



NSR Modernization

David Brown, Ben Shu, Ramon Arcilla, Boris Pritychenko
National Nuclear Data Center,
Brookhaven National Laboratory, Upton NY, USA

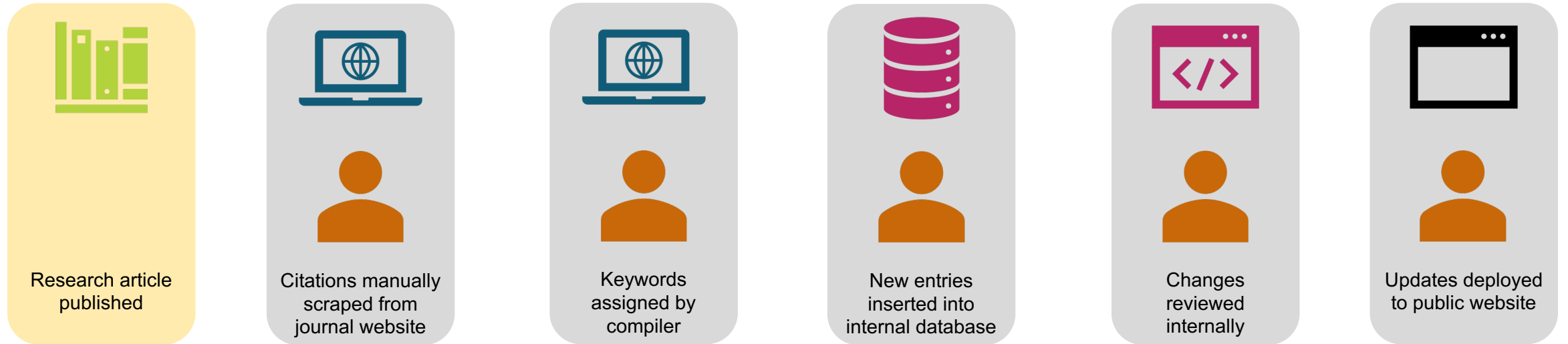
NRDC, Madrid Spain,
19 June 2025



NSR Workflow

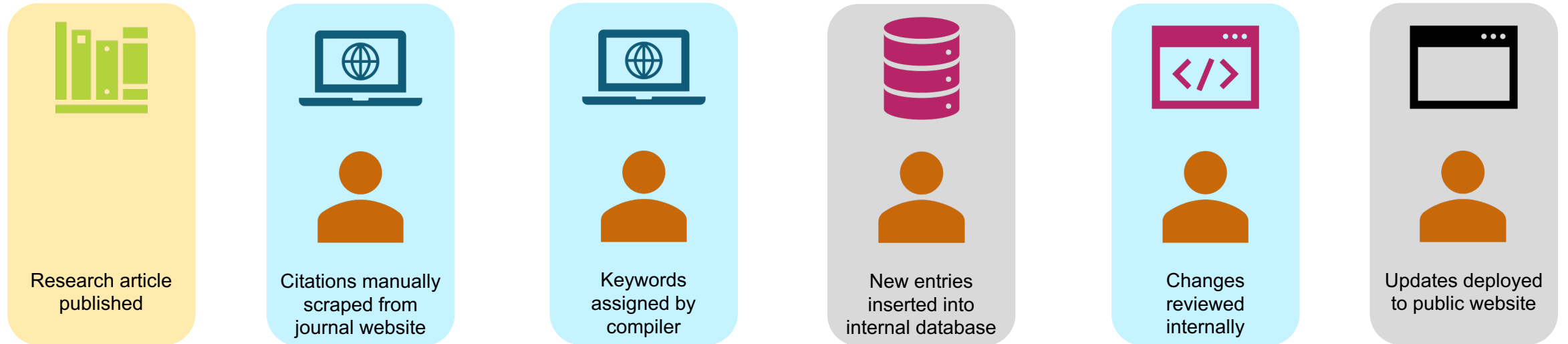
Ben Shu

Previous Workflow



- All steps handled by human beings
 - But we have no more contractors as of April 2025

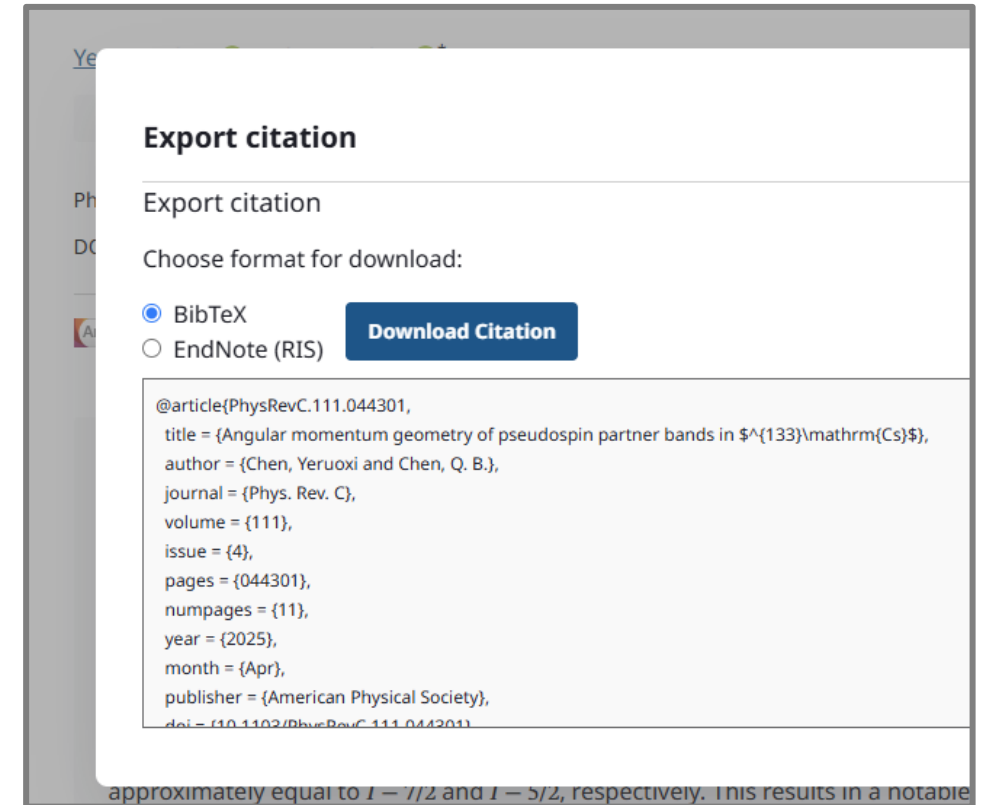
Previous Workflow



- Blue steps are rate determining; are the target for improvements
 - Legacy process must be carefully documented

Web scraping

- Article citations formerly obtained by people searching in websites
 - **Ex. Physical Review C (PRC)**
 - Click on one article, download BibTeX
 - Rinse and repeat for an entire issue
 - Vol 11, Issue 3 – 85 articles added
- This can be automated by web scraping tools



