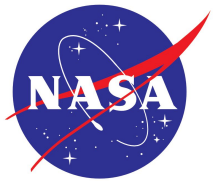
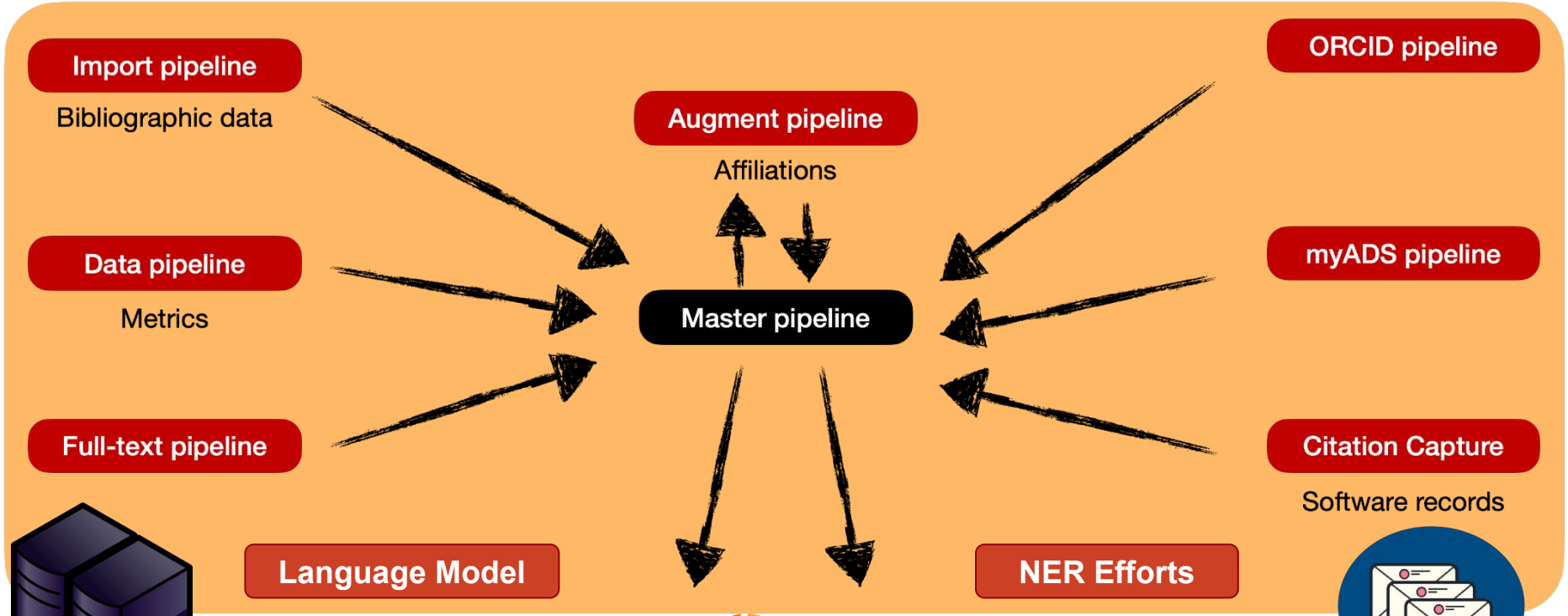
System Development
Backoffice Classic Replacement Progress

Kelly Lockhart, Peter Williams, Jenny Koch, Carolyn Grant, Edwin Henneken and the ADS Team

