Bullet Cluster Data Acquisition Manifest for Pixel-Level FUM/UM + BCR Witness Test 1E0657-56 / 1E0657-558

Within the First Utterance Model / Universal Mechanics (UM/FUM) framework
Locked 2026-05-14

PATENT PENDING — USPTO Application No. 19/640,364
This manifest organizes public-domain astronomical archive data for application of the locked
three-layer $K_{act,observed}$ composite law and the chunked-data pixel-κ algorithm.

Purpose. The pixel-level execution of $\kappa_{pred}(x,y) = K_{act,observed}(x,y) \cdot \kappa_{baryonic}(x,y)$ for the Bullet Cluster requires five categories of input data: (1) observed lensing convergence $\kappa_{observed}(x,y)$ maps; (2) baryonic surface mass density $\Sigma_{baryonic}(x,y)$ components (X-ray plasma + stellar mass + ICL); (3) lensing geometry parameters (z_L, z_S, D_L, D_S, D_{LS}); (4) WCS pixel registration headers; (5) noise / uncertainty maps for residual significance. This manifest provides for each category: the canonical published archive DOI or URL, the specific observation IDs, the access procedure, and (where retrieval was possible in-session) the file delivered into this folder. The SL multiple-image catalog from Cha et al. 2025 is already retrieved into this folder as Exhibit 1.

§1. Locked Framework Input Equations (recap)

```
K_act,internal(g_bar) = max(1, sqrt(g_critical / g_bar))
g_critical = c * H0 / (2 * omega_C1) = 1.042e-10 m/s^2
Lambda(g_bar) = 1 / [1 + (K_act,internal(g_bar) - 1) * 3 / (4 * phi^3 * Eidolon)]
K_act,observed(x,y) = K_act,internal(Lambda(x,y)) * F_env(eps_ext, eta_tid) * F_ML(Del
kappa_baryonic(x,y) = Sigma_baryonic(x,y) / Sigma_crit_lens
kappa_pred(x,y) = K_act,observed(x,y) * kappa_baryonic(x,y)
Sigma_crit_lens = c^2 * D_S / (4 * pi * G * D_L * D_LS)
g_bar(x,y) ~= 2 * pi * G * Sigma_baryonic(x,y)
```

§2. Dataset 1 — Strong Lensing Multiple-Image Catalog (RETRIEVED)

> **STATUS: INCLUDED IN THIS PACKAGE** — File `SL_multiple_image_catalog.txt` (9,003 bytes; 217 image rows; 9 columns).

Field	Value
Source	Cha, S. et al. (2025). ApJ 987 L15 — JWST high-caliber Bullet Cluster lensing analysis
Zenodo DOI	`10.5281/zenodo.15208501`
Direct download URL	`https://zenodo.org/records/15208501/files/SL_multiple_image_catalog.txt?download=1`
License	CC-BY 4.0
Citation	Cha, Sangjun, et al. (2025). Zenodo. https://doi.org/10.5281/zenodo.15208501

File format: tab-separated text with columns `ID`, `RA(ICRS)`, `DEC(ICRS)`, `spec_z`, `model_z`, `photz_50`, `photz_16`, `photz_84`. Provides 146 strong lensing constraints from 37 systems used in the Cha 2025 reconstruction.

§3. Dataset 2 — Observed Lensing Convergence κ(x,y) (ARCHIVE ACCESS REQUIRED)

> **STATUS: PUBLIC ARCHIVE — astroquery / MAST Portal retrieval required.**

3.1 Primary Source — Cha et al. 2025 JWST + HST Reconstruction

Field	Value
Title	JWST and HST images of the Bullet Cluster (Cha et al. 2025)
MAST DOI	`10.17909/8zea-jv19`
JWST Program ID	GO-4598 (PI: Maruša Bradač)
HST filters used in reconstruction	F435W, F606W, F775W, F814W, F850LP

JWST instrument	NIRCam
MAST Portal Search URL	`https://mast.stsci.edu/portal/Mashup/Clients/Mast/Portal.html?searchQuery={"service":"DOIOBS","inputText":"10.17909/8zea-jv19"}`