Web scraping

- Article citations formerly obtained by people searching

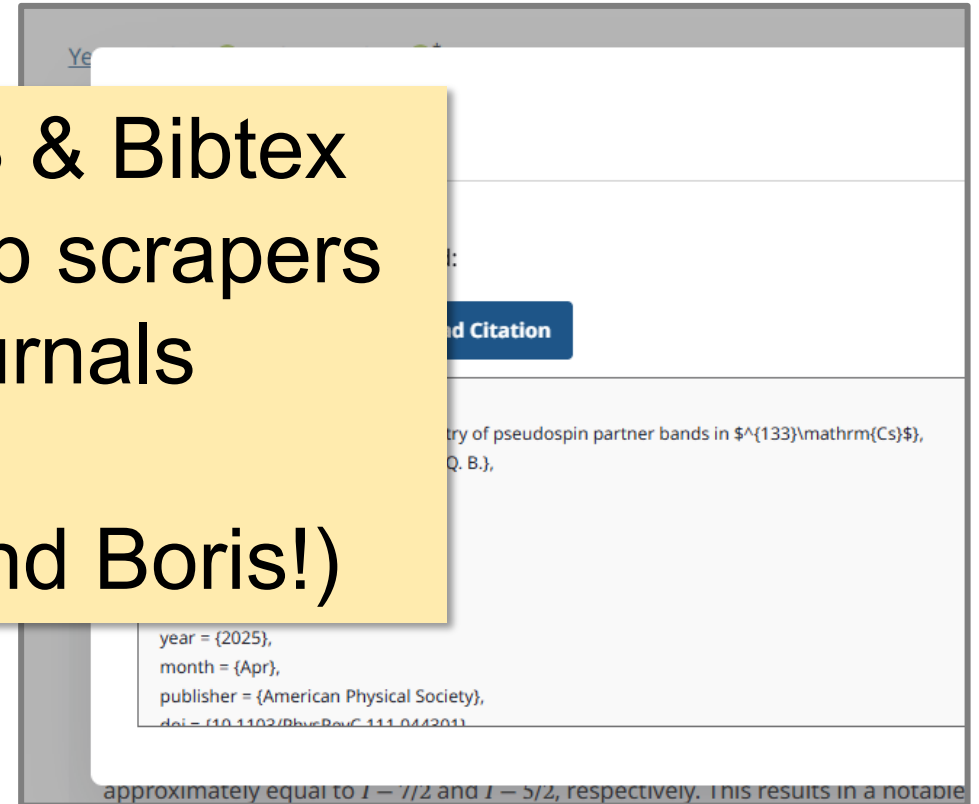
- **Ex. Physical Review C**

- Click on one
- Rinse and repeat
- Vol 11, Issue 1

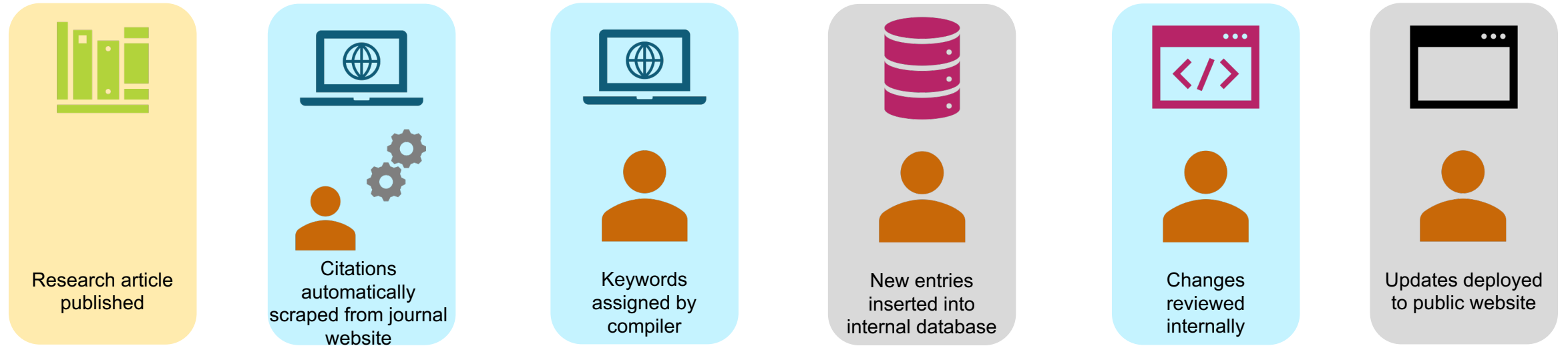
We now have RIS & Bibtex translators and web scrapers for several journals

(thank you Ben and Boris!)

- This can be automated by web scraping tools



Slightly Improved Workflow



- Simple scrapers have sped up ingest from 1 week to 1 hour
- We've sped up article ingest, but we can go faster!

Development Workflow

- **scrape-from-prc**

- Automatically scrapes and processes PRC articles
- Connects to PRC website to obtain articles
- Downloads BibTeX, converts to NSR "exchange"

```
@article{PhysRevC.111.044301,  
  title = {Angular momentum geometry of pseudospin partner bands in  $^{133}\mathrm{Cs}$ },  
  author = {Chen, Yeruoxi and Chen, Q. B.},  
  journal = {Phys. Rev. C},  
  volume = {111},  
  issue = {4},  
  pages = {044301},  
  numpages = {11},  
  year = {2025},  
  month = {Apr},  
  publisher = {American Physical Society},  
  doi = {10.1103/PhysRevC.111.044301},  
  url = {https://link.aps.org/doi/10.1103/PhysRevC.111.044301}  
}
```

```
<KEYNO   >2025CHP  
<HISTORY >C20250409  
<CODEN   >JOUR PRVCA 111 044301  
<REFERENCE>Phys.Rev. C 111, 044301 (2025)  
<AUTHORS >Y.Chen, Q.B.Chen  
<TITLE   >Angular momentum geometry of pseudospin partner bands in  $^{133}\mathrm{Cs}$   
<DOI     >10.1103/PhysRevC.111.044301
```

Development Workflow (contd.)

- **update-prc-errata**

- Checks for updates to existing NSR entries
- Preserves key number, title, keywords, selectors, etc.

```
<KEYNO >2025WAP
<HISTORY >C20250409
<CODEN >JOUR PRVCA 111 049901
<REFERENCE>Phys. Rev. C 81, 037302 (2010); Erratum Phys.Rev. C 111, 049901 (2025)
<AUTHORS >E.H.Wang, S.H.Liu, J.H.Hamilton, A.V.Ramayya, Y.X.Luo, J.O.Rasmussen,
J.K.Hwang, A.V.Daniel, G.M.Ter-Akopian, S.J.Zhu, W.C.Ma
<TITLE >Erratum: Nuclear structure of the odd-odd N=85
140}Cs [Phys. Rev. C 81, 037302 (2010)]
<DOI >10.1103/PhysRevC.111.049901
```

Updated NSR entry

```
<KEYNO >2010LI10
<HISTORY >A20100426 M20250409
<CODEN >JOUR PRVCA 81 037302
<REFERENCE>Phys. Rev. C 81, 037302 (2010); Erratum Phys.Rev. C 111, 049901 (2025)
<AUTHORS >E.H.Wang, S.H.Liu, J.H.Hamilton, A.V.Ramayya, Y.X.Luo, J.O.Rasmussen,
J.K.Hwang, A.V.Daniel, G.M.Ter-Akopian, S.J.Zhu, W.C.Ma
<TITLE >Nuclear structure of the odd-odd N=85 neutron-rich nucleus {+140}Cs
<DOI >10.1103/PhysRevC.111.049901
<KEYWORDS>RADIOACTIVITY {+252}Cf(SF); measured E|g,I|g,|g|g-coin, |g|g(|q) using
```

Erratum scraped from PRC

Development Workflow (contd.)