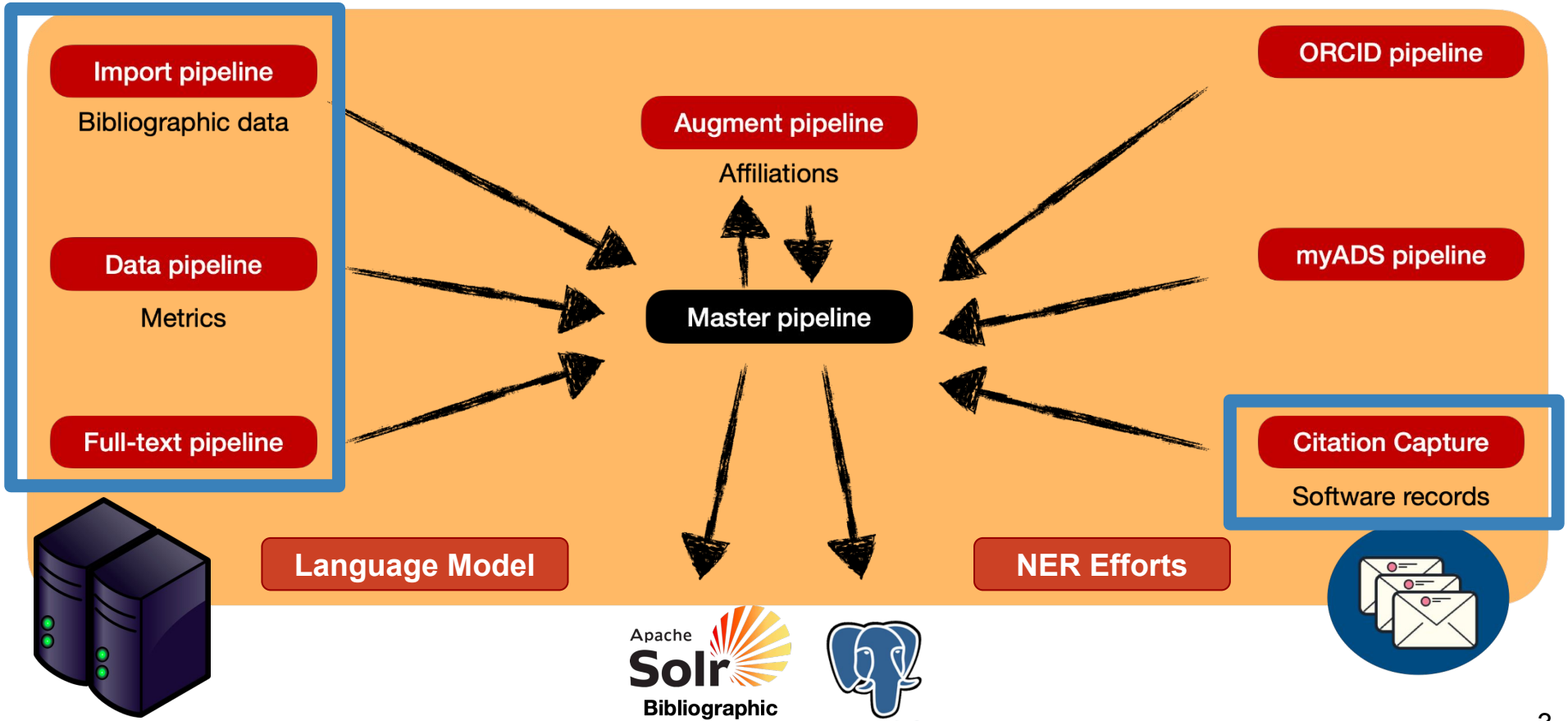
ADS Users Group Meeting, 9-10 Nov. 2022



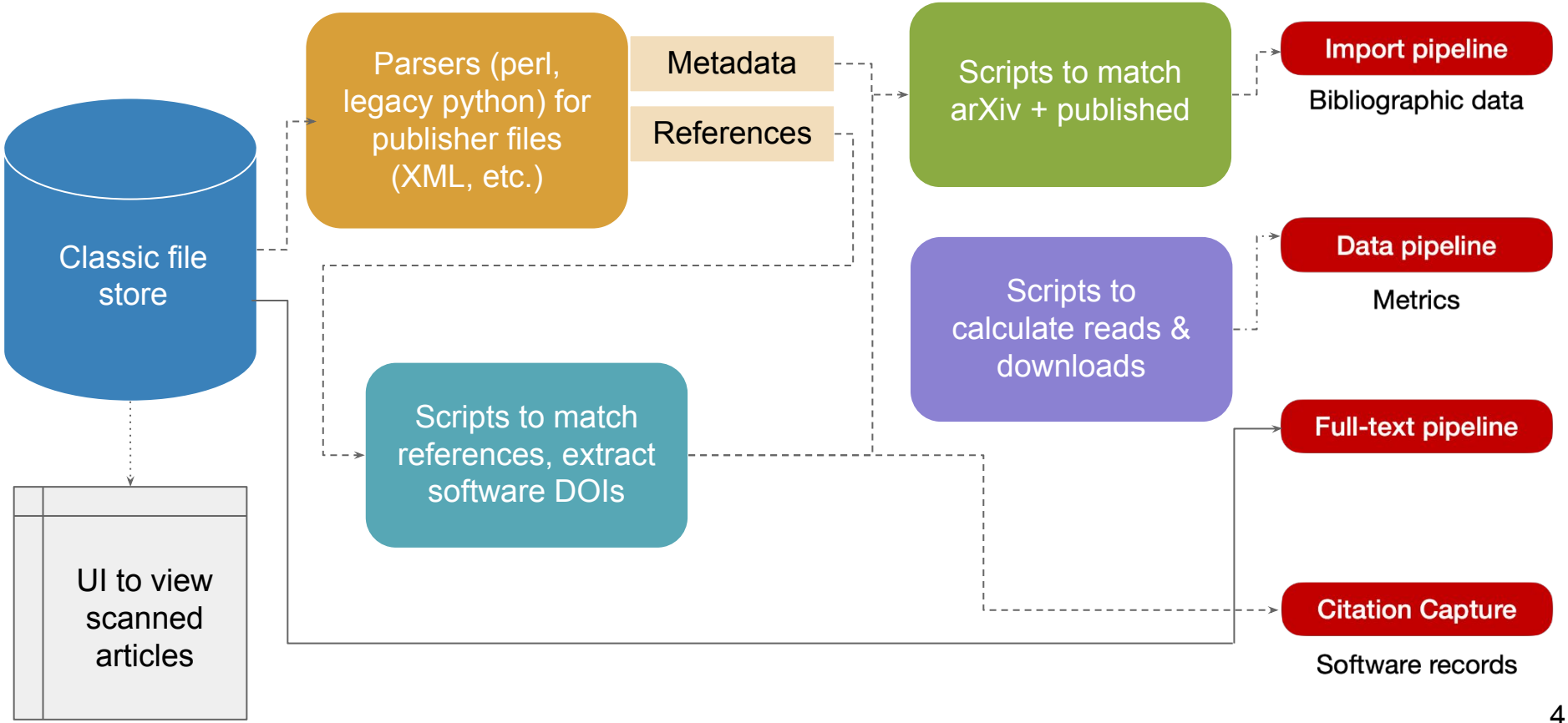
Current pipelines



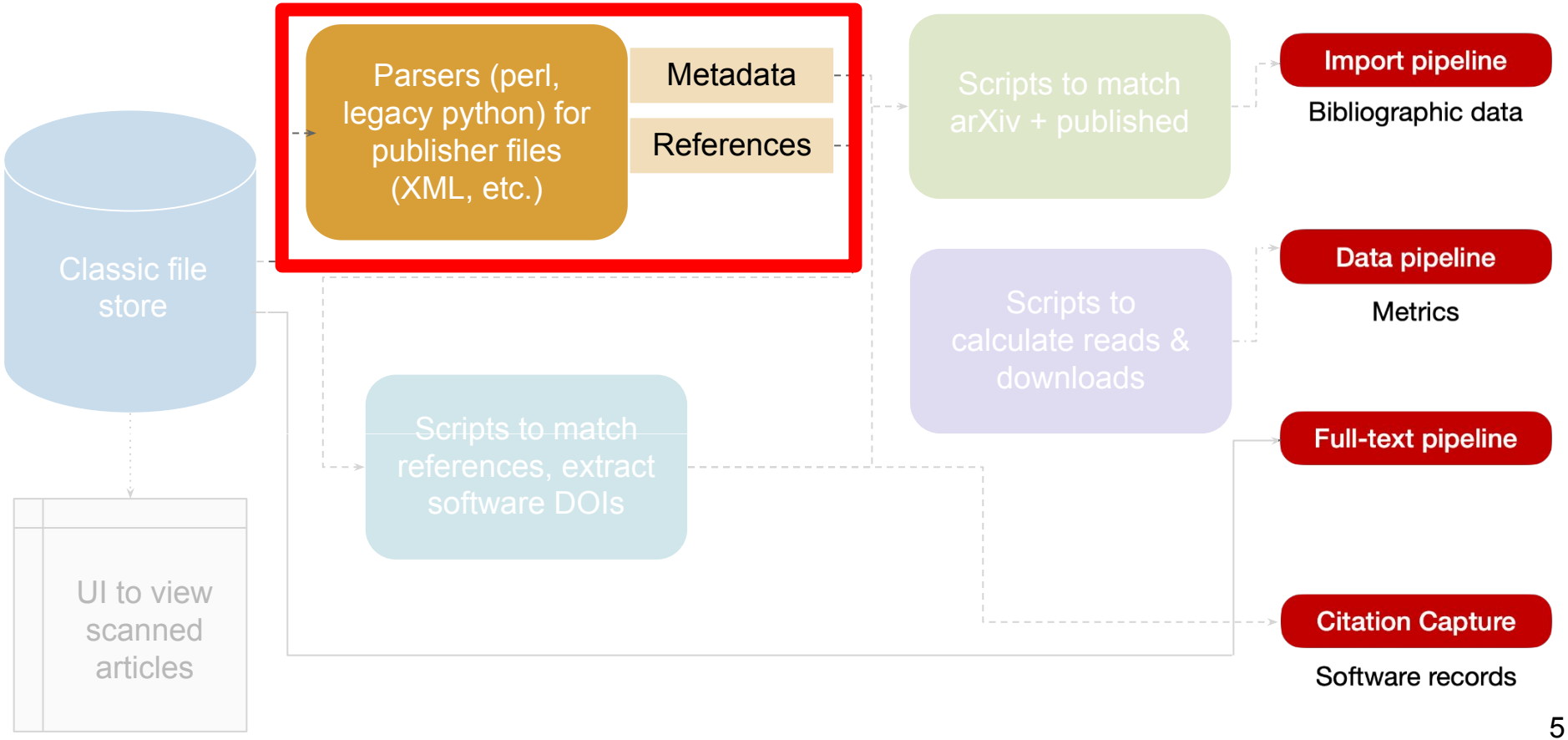
Current pipelines with legacy dependencies



Legacy dependencies



Parsers


Parsers (perl, legacy python) for publisher files (XML, etc.)
Metadata
References
Classic file store
Scripts to match arXiv + published
Import pipeline
Bibliographic data
Scripts to calculate reads & downloads
Data pipeline
Metrics
Scripts to match references, extract software DOIs
Full-text pipeline
UI to view scanned articles
Citation Capture
Software records