3.2 astroquery Retrieval Recipe

```
from astroquery.mast import Observations

# Resolve the DOI to observation set
obs_table = Observations.query_criteria(
    obs_collection=['JWST', 'HST'],
    proposal_id=['4598'],          # JWST GO-4598
    target_name='*Bullet*'
)

# Get product list (includes mass reconstruction maps when archived as HLSP)
products = Observations.get_product_list(obs_table)

# Filter for science products and FITS images
science = Observations.filter_products(
    products,
    productType=['SCIENCE'],
    extension='fits'
)

# Download to local directory
Observations.download_products(
    science,
    download_dir='./Bullet_Cluster_Data/JWST_HST_GO4598/'
)
```

3.3 Legacy Reference — Bradač et al. 2006 / 2009 SL+WL Reconstruction

Field	Value
Publication	Bradač, M. et al. (2006). ApJ 652, 937 — strong+weak lensing unified mass reconstruction
ADS	`https://ui.adsabs.harvard.edu/abs/2006ApJ...652..937B/abstract`
Mass estimate (main cluster)	$M(<250\ \mathrm{kpc}) = 2.8 \pm 0.2 \times 10^{14}\ M_\odot$
Mass estimate (subcluster)	$M(<250\ \mathrm{kpc}) = 2.3 \pm 0.2 \times 10^{14}\ M_\odot$

Data products	FITS mass maps; not archived under a single DOI — contact author for FITS distribution, or use the Cha 2025 superseded reconstruction.

3.4 Clowe et al. 2006 Original Maps

Field	Value
Publication	Clowe, D. et al. (2006). ApJ 648, L109 — "A Direct Empirical Proof of the Existence of Dark Matter"
ADS	`https://ui.adsabs.harvard.edu/abs/2006ApJ...648L.109C/abstract`
Data release	2006 November 15 release: X-ray surface density Σ-map + SL+WL convergence κ-map
Modern preferred substitute	Cha et al. 2025 (DOI 10.17909/8zea-jv19) — higher resolution, JWST-constrained

§4. Dataset 3 — Baryonic Component A: X-Ray Plasma (Chandra)

> **STATUS: PUBLIC ARCHIVE — Chandra Data Collection DOI resolved.**

4.1 Canonical Chandra Data Collection

Field	Value
Chandra DOI	`10.25574/cdc.373`
Resolves to ChaSeR query	`https://cda.cfa.harvard.edu/chaser/?obsid=3184,4984,4985,4986,5355,5356,5357,5358,5361`
Instrument	ACIS-I (imaging mode)
Total exposure	~500 ks (Markevitch et al. 2006)
Target	1E0657-56 / 1E0657-558

4.2 ObsID List (9 observations comprising the canonical 500 ks dataset)

ObsID	Provenance
3184	Initial discovery-class observation (Markevitch et al. 2004)
4984, 4985, 4986	Deep ACIS-I follow-up series

5355, 5356, 5357, 5358	Extended ACIS-I integration
5361	Final ACIS-I segment

4.3 Retrieval Procedure

Chandra public data requires the CIAO software toolkit (Chandra Interactive Analysis of Observations) for reduction. Public reprocessed level-2 event files are available via ChaSeR.

```
# Via CIAO (preferred for proper exposure-corrected images)
# Install CIAO: https://cxc.cfa.harvard.edu/ciao/

download_chandra_obsid 3184,4984,4985,4986,5355,5356,5357,5358,5361

# Combined exposure-corrected image (per ObsID, then merge):
chandra_repro indir=obs_3184 outdir=obs_3184_reprocessed
fluximage obs_3184_reprocessed/acisf03184_repro_evt2.fits bin=2 \
    bands=csc psfecf=0.9 outroot=obs_3184

# Merge all ObsIDs to single exposure-corrected map (the canonical Σ_X-ray):
merge_obs "obs_*_repro_evt2.fits" outroot=bullet_merged bands=csc \
    psfecf=0.9 binsize=2
```

4.4 Direct Browser-Based Retrieval (no CIAO)

From the ChaSeR query URL above, select each ObsID and request "Retrieve By ObsID" → primary + secondary tarballs. Each tarball is typically