- **save-sql-backup**
 - Saves a copy of NSR using **mariadb-dump**
 - Runs *before* any database changes
 - Can be used as a restore point, just in case

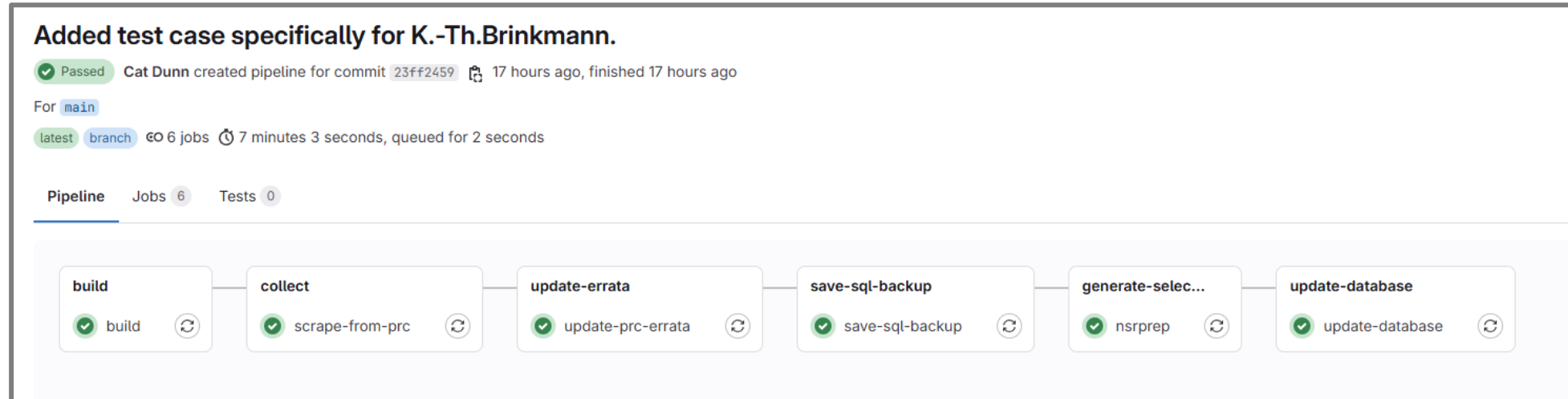
Development Workflow (contd.)

- **nsrprep**
 - Based on legacy web app used by compilers
 - Performs format/validation checks
 - Looks for unknown author names
 - Generates selectors used for NSR website searching
 - Only generates these for **keyworded** entries
 - If an entry is not keyworded, it cannot be found with:
 - **Quick Search:** Nuclide, Reaction
 - **Indexed Search:** Nuclide, Subject, Reaction, Z, A

Development Workflow (contd.)

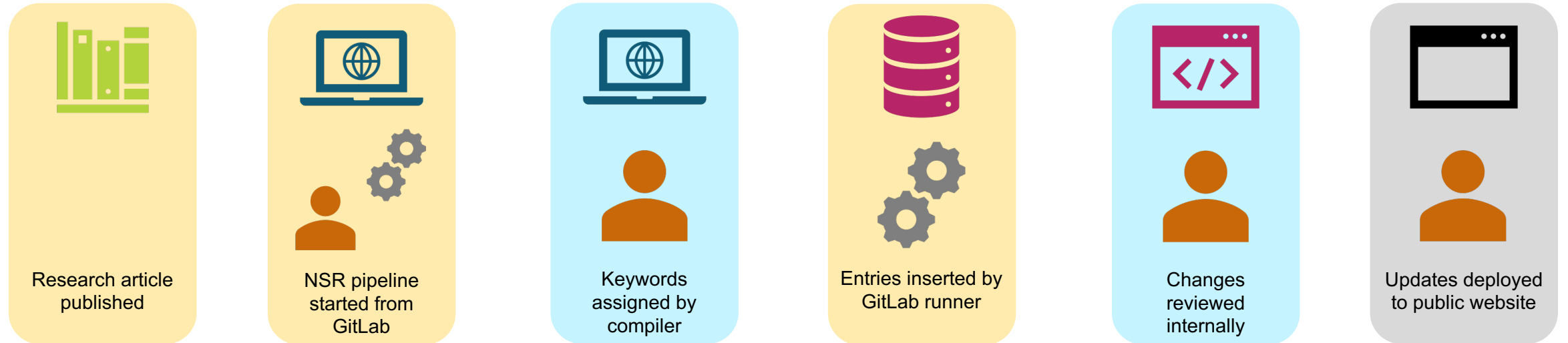
- **update-database**
 - Makes changes to **nsr_automated**
 - Internal only – not deployed to NSR website
 - Inserts new entries for standard PRC articles
 - Updates existing entries for PRC errata
- **Issue:** Currently no way to fix errors before insertion
 - i.e. format errors, PRC typos
 - Fixes require direct SQL changes or restarting pipeline

Development Workflow (contd.)



- Pipeline functional as of April 3rd, 2025
 - No setup/installation - only GitLab access required
 - Click a few buttons, go do something else

A Much Faster Workflow



- Full pipeline takes minutes to run
- We are not taking advantage of the full power of gitlab

NSR Exchange format can be modernized

- The current NSR "exchange" format...

<KEYNO	>	2023WIZZ	Unique ID
<HISTORY	>	A20230605	Entry/modification dates
<CODEN	>	CONF Sacramento...	Source type and information
<REFERENCE>		Proc.15th.Intern...	Citation information
<AUTHORS	>	D.Wiarda, ...	List of authors
<TITLE	>	Modernization...	Formatted title
<KEYWORDS>		[N/A]	Sentences about content
<SELECTRS>		[N/A]	Structured search keys