Problems with legacy parsers

- Technical issues:
  - Inconsistencies in style, call syntax, and input/output
  - Old, not easily maintainable code: some in perl, some in legacy python
- Project management issues:
  - Parsing code is spread across multiple libraries/pipelines: code to parse metadata is separate from code to parse fulltext, for example, even though they parse the same input XML

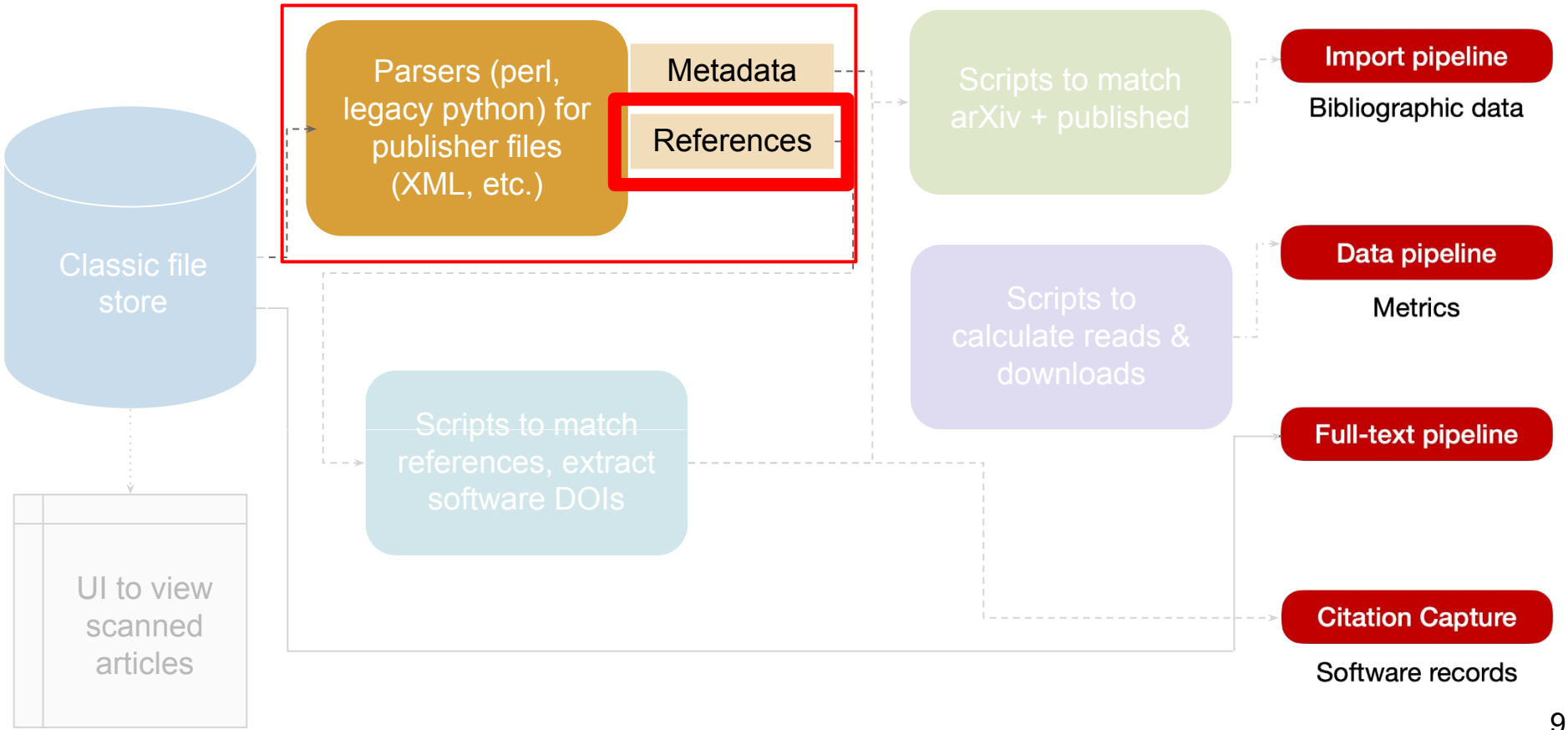
New parsers

- New parsing library is unified in style, syntax, input, and output (the new data model)
- Parsers completed and in testing
  - JATS, arXiv, Elsevier, DataCite, CrossRef, Wiley
  - These are ~¼ of total parsers, but cover ~⅔ of records ingested

Next steps

- Finish remaining parsers
  - 10 more in Perl, plus several more in legacy Python
- Start using new parsers in production
  - DataCite parser: ingestion script using it now, will be added to Citation Capture pipeline soon
  - Harvester pipeline, in progress, will be needed to fully make use of these, likely starting with one publisher at a time
- Long term goals
  - Port parsing code from other pipelines/libraries to the parsing library, such as fulltext parsing, and citation context for machine learning efforts

