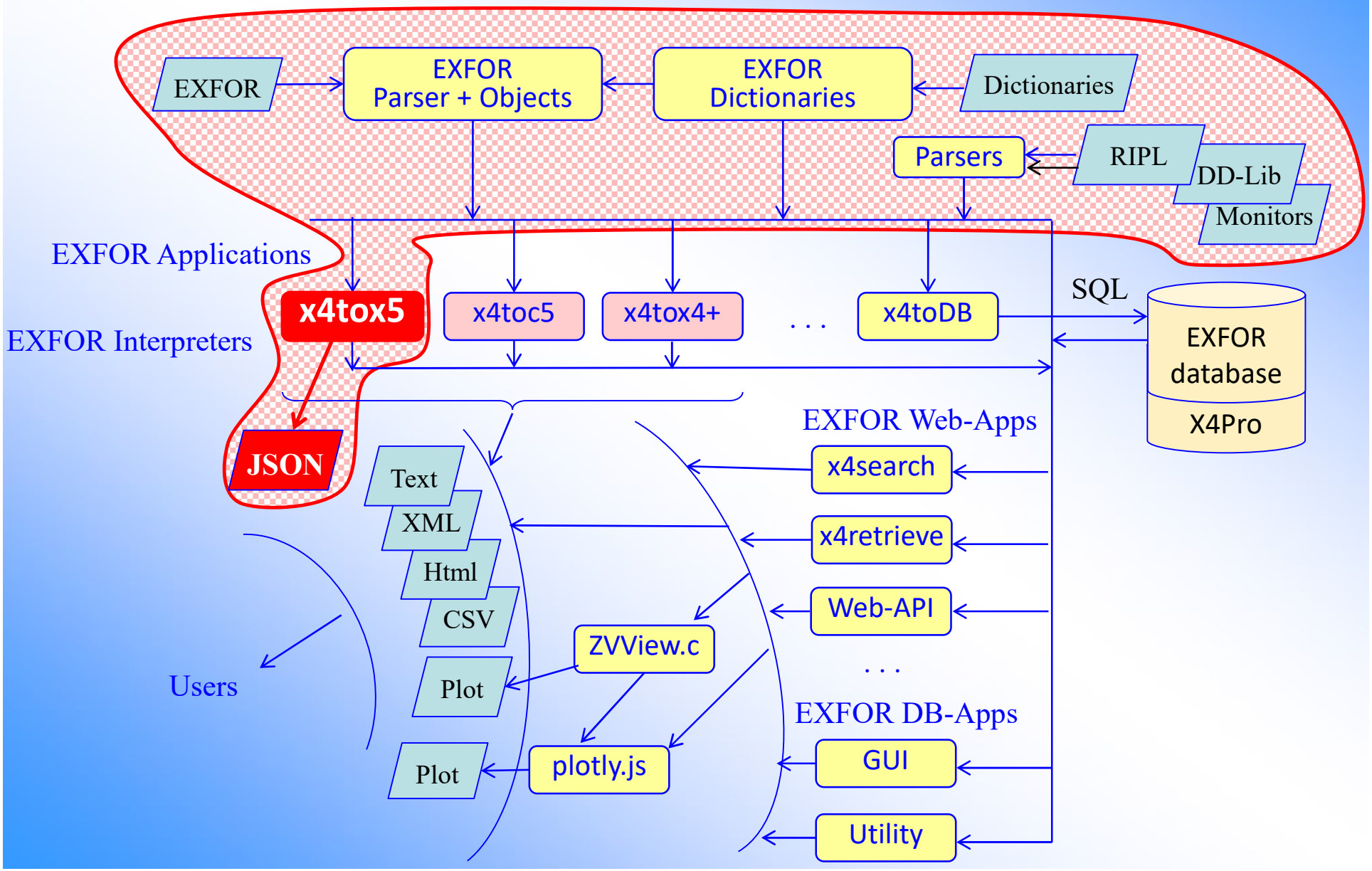
Part III.

Data distribution paradigm.