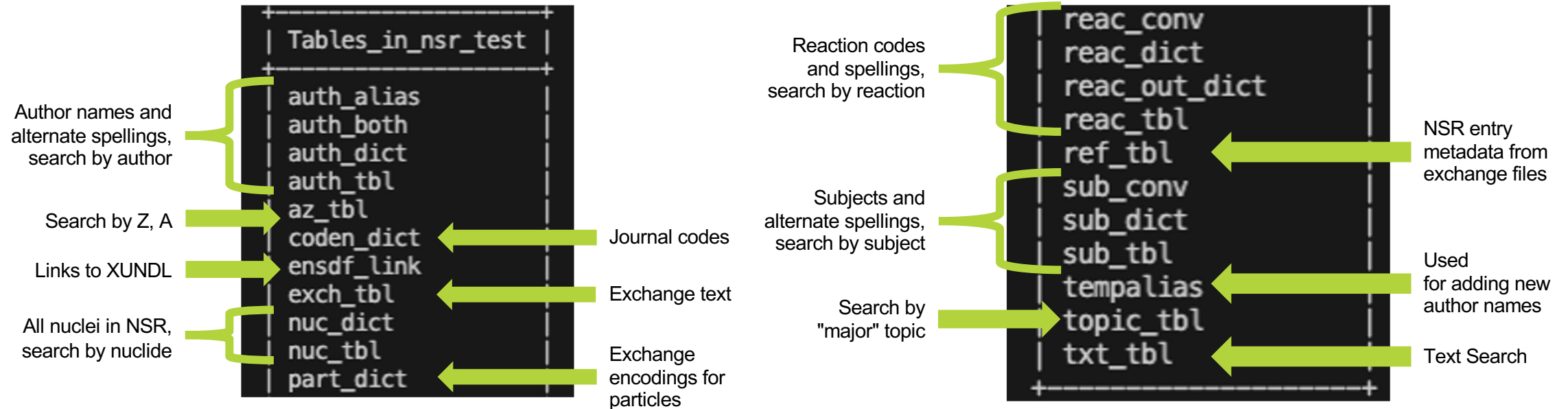
NSR Exchange format can be modernized

- ...can be translated into JSON...

```
{ } 2023WIZZ.json IM X
databases > nsr > entries > { } 2023WIZZ.json > title
1 {
2   "_id": "2023WIZZ",
3   "keyNumber": "2023WIZZ",
4   "publicationYear": 2023,
5   "entryDate": "2023-06-25",
6   "codenText": "CONF Sacramento(Nucl Data for Sci and Technol) Proc,P12005,Wiarda",
7   "referenceType": "CONFERENCE",
8   "referenceText": "Proc.15th Intern.Conf.Nuclear Data for Science and Technology (ND
9 >   "authors": [ ...
69 ],
70   "title": "Modernization efforts for the R-Matrix code SAMMY",
71   "doi": "10.1051/epjconf/202328412005",
72   "keywords": [ ...
```

NSR Exchange format can be modernized

- ...but the exchange files don't tell the full story
- NSR's SQL database has supporting tables:



What does this get us?

A JSON format enables:

- A JSON-schema
- Format and correctness checkers
- An editor
- Better NSR archive option
- An AI/ML-friendly format!

Future work
for a summer
student?

Even without these, with gitlab we have

- Version control
- Built in review step (branch merge process)
- We use this to great effect in ENDF & JEFF



National Nuclear Data Center

Databases

Structure & Decay

Reactions

Researches

Brookhaven National Laboratory

Nuclear Science References (NSR)

Nuclear Science literature database, containing about 250,000 articles covering more than 120 years of nuclear science research. Approximately 80 journals are routinely scanned for relevant articles; it also includes PhD thesis and private communications. Each article in NSR is assigned a distinctive 8-character primary key, known as 'key number', and a set of keywords that briefly describe the article's content. These keywords follow strict rules since they are used in web search forms. Typically, an article is incorporated in NSR within a few weeks after its web publication. About 50,000 articles are added per year. All the compilation effort is supported by the US Nuclear Data Program and coordinated by the National Nuclear Data Center at Brookhaven National Laboratory, which is also responsible for its web dissemination.

242,120

Articles

196,773

Keyworded Abstracts

108,272

Authors

8,578

Nuclides

8,914

Reactions

382

Decays

Deposition Summary		Dataset Details	
Depositor:	Boris Pritychenko	Total articles:	242120
Contact:	pritychenko@bnl.gov	Total keyworded abstracts:	196773
Deposition date:	2022/04/06	Total authors:	108272
Last modified:	2022/04/06	Nuclides:	8578
DOI:	10.18139/nndc.nsr/1866086	Reactions:	8914
		Decays:	382

Latest Dataset

FileDate

<https://www.nndc.bnl.gov/nsrarchivals/>

- neutrons
- Pinned
- Issues 208
- Merge requests 25
- Manage
- Plan
- Code
- Merge requests 25
- Repository
- Branches
- Commits
- Tags
- Repository graph
- Compare revisions
- Snippets
- Locked files
- Build
- Secure
- Deploy
- Help

ENDF / library / neutrons / Merge requests / 1110

4

1

2

Q Search or go to...

Draft: Review/n-017_CL037

[Open](#)
Gustavo Nobre requested to merge [Review/n-017_CL037](#) [into phase2](#) 2 months ago

[Overview](#)
[Commits](#)
[Pipelines](#)
[Changes](#)

[Add a to-do item](#)

Phase I Review

ENDF/B-VIII.1, Neutrons Sublibrary

- Filename: n-017_CL037.endf
- Sublibrary: Neutrons
- Material: 17-Cr-37 (MAT=1731)
- Evaluators: Sayer/Guber/Kawano/Hanselman+
- Submitter: Kenneth Hanselman
- Submitter email: khanselman@lanl.gov
- Review form generated at: 11/04/2025 13:43:38
- Reviewer: _____
- Reviewer email: _____
- Date: _____

Instructions

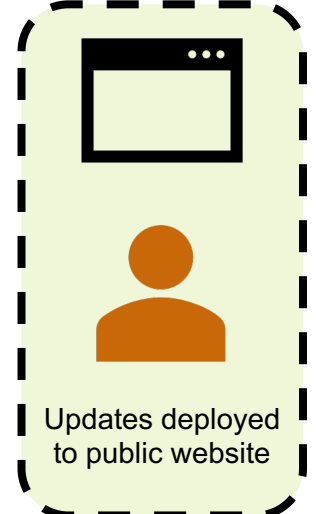
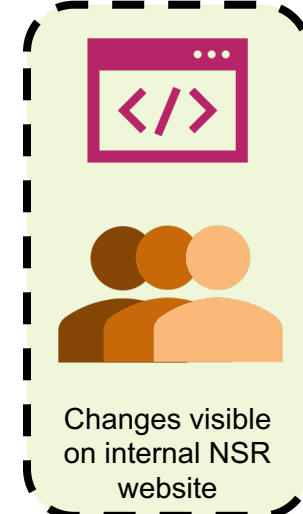
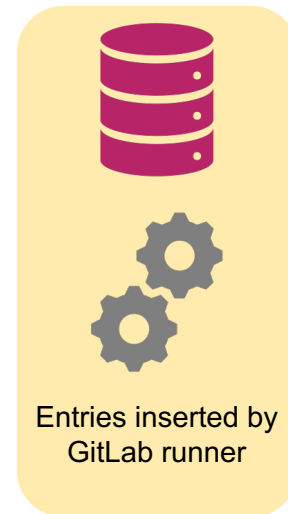
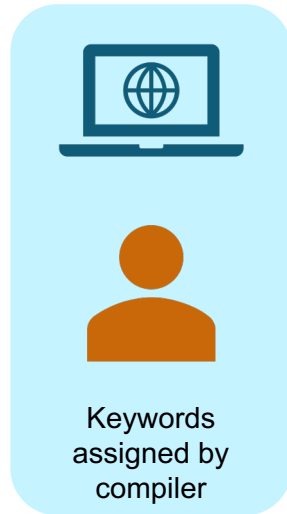
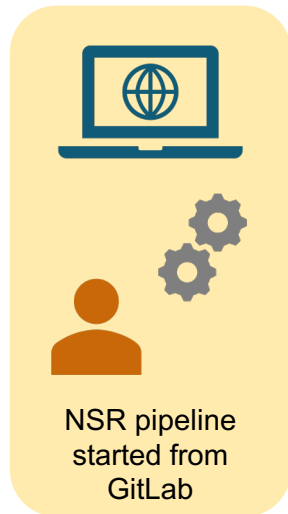
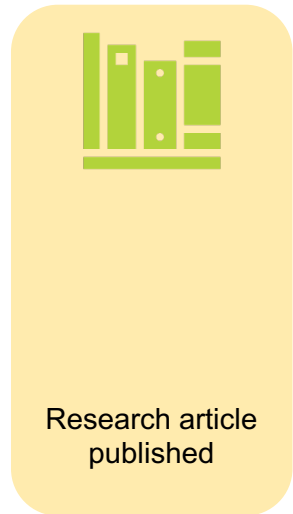
You are being asked to review the evaluation for [n-017_CL037.endf](#) from branch [phase1](#). Please edit this file on the [Review/n-017_CL037](#) merge request tracker page at 1110. You can do this simply by clicking the "edit" button and filling the empty fields. This document uses the Markdown format (see <https://docs.gitlab.com/ee/user/markdown.html> for a reference).