Reference Extraction



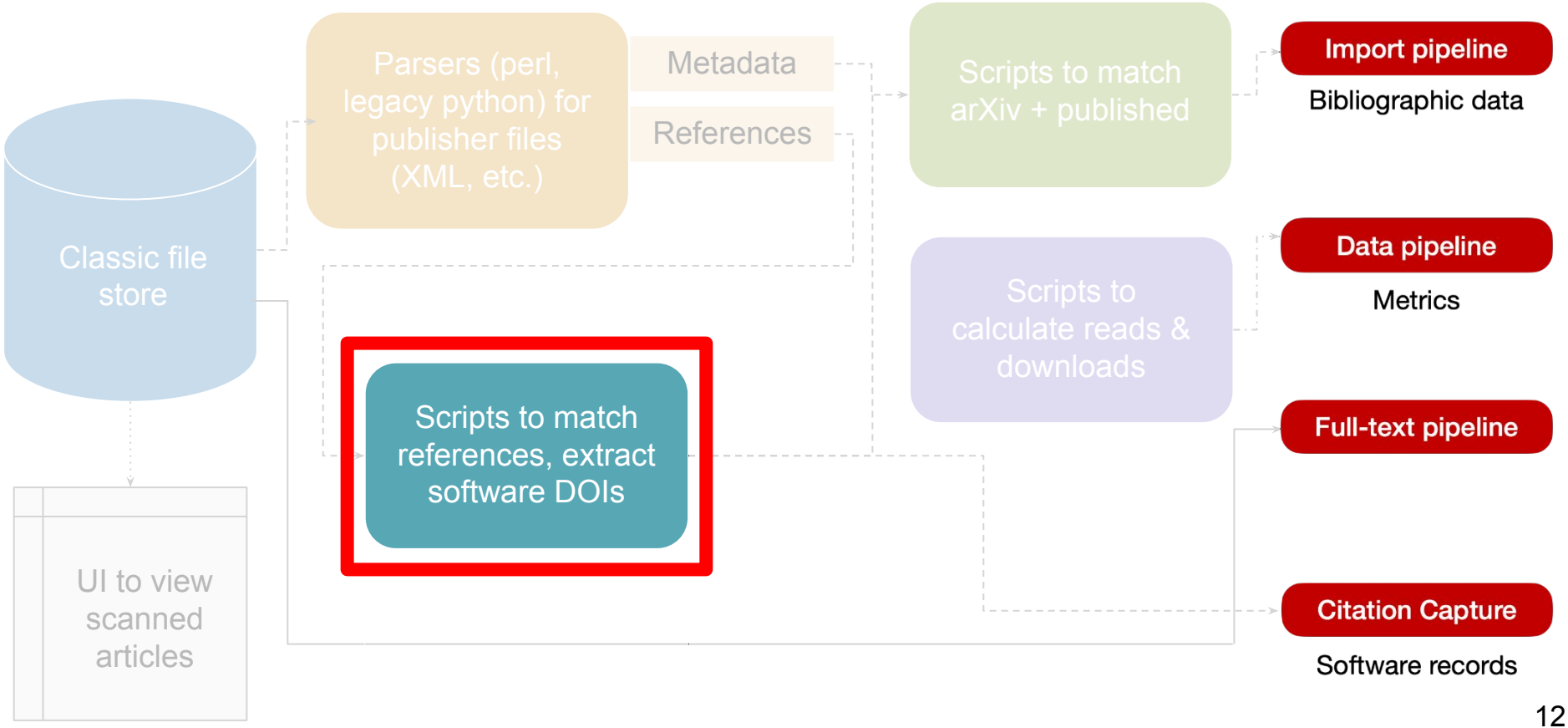
Reference Extraction: Context

- The ADS citation network doesn't build itself
  - Many data sources do not provide structured reference information
  - Notably ArXiv; a typical day adds tens of thousands of references
    - Ex: 2021-11-07 had 31,895
  - Fulltext ⇒ "refstrings" ⇒ bibcodes
  - Fulltext is generally PDF, but ArXiv has lots of TeX
    - 2021-11-07: 821 TeX, 122 PDF, 8 withdrawn, 12 fail processing
- Good reasons to update the current pipeline
  - A "classic" Perl framework
  - Very good at ArXiv TeX, but some known limitations (e.g. Unicode)
  - PDF extractor also good, but lots of new ML tools to bring to bear
  - PDF extraction *could* do lots more than references (abstracts, figures, ...)

Reference Extraction: Status

- ArXiv TeX extraction has been updated
  - Upgraded to ArXiv's current TeX install
  - New Python implementation in modernized Docker framework
  - Robustness improvements for corner cases
  - Doing any better probably requires much more work (*real* TeX parsing)
- Modernized PDF extraction is nearing deployment
  - Industry-standard tool GROBID offers solid improvements
  - Other options investigated, not competitive
- Exploring future directions for generalized PDF extraction
  - Created ADS-tuned training set for testing/ranking approaches
  - GROBID's models can be customized for ADS content
  - New tools are becoming available (e.g. ScienceParsePlus)
  - Likely makes more sense to adopt/collaborate than build from scratch

Reference Resolver



Reference Resolver: Service & Pipeline

➢ Presented 2 years ago at ADSUG 2020 (PDF)

- Service (completed):
  - Reference Resolving: find record that matches a reference string input and outputs the bibcode and a computed confidence score
- Pipeline (future development):
  - Framework that will make use of Reference Service
    - Input new document's bibliography and outputs matching documents as the linked references in ADS (citation graph)
- Goals for use:
  - Replacing classic machinery
  - Content & curation support

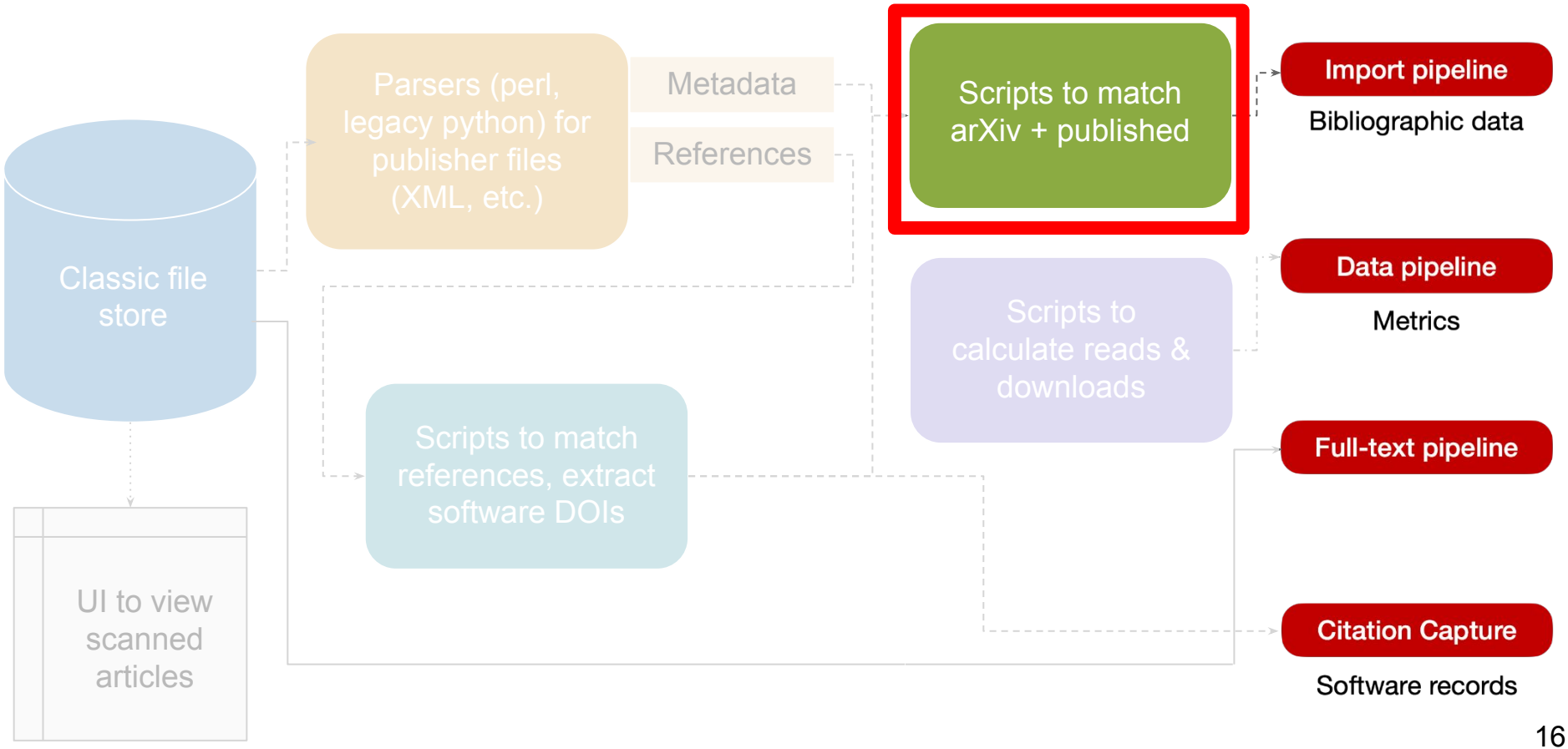
Reference Service for Content & Curation

- Identifying Coverage Gaps
  - ARC/Space Science & Astrobiology Division - Blog post (Nov 2021) details project
    - Matching ARC/SS Division bibliography with ADS content for coverage of NASA papers
    - ADS Ref Service API - match by reference strings (Author + Year + Publication)
      - This found the majority of results
    - ADS search API - match additional by DOI or Title
  - HOLLIS Harvester
    - Project detailed in Jenny's GitHub HOLLIS Harvester documentation
      - Matching HOLLIS monographs with ADS content for coverage of gray literature
    - Searched Ref Service API with reference strings (Author + Title + Pub Year)
    - Match existing items; reviewed unmatched/new items for curation and ingest

Reference Resolver: Development

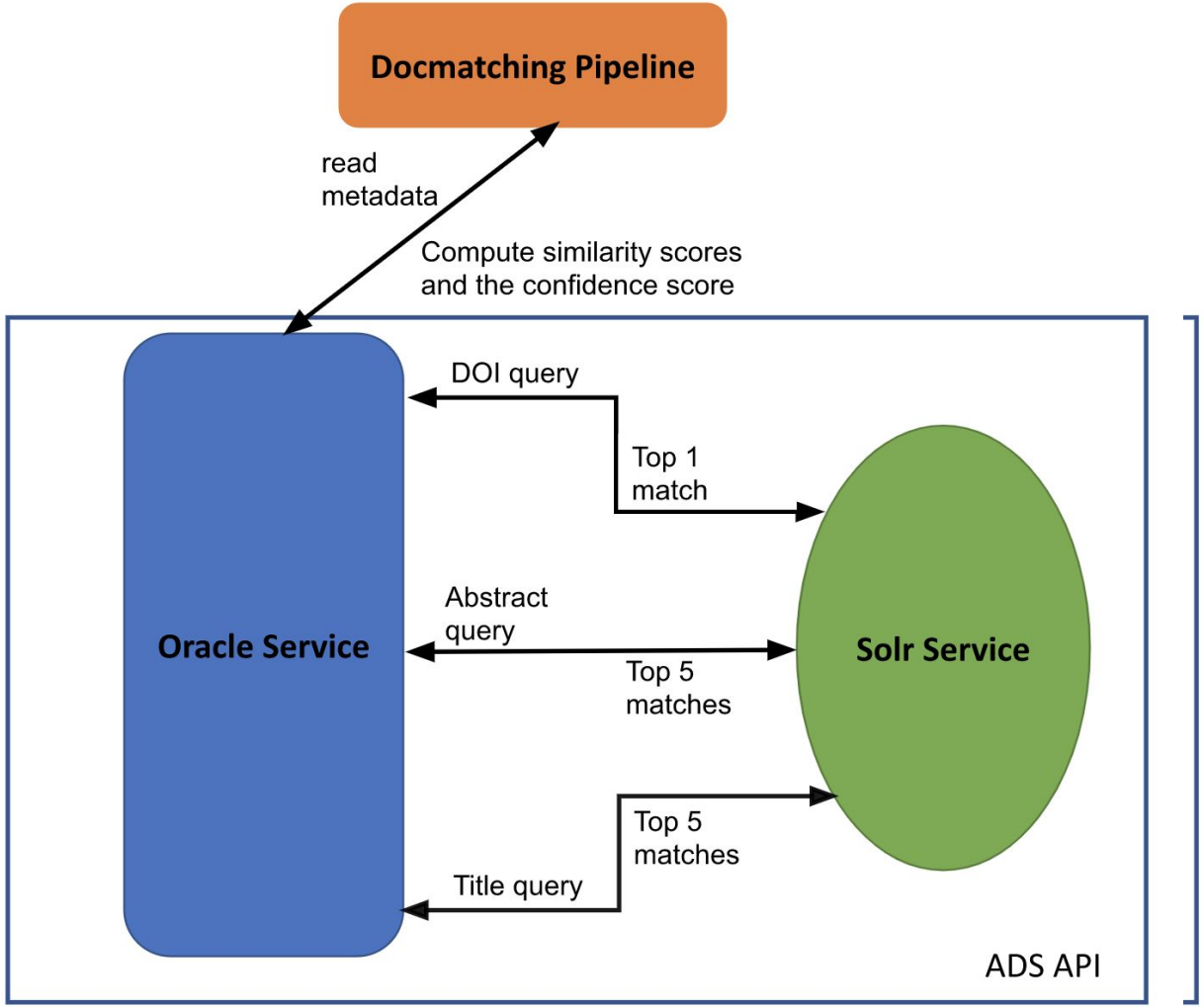
- Reference service is complete and its accuracy exceeds ADS Classic's resolver
- Future work:
  - Parsers for some smaller publishers' references (e.g. conferences, etc.)
  - Integrate into pipeline infrastructure

Docmatcher


Parsers (perl, legacy python) for publisher files (XML, etc.)
Metadata
References
Scripts to match arXiv + published
Import pipeline
Bibliographic data
Classic file store
Scripts to calculate reads & downloads
Data pipeline
Metrics
Scripts to match references, extract software DOIs
Full-text pipeline
UI to view scanned articles
Citation Capture
Software records

ArXiv matching with new docmatcher

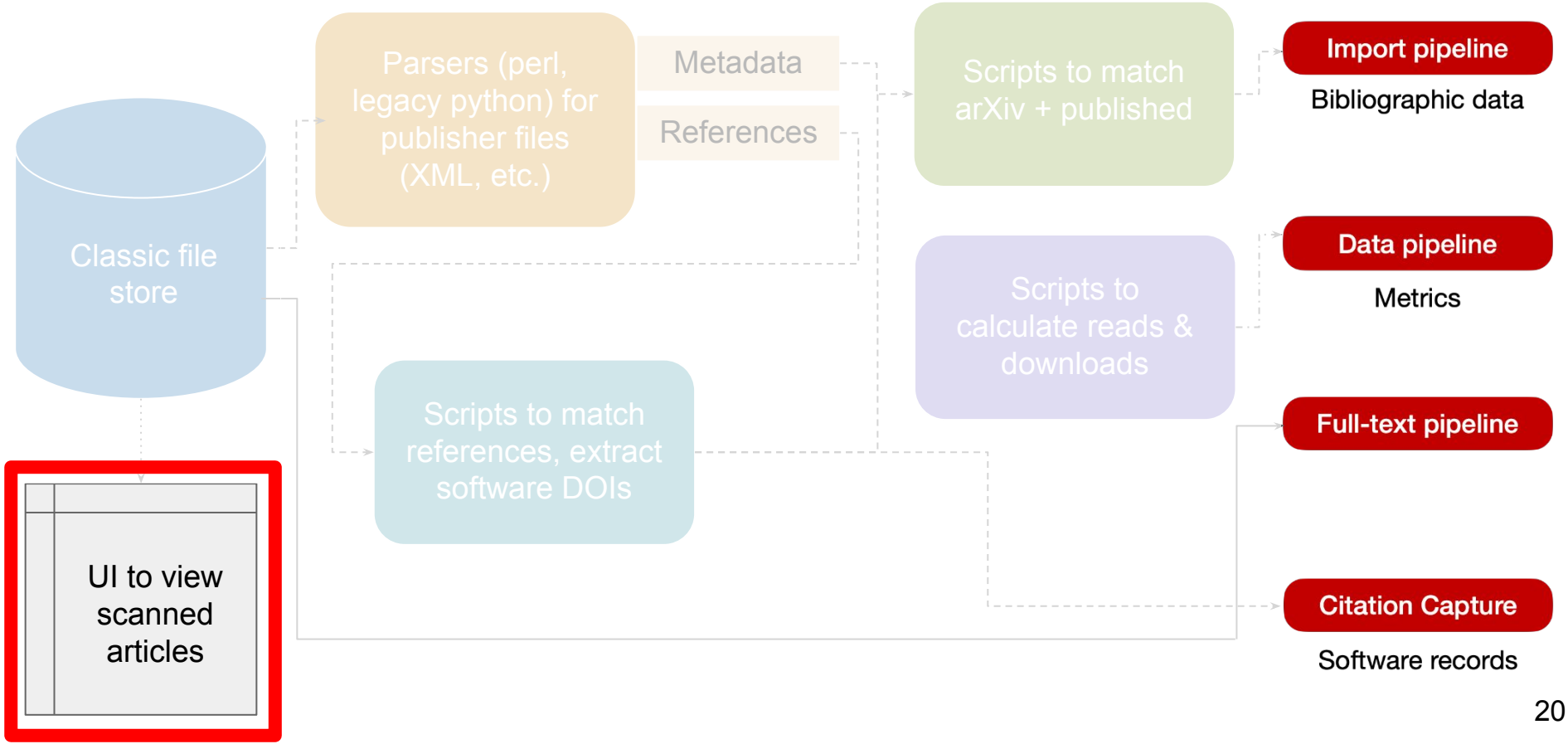
- In production as of October, 2022
  - Running in parallel with classic matching
  - 116,500 arXiv in October, 2022
  - Both classic and arxiv matched on the order of 8% (slightly more with docmatcher) but docmatcher's are more accurate
  - 20% checked to help determine threshold for accepting
  - Correct matches at > 97%
  - Blog post September, 2022


Docmatching Pipeline
read
metadata
Compute similarity scores
and the confidence score
DOI query
Top 1
match
Abstract
query
Top 5
matches
Top 5
matches
Title query
Oracle Service
Solr Service
ADS API

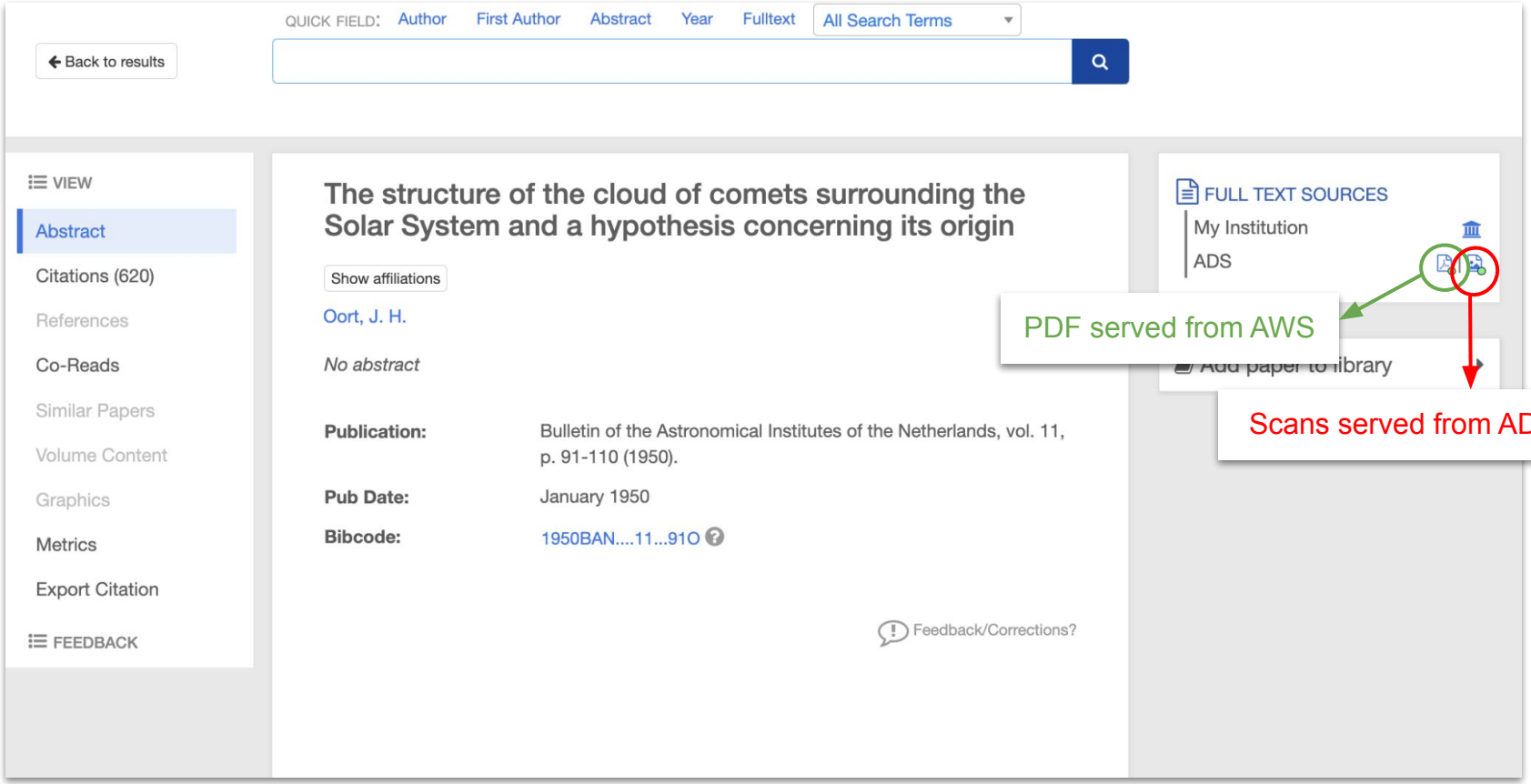
ArXiv matching with new docmatcher

- Next Steps
  - Automate curated matches back into system
  - Remove classic matching from indexing processes
    - Will speed up classic indexing (probably by a factor of 2)
  - Turn off classic matching December, 2022
    - Will reduce number of user submissions
    - Will reduce number of user corrections

Scan Explorer



ADS Digitization Efforts

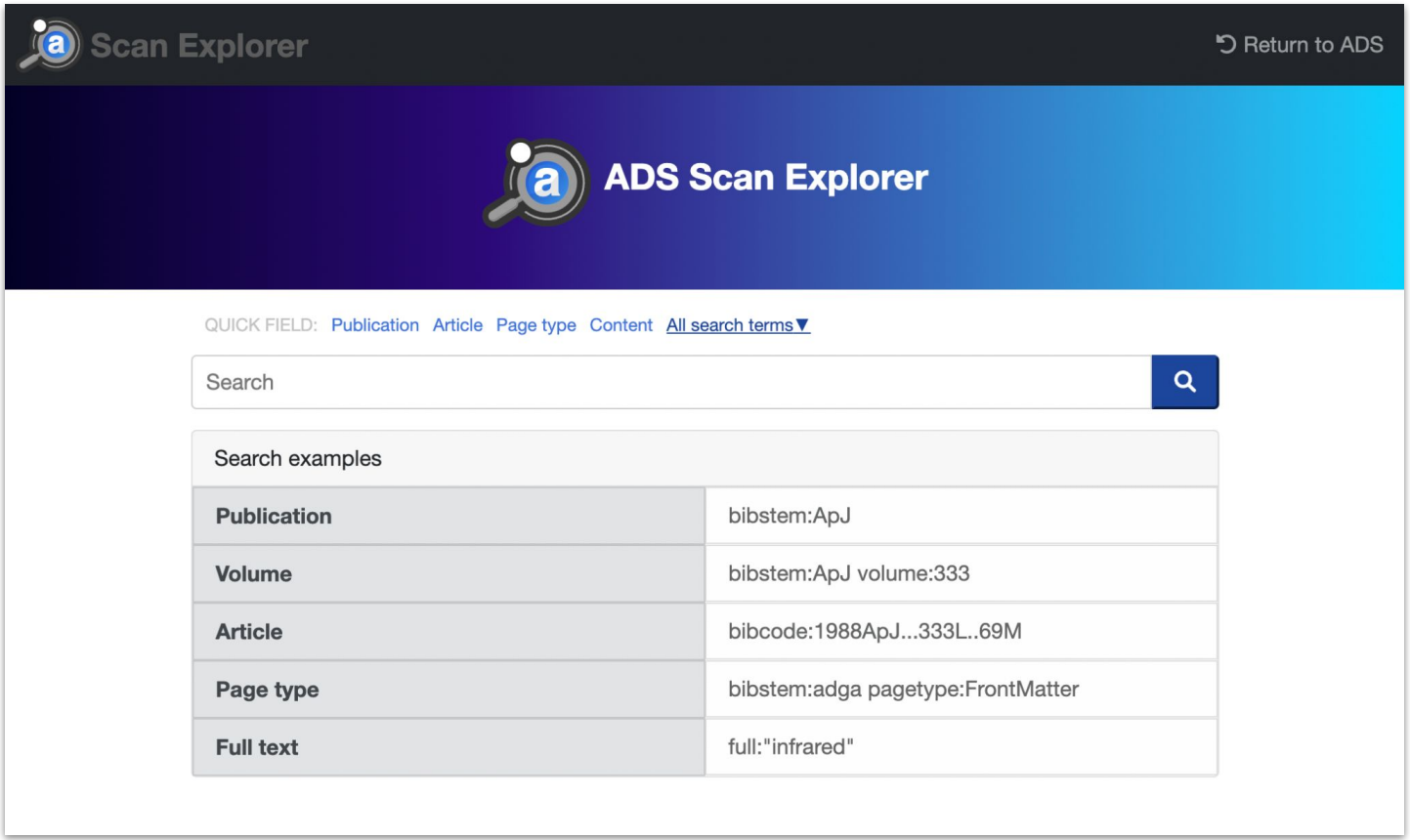


ADS Scan Explorer

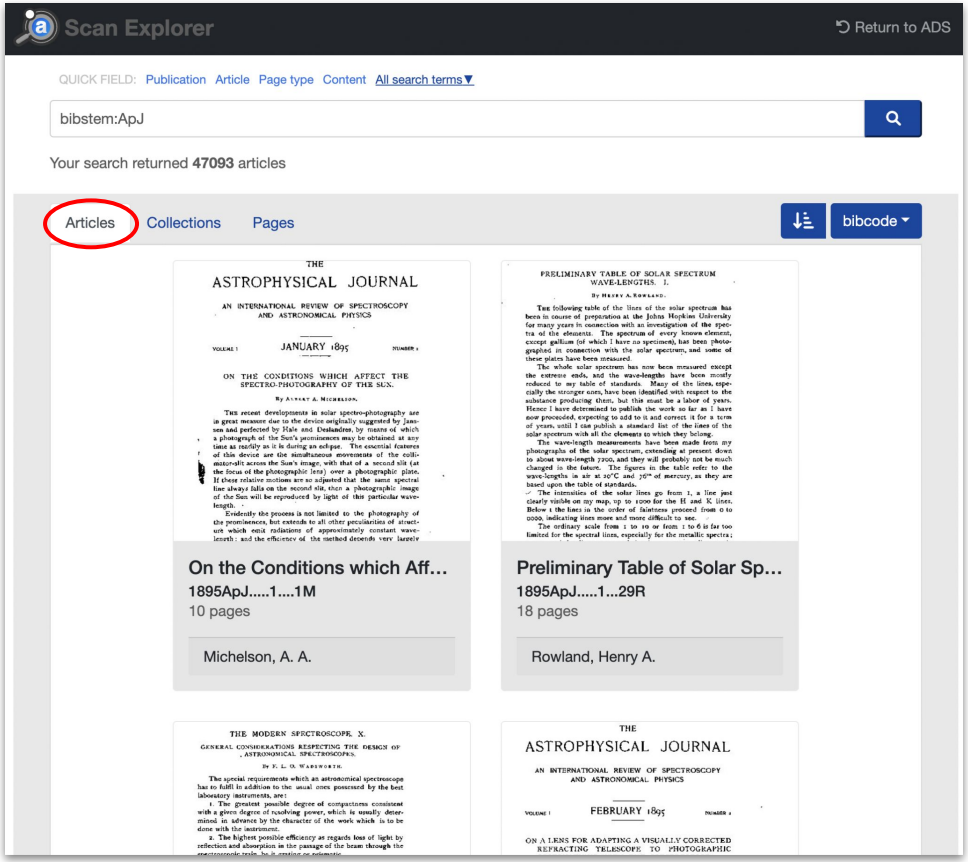
- Deployment:
  - The different components of the ADS Scan Explorer should be easily deployable as Docker containers.
- Pipeline/Provisioning:
  - The ADS Scan Explorer is required to have infrastructure that will allow it to be provisioned with new data and which will allow standard data operations on existing content.
- API/Image Server:
  - API architecture for serving images based on IIIF standards
- User Interface/ADS-branded viewer:
  - Image viewer compatible with IIIF standards and current ADS UI software (Mirador, https://projectmirador.org/)

Implementation: outsourced to Winter Way based on a SOW written by the ADS

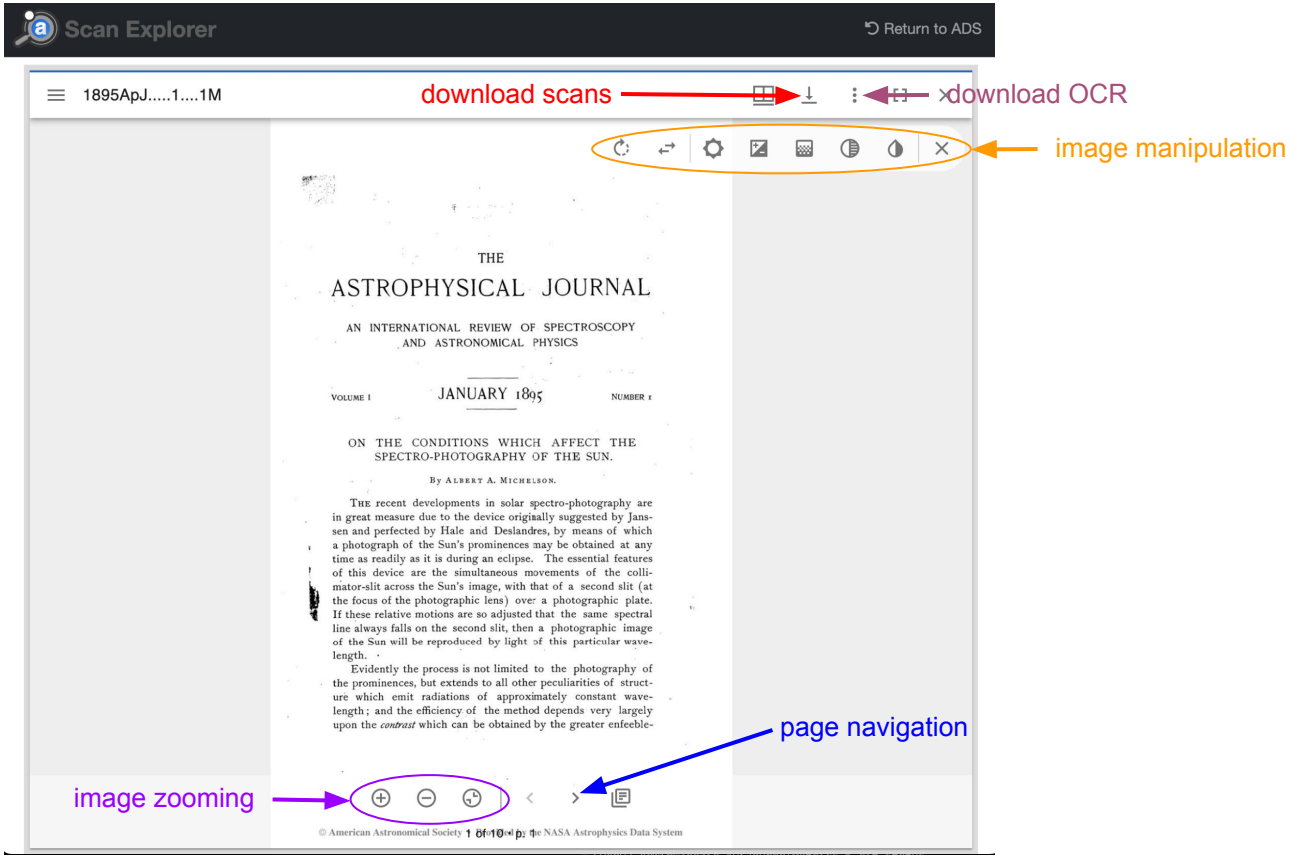
ADS Scan Explorer (https://dev.adsabs.harvard.edu/scan)



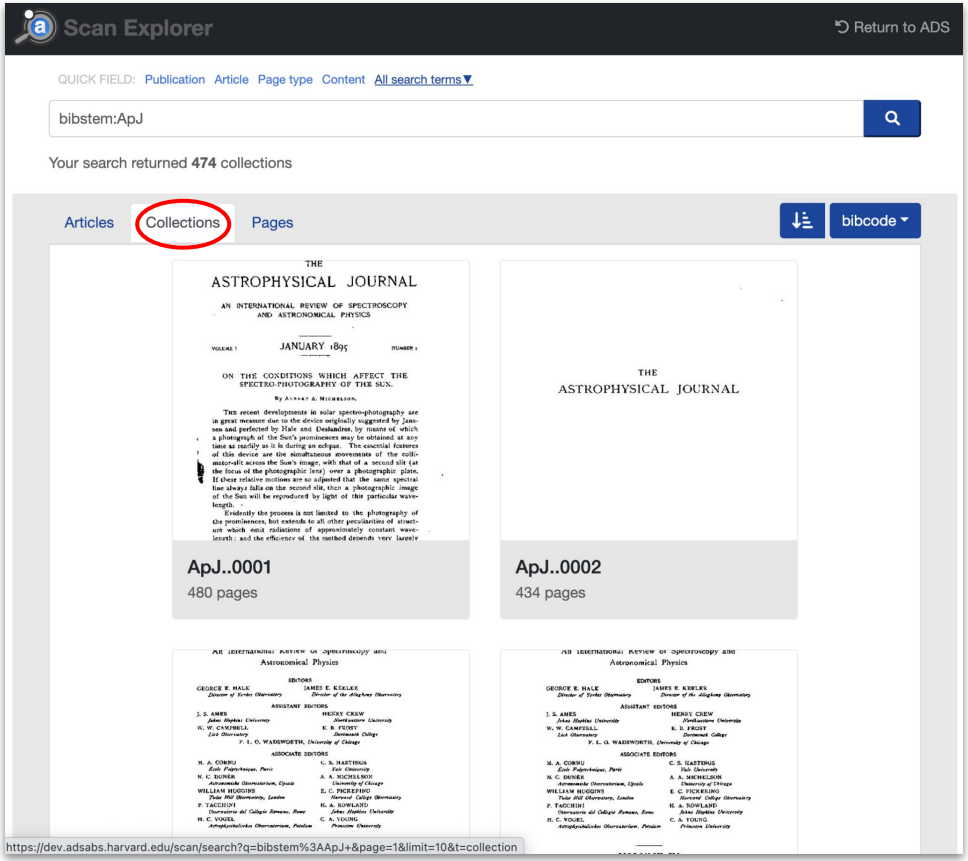
ADS Scan Explorer - Article View



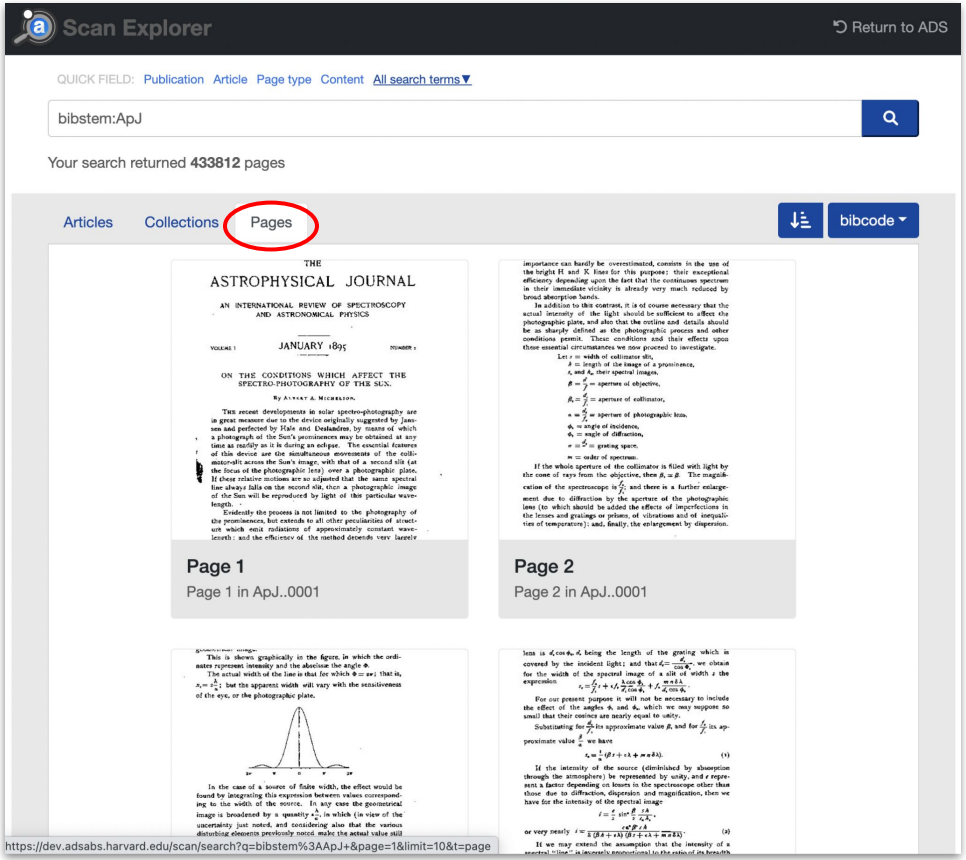
ADS Scan Explorer - Article View



ADS Scan Explorer - Collections View



ADS Scan Explorer - Pages View



Final thoughts

- ADS Scan Explorer functionality
  - Supports all functionality of Classic interface
  - Additional functionality (e.g. search & download OCR text)
  - Mirador image viewer supports plugins (e.g. annotation)
- Goal: in production by end of calendar year
- Outsourcing success story