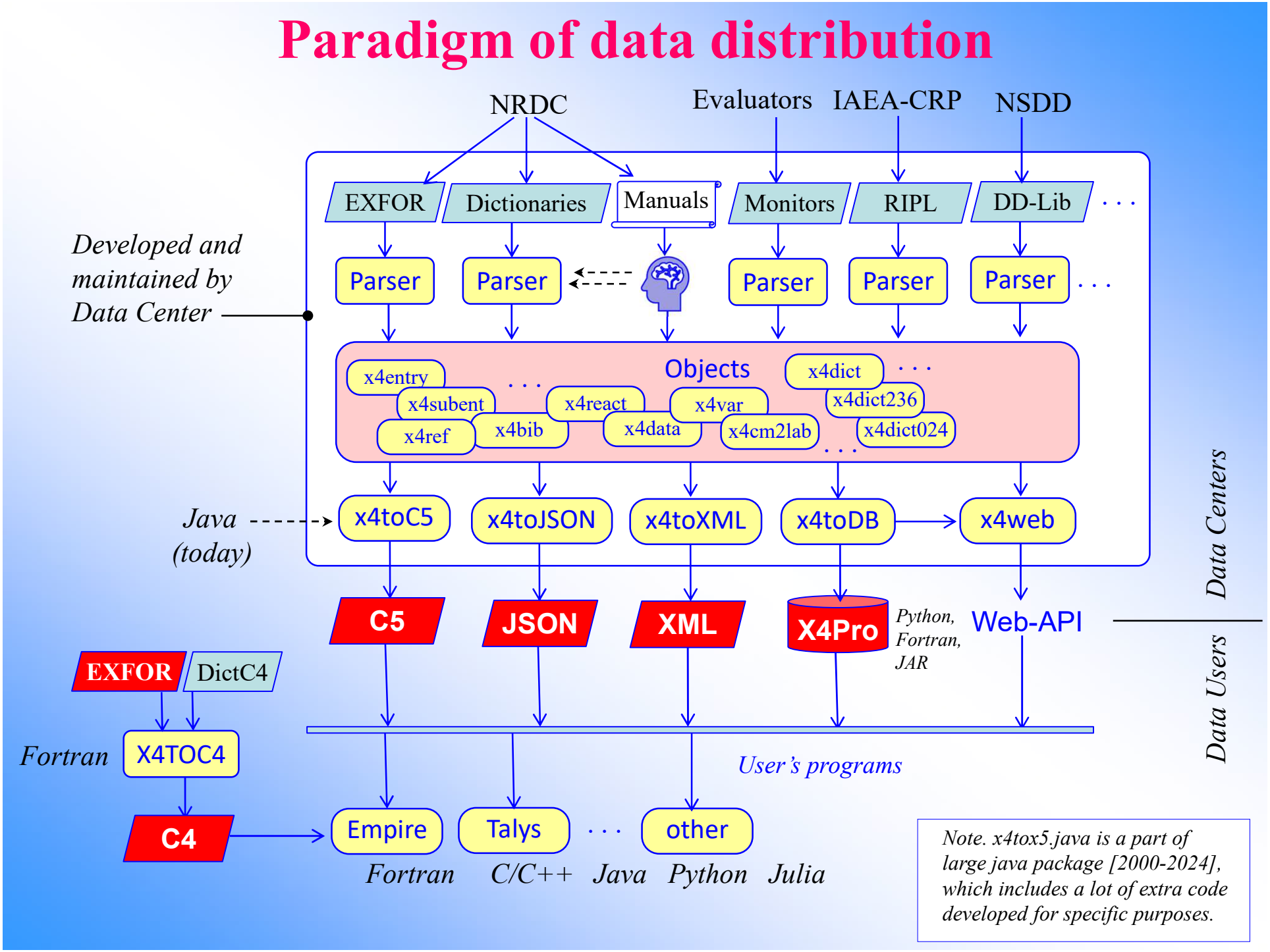
System design.

X5json in EXFOR Java codes and data flows

Initially x4tox5.java was created as an extension of EXFOR database retrieval system to produce additional JSON output.



Paradigm of data distribution



Overview: EXFOR codes and systems

Fortran

Built-in parsers

1986 X4TOC4 ZCHEX *1998*

D.Cullen, A.Trkov, V.McLane, V.Zerkin

C

exfor.c *Parser V.Zerkin 1999*

x4toXmlSql correction

Java

JANIS *N.Soperra et.al. 2004*

x4parsers/classes, x4toc5 /x4+/x5/xml/csv/web/db,... *V.Zerkin 2005-2024*

Python

x4i GPLv2 *D.Brown 2011*

x4i3 GPLv2 *A.Fedynitch 2020*

exfor-parserpy *G.Schnabel 2019*

exforparser, exfor_dict., exfor_json,... *S.Okumura 2022(?)*

ForEX: X4TOJ4, J4TOX4, DICA2J,... *N.Otsuka 2024*

x4d, x4py,... *V.Zerkin 2024*

Fortran, C, Java codes:

1. *Functioning many years (10~30)*
2. *Proven quality*
3. *Powerful (a lot of functionality)*
4. *Produce useful results already now*
5. *Needs intermediate step to deliver data to other languages (C5/XML/CSV/JSON/X4Pro/Web-API)*
6. *Large, need professional maintenance*

READY

Python codes:

1. *Can be used as is in Python codes (good for young generation using Python)*
2. *Easier to read and understand*
3. *Unknown quality (debug)*
4. *Needs JSON/XML/text to feed other languages*
5. *Require deep study of EXFOR, time and efforts to implement full functionality existing in Java*

NOT READY

Concluding remarks

1. X5 provides enriched EXFOR in JSON
    1) X5 gives comprehensive presentation of EXFOR content
    2) X5 structure reproduces data hierarchy of EXFOR file
    3) X5 presents original data values and data in computational form
    4) X5 includes interpretation of meta-data from EXFOR-Dictionaries
    5) X5 presents EXFOR data given as extension of free-text in coded way
    6) X5 is enriched by data from external data sources: monitor and standard cross sections data, decay data from ENSDF
    7) X5 allows to perform automatic data renormalization on monitor cross sections, decay and monitor-decay data
2. X5 is available via Web retrieval system (including Web-API). Full EXFOR translated to X5 is available on GitHub and on NDS download area
3. x4tox5 is implemented on Java as part of NDS EXFOR package

Thank you.

Citing of the materials of this presentation should be done with proper acknowledgement of the author