File Contents

Questions to consider with each dataset: Is the current dataset an improvement over the existing dataset in the ENDF evaluation? Is the current dataset (e.g. cross section) some form of standard or reference? If so, how does the dataset compare to the reference? Does the current set take into account all relevant differential data? Data and covariance adequacy should be given on a scale of 0-5 with 5 being the highest. Use N/A if not given in the ENDF file.

Reaction (MT)	1	2	3	4	6	32	33	Data Adequacy	Cov. Adequacy	Comments
0 Assignees None - assign yourself 0 Reviewers Approval is optional: Assign None - assign yourself Labels None Milestone None Time tracking No estimate or time spent 1 Participant										

With Gitlab, we can ensure quality while getting speed



- With a QA process in hand, and an automated workflow, now we can target the hardest step

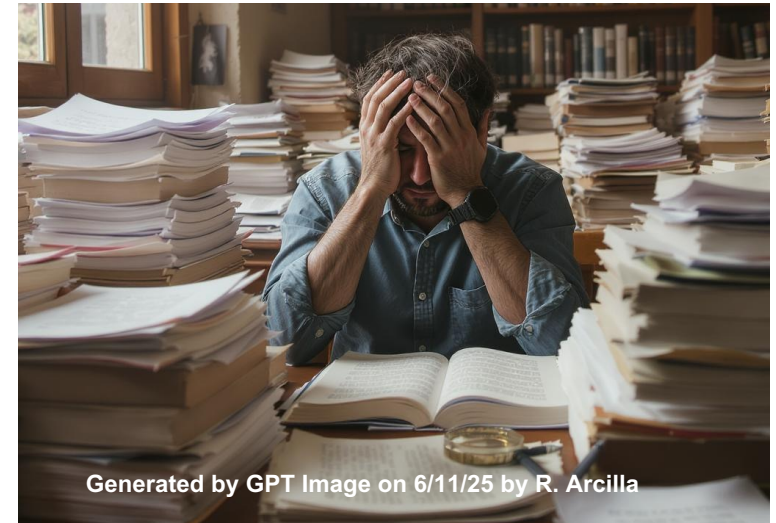
Keywording

Currently a compiler must produce keyworded abstract – time consuming and requires somewhat special skills

As of FY25, NNDC has no contractors

Options:

- Teach more people to do it (ENSDF/ENDF evaluators, students, NNDC staff?)
- Automate it?



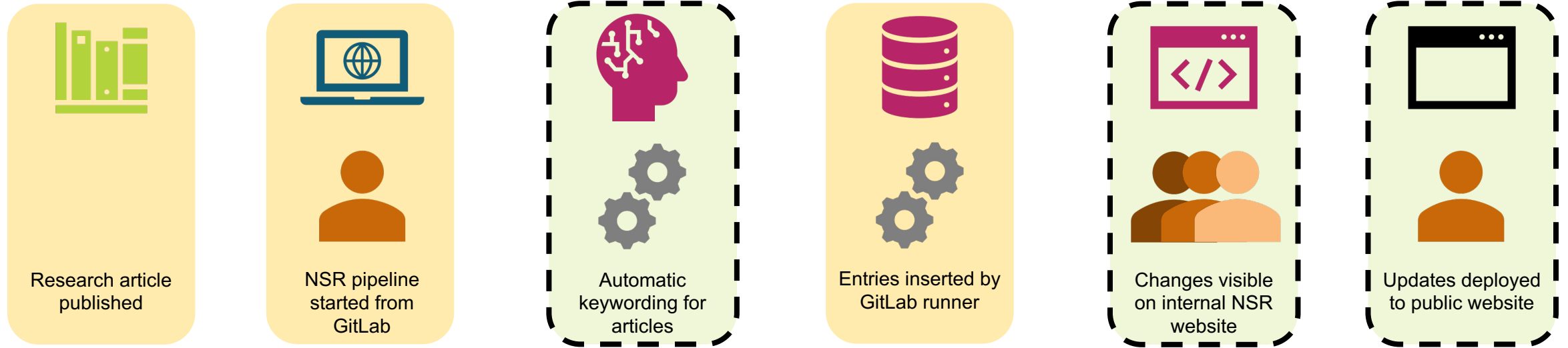
Nuclear Science References Coding Manual

D.F. Winchell
National Nuclear Data Center
Brookhaven National Laboratory
Upton, New York, USA

May 24, 2007

<https://www.nndc.bnl.gov/nsr/docs/nsr-coding-manual.pdf>

Future Workflow



- Need to fill in missing steps
 - Keyword assignment via chatNSR
 - Tools for human reviewers to make edits

chatNSR: An AI-Enhanced Nuclear Science References Knowledge Base

Ramon Arcilla

Motivation

Why automate keyword abstract generation?

Volume Challenge

- Growing nuclear science literature (~5,000+ annually) strains current manual keywording capacity.

Time Intensive

- Manual keywording requires ~3.5 hours (average) per article.

Limited Coverage

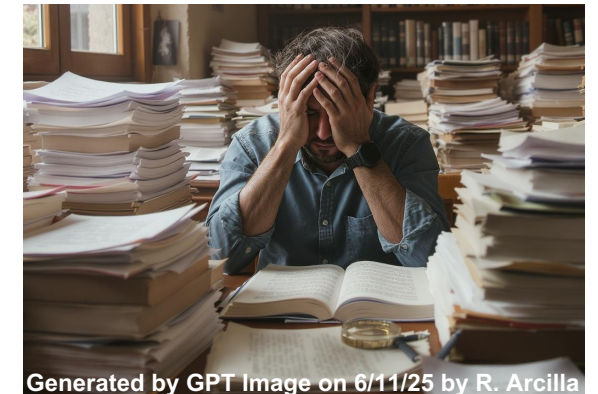
- Manual keywording limits timeliness and breadth of NSR knowledge base content.

Discovery Issues

- Inconsistent keywording limits paper discoverability.

Cost Inefficiency

- Manual keywording demands significant resources.



The chatNSR Project

Current Focus:

- Keywording, the most tedious and costly activity

Key AI Technologies in Use

- Open-Source Large Language Models (LLMs)
 - Understand user queries/instructions and generate keyword abstract
- Cache-Augmented Generation (CAG)
 - Loads entire article into LLM's extended context window (Knowledge Cache) for efficiency, more accurate keywording
- Retrieval-Augmented Generation (RAG)
 - Finds contextually relevant parts of article in vector DB for LLM's use in keywording
- System Prompt Engineering (SYSPROMPT)
 - User instructions to guide LLMs for consistent, accurate keywording

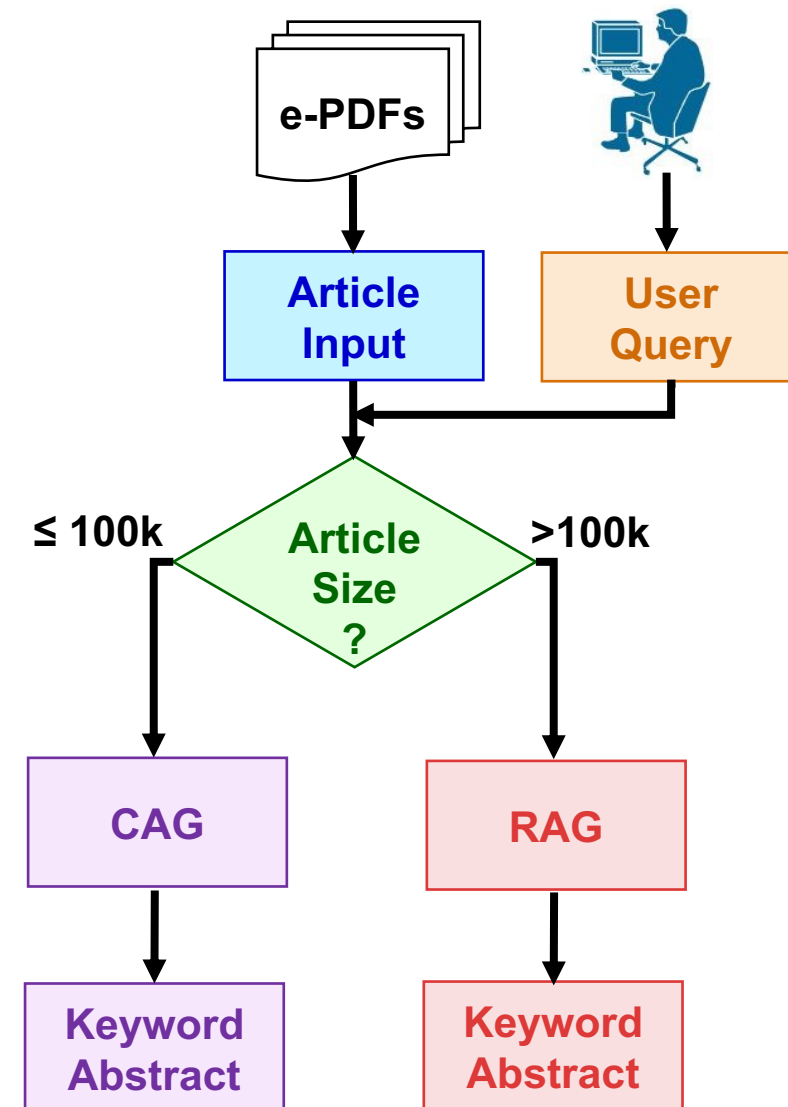


Generated by GPT Image on 6/10/25 by R. Arcilla

The chatNSR Project

Architectural Overview: Two-Path Automation Workflow

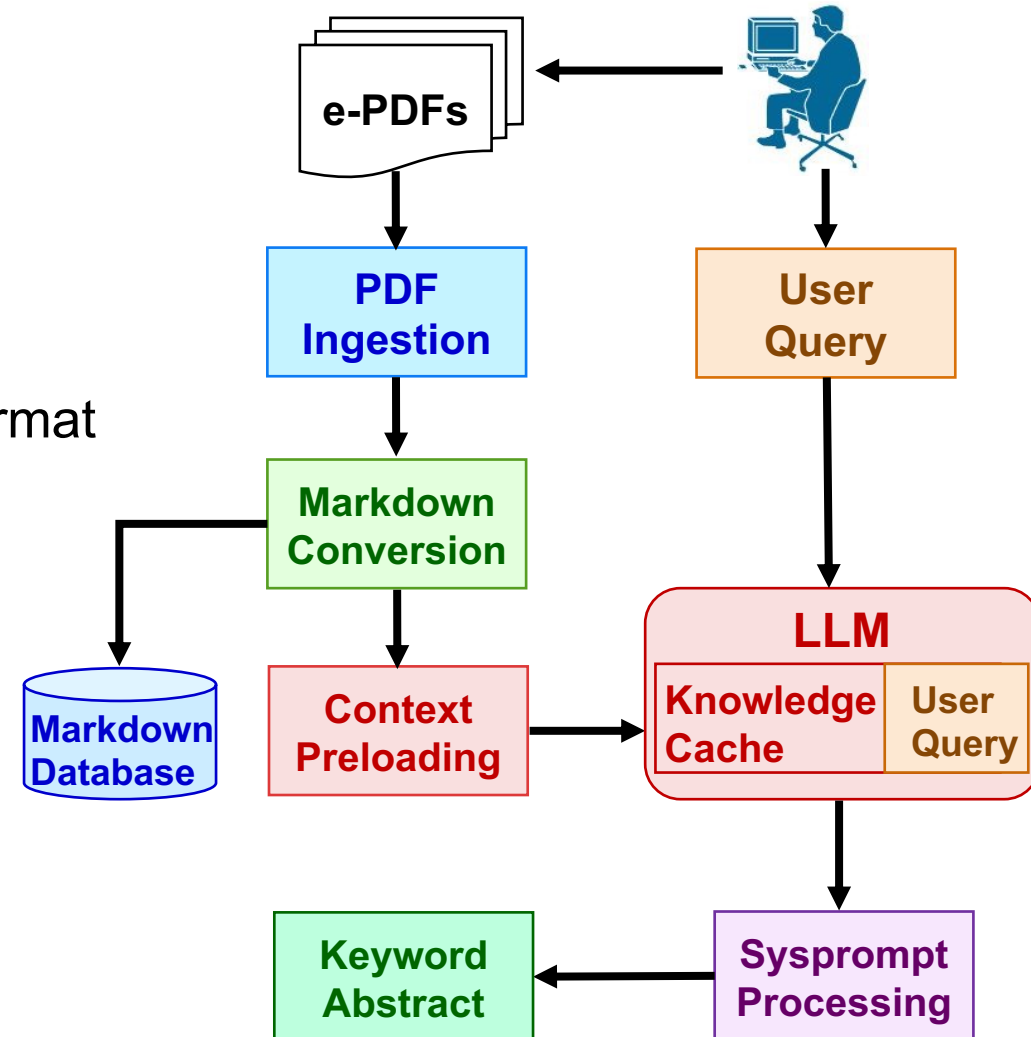
- Document Input
 - Article enters the automation pipeline
- Size Assessment
 - System determines article's token count
- Path Selection
 - CAG for smaller articles ($\leq 100k$ tokens = 100 pages), RAG for larger ones ($> 100k$ tokens)
- Keyword Output
 - Keyword abstracts generated via optimal path



Cache-Augmented Generation (CAG)

Workflow

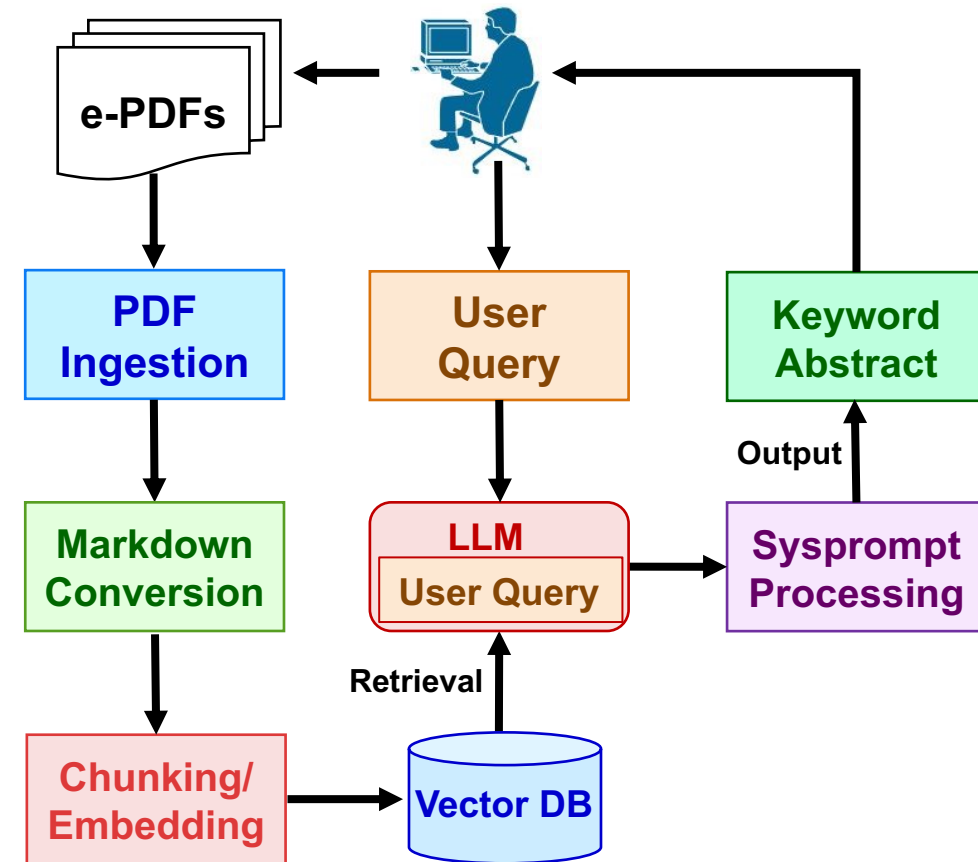
- PDF Ingestion
 - Article enters system and undergoes initial parsing
- Markdown Conversion
 - Content structure preserved in clean Markdown format
- Context Preloading
 - Full article loaded with Knowledge Cache creation
- System Prompt Processing
 - LLM guided by engineered instructions
- Response Output
 - Keyword abstracts delivered to user



Retrieval-Augmented Generation (RAG)

Workflow

- Document Processing
 - PDF ingestion and conversion to markdown format.
- Chunking & Embedding
 - Document segmentation and vector representation creation.
- Retrieval Mechanism
 - Query-based selection of relevant document sections.
- Generation with Prompting
 - LLM processes retrieved chunks with specialized instructions.
- Output Delivery
 - Keyword abstract returned to requesting user.



Advantages of the Two-Path Automation Approach

Universal Scalability

- System handles articles of any length.
- Processing adapts to content size automatically.

Consistent Quality

- Keyword abstract coherence maintained across both workflows.
- Keyword abstract remains relevant regardless of article size.

Resource Efficiency

- Computational resources allocated based on article needs.
- Simpler approach (CAG) used when possible.

Workflow Integration

- Unified API endpoint despite dual backend systems.
- Seamless integration with existing document processes.



EXAMPLE: Keywording D.A. Brown's article "Zirconium Evaluations for ENDF/ B-VII.2 for the Fast Region"

The screenshot displays the chatNSR interface, which is an AI-enhanced nuclear science knowledge base. On the left, there are three preview cards showing document thumbnails. The main area on the right contains the chat interface. At the top, it says "chatNSR An AI-Enhanced Nuclear Science References Knowledge Base". Below this, there's a section for "Select Existing Document or Upload New Document" with a dropdown menu showing "Zirconium Evaluations for ENDF-B-VII.2 for the Fast Region". The "Enter your question:" field contains the prompt: "Generate one or more keyword abstracts." Below this is a "Submit Question" button. The response area shows the AI's output, which includes a detailed analysis of the user's request, identifying key elements like isotopes, reaction types, and energy ranges. It then generates a keyword abstract in a structured format, including a "COMPILATION" section with specific details about the zirconium isotopes and the ENDF-B-VII.2 evaluation.

chatNSR
An AI-Enhanced Nuclear Science References Knowledge Base

Select Existing Document or Upload New Document
Zirconium Evaluations for ENDF-B-VII.2 for the Fast Region

Enter your question:
Generate one or more keyword abstracts.

Submit Question

<think> Okay, let's tackle this query step by step. The user provided a detailed prompt about generating keyword abstracts for a compilation or evaluation paper, specifically focusing on Zirconium isotopes and neutron-induced reactions. They also included examples and strict formatting rules.

First, I need to identify the key elements from the context given. The main nuclei involved are various Zr isotopes: ^{90}Zr , ^{91}Zr , ^{92}Zr , ^{93}Zr , ^{94}Zr , ^{95}Zr , and ^{96}Zr . The reactions mentioned are neutron-induced (n, x), with an energy range from 0 to 20 MeV. The quantities calculated include cross sections (σ) and angular distributions ($\sigma(\theta)$), particularly in the resonance region. The model used is EMPIRE-3.1, and comparisons were made against experimental data and evaluated databases like ENDF/B-VII.1.

The user emphasized that for compilation or evaluation papers, only a single "COMPILATION" abstract should be generated, overriding other types. So I must avoid creating separate abstracts for NUCLEAR REACTIONS or others. The format should strictly follow the example provided: 'COMPILATION [list of nuclei](#), E=[energy]; calculated, evaluated [quantities] using MODEL. Compared with [sources].'

I need to ensure all information comes directly from the context without inferring anything. The energy range is explicitly stated as 0-20 MeV, so that's included. The model EMPIRE-3.1 and comparison sources like ENDF/B-VII.1 are mentioned in the text, so they should be part of the abstract.

isotopes: ^{90}Zr , ^{91}Zr , ^{92}Zr , ^{93}Zr , ^{94}Zr , ^{95}Zr , ^{96}Zr . The reactions mentioned are neutron-induced (n, x), with an energy range from 0 to 20 MeV. The quantities calculated include cross sections (σ) and angular distributions ($\sigma(\theta)$), particularly in the resonance region. The model used is EMPIRE-3.1, and comparisons were made against experimental data and evaluated databases like ENDF/B-VII.1.

The user emphasized that for compilation or evaluation papers, only a single "COMPILATION" abstract should be generated, overriding other types. So I must avoid creating separate abstracts for NUCLEAR REACTIONS or others. The format should strictly follow the example provided: 'COMPILATION [list of nuclei](#), E=[energy]; calculated, evaluated [quantities] using MODEL. Compared with [sources].'

I need to ensure all information comes directly from the context without inferring anything. The energy range is explicitly stated as 0-20 MeV, so that's included. The model EMPIRE-3.1 and comparison sources like ENDF/B-VII.1 are mentioned in the text, so they should be part of the abstract.

Also, the user wants each keyword abstract in a code block with markdown formatting, each as a single line. Since this is a compilation paper, only one abstract is needed here. I'll structure it exactly as per the example, making sure to list all Zr isotopes and include all necessary details without any extra information.

Double-checking for any missing elements: nuclei, reaction type, energy range, calculated quantities, model used, comparisons. Everything seems covered. Now, format it correctly within a code block.

</think>

```
COMPILATION  $^{90}\text{Zr}$ ,  $^{91}\text{Zr}$ ,  $^{92}\text{Zr}$ ,  $^{93}\text{Zr}$ ,  $^{94}\text{Zr}$ ,  $^{95}\text{Zr}$ ,  $^{96}\text{Zr}(n, x)$ , E=0-20 MeV; calculated, evaluated  $\sigma$ ,  $\sigma(\theta)$  including resonance region using EMPIRE-3.1 code. Compared with available data and evaluated databases (ENDF/B-VII.1).
```

Human-Generated

COMPILATION ^{90}Zr , ^{91}Zr , ^{92}Zr , ^{93}Zr , ^{94}Zr , ^{95}Zr , $^{96}\text{Zr}(n, x)$, E=0-20 MeV; calculated, evaluated σ , $\sigma(\theta)$ including resonance region using EMPIRE-3.1 code. Compared with available data and evaluated databases.

chatNSR-Generated

COMPILATION ^{90}Zr , ^{91}Zr , ^{92}Zr , ^{93}Zr , ^{94}Zr , ^{95}Zr , $^{96}\text{Zr}(n, x)$, E=0-20 MeV; calculated, evaluated σ , $\sigma(\theta)$ including resonance region using EMPIRE-3.1 code. Compared with available experimental data and evaluated databases (ENDF/B-VII.1, JENDL-4.0).

Current Status

Testing

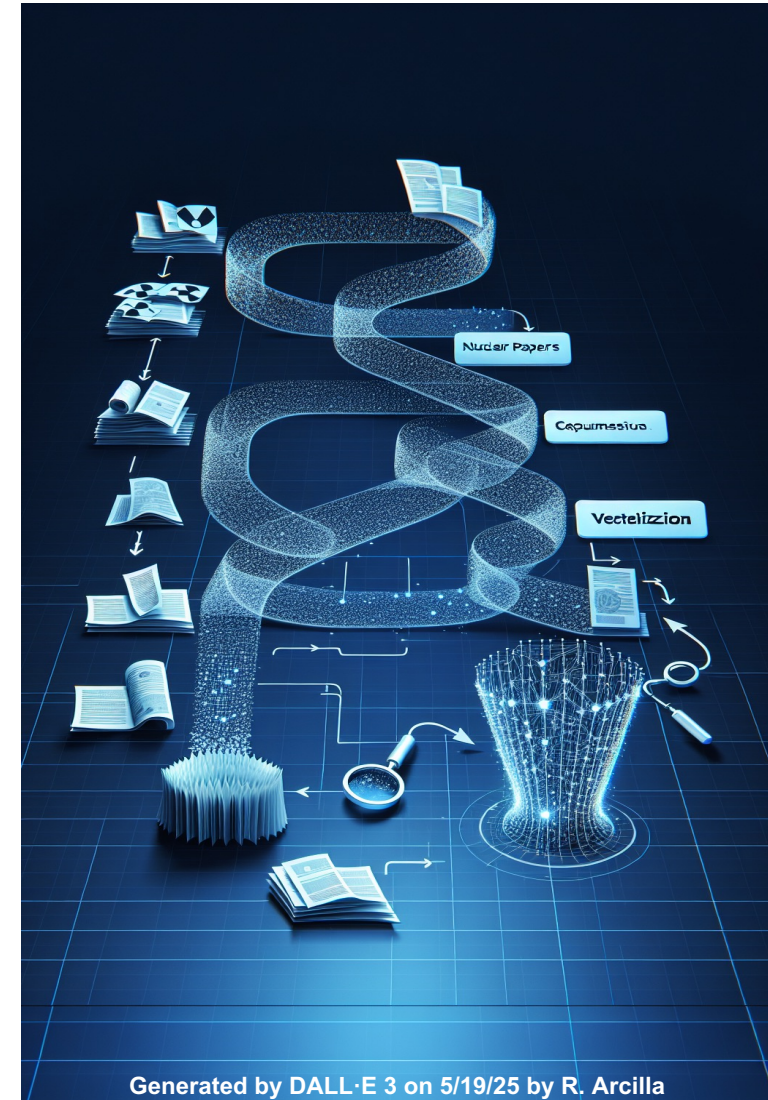
- Dozen open-source LLM's tested
- Dozens of articles (different topics) for LLM training
- Perplexity AI's ***r1-1776:70b*** LLM has highest accuracy

Accuracy

- 80-85% accuracy for articles ≤ 10 pages
- More work needed for articles > 10 pages

Methods and Tools in Use

- Continuous human feedback to LLM
- AI-assisted coding and innovative prompt engineering
- PhysBERT (LBNL) model for embedding and retrieval
- IBM Docling software for PDF-to-Markdown conversion
- Perplexity AI's ***r1-1776*** LLM for keyword generation



Summary and Conclusion

Keywording Automation

- Continuous human feedback improves LLM's accuracy.
- Article (varied topics) high-volume ingestion and training enhances LLM's performance.
- CAG, RAG, System Prompts minimize/prevent LLM hallucination.
- 90% time reduction in keyword abstract generation
- PDF-to-Markdown conversion improves LLM's article understanding

On-premise Keywording

- Open-source LLMs and AI tools demonstrate cost-free, flexible, adaptable keywording
- More secure environment behind BNL FireWall
- Compliant with DOE rules and copyright laws



Generated by DALL-E 3 on 5/19/25 by R. Arcilla

Future Direction

LLM capability and size

- Move to more advanced LLMs with > 70 billion parameters for higher accuracy.

Article Ingestion

- Massive article ingestion with one line command:

```
>>> genabs  articles-directory/  keywords-directory/
```

Multi-modal support

- Processing of article's tables, figures, and equations

Computational resources

- Upgrade of computational resources to meet increased processing needs.



Generated by DALL·E 3 on 5/19/25 by R. Arcilla

**If we can (nearly) fully
automate NSR, then both
XUNDL and EXFOR
should be possible**



Thank you so much for your attention!

This work is sponsored by the Office of Nuclear Physics, Office of Science of the U.S. Department of Energy under Contract No. DE-SC0012704 with Brookhaven Science Associates, LLC.