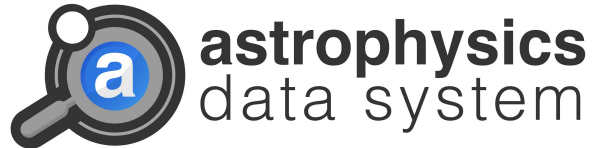# Platform Transition:
Removing "beta" from Bumblebee

Alberto Accomazzi & the ADS Team
aaccomazzi@cfa.harvard.edu
@aaccomazzi

ADS Users Group Meeting - 11/2/2017

Panel Recommendations

The ADS Users Group recommends that ADS proceed with an expedited move to the ADS Bumblebee from the classic platform, and discontinue support for the classic platform. Such a transition should be fairly smooth since Bumblebee has available tabs for both the "Classic Form" and the "Modern Form." [...]

We recommend that the ADS group prioritize the launch of Bumblebee with its classic functionality and engage the public to use it. Additional features (e.g., user's profiles) can be deferred.

We recommend that a schedule be created for this very much needed transition, especially given the short-staffing situation on ADS (see below). We acknowledge that a robust restructuring of infrastructure may be needed to accommodate dramatically larger numbers of users on Bumblebee. We note that the discontinuation of ADS Classic will significantly free up resources to enable further development

Rollout Plan

- Build out Bumblebee to achieve feature parity with ADS Classic
  (add features to search interface and corresponding microservices)
- Achieve content parity with ADS Classic
  (improve data migration pipeline to new data store)
- Provide a more reliable user experience
  (minimize usability problems and search errors)
- Increase system capacity to match expected load from ADS Classic users
  (2 orders of magnitude increase)
- Manage transition plan and expectations
  (be prepared for questions / complaints / unforeseen problems)

Feature Parity - What

- ADS Classic and Bumblebee use very different search engines, which expose different APIs and features, so 100% parity is neither possible nor desirable
- Therefore, parity should be evaluated based on required functionality and priority given to the most frequently used features
- Usage data and direct user feedback have guided us in prioritizing the implementation of functionality

Feature Parity - Why

- We don't want users to go back to their old habits because some feature is missing from Bumblebee
- We want to have a clean "cutover" so that we stop relying on Classic even as a back-end service (e.g. for exporting records)
- Some rarely used features in the Classic search engine are needed for layered services (e.g. myADS notifications)

FILTERS

Select References From:
- All bibliographic sources   Select only articles
- All refereed articles
- All non-refereed publications

Select/deselect publications: [_____] (',' separated list)

Select References With:
- A bibliographic entry
- At least one of the following (OR):
- All of the following (AND):
- None of the following (NOT):

Abstracts	Data Links	
Full Text Articles	Scanned Articles	Electronic Articles
arXiv e-print	Table of Contents	Mail Order Links
References	Citations	Other related articles
SIMBAD Objects	NED Objects	PDS Information
Author Comments	Library Links	Also-read
Multimedia	HEP/SPIRES Links	

Select References In:
- All Groups
- At least one of the following groups (OR):
- All of the following groups (AND):

ARI	CfA	CFHT	Chandra
ESO/Lib	ESO/Telescopes	Gemini	Herschel
HST	ISO	IUE	JCMT
Keck	Leiden	LPI	Magellan
NOAO	NRAO	NRAO/Telescopes	ROSAT
SDO	SMA	Spitzer	Subaru
Swift	UKIRT	USNO	VSGC
XMM			

Entry Date:
Since: [__] [__] [__]   Before: [__] [__] [__]
Day(DD) Month(MM) Year(YYYY)   Day(DD) Month(MM) Year(YYYY)

Min Score: [__]

[Send Query] [Return Query Form] [Store Default Form] [Clear]

Example: Abstract Search Feature Parity

Search fields
- Authors
- Objects (SIMBAD, NED, ADS)
- Title words
- Abstract words

Filters / modifiers
- Database
- Publication Date
- Number of items
- Refereed
- Article only
- Specific journals
- "References with" (property selection)
- "References in" (bibliographic group selection)
- Entry date
- Minimum score

Formatting
- All formats natively supported by BBB (Note: these are only available from the export service rather than search engine)

Sorting
- Hybrid Score (technically implemented but not yet exposed)
- Normalized score (not available as native sort)
- Citation count
- Normalized citation count (citation_count / author_count)
- First author name
- Number of authors
- Publication Date
- Entry date
- Page (~ bibcode)

Settings
- Require field for selection: combination of "and" across fields or implemented via "+" syntax per field
- Synonym replacement: "done" for most fields
- Relative weights: not being implemented
- Use for Weighting: same for this one
- Weighted scoring: also let go

Key: done, in progress, todo, not being implemented

Content Parity - What

- Bumblebee currently depends 100% on ADS Classic for data ingest
- A Data migration pipeline takes content from ADS classic and pushes it to new system
- Occasional ingest problems do occur due to the new system's stricter requirements in metadata encoding and schema
- Currently working on improving error detection and recovery so we can react to them quickly

Content Parity - Why

- We definitely don't want to give the impression that Bumblebee is lacking content wrt Classic
- For weekly updates, a few missed records can be fixed, reingested over the next couple of days
- For daily arXiv updates, slow ingest becomes a blocker to achieving parity
- Just now we are starting to work on simple metadata ingest for daily updates which is Classic-independent

User Experience

- Make the Bumblebee application more robust
- Provide on-line documentation which includes FAQ, quickstart, transition questions
- Minimize errors from API services and the SOLR search engine
- When things go wrong, provide reasonable feedback to the user
- Perform local user testing before launch to help iron out problems

Execution

- Critical requirements must be satisfied before "Beta" removed from Bumblebee
  - All readiness criteria met: content & feature parity, error-free experience
  - User testing indicates no major problems
  - Scalability issues fully solved and system ready for on-demand growth
- Messaging
  - Communicate to users the need for transition to ADS Classic its advantages
  - Prepare mitigation plan and support material
- Plan timeline
  - Achieve Bumblebee readiness at time T, freeze code
  - Advertise locally and conduct user testing on sample population at T + 2 weeks
  - Work on bug-fixes, evaluate user feedback and fix major issues
  - Deprecate use of ADS Classic to the general public at T + 3 months
  - Discontinue use of the ADS Classic search at T + 15 months

Where we are

- Working as fast as we can on multiple fronts
  - Implementation of missing system functionality in new microservices (e.g. link redirection, native export, author/affiliation forms, NED object search)
  - UI changes to include additional functionality (e.g. custom export, linking)
  - API architecture being upgraded to avoid congestion problems
  - Ingest pipeline upgraded and being tested as we speak
- Bumblebee Readiness not yet reached
  - System components still being developed and tested
  - Known issues with search still to be tackled
  - Some UI changes still in progress
- Currently working on reaching readiness on Jan 1st, 2018
  - Note: this is a tight schedule with little room for contingencies
  - ADS Classic deprecation expected in early Q2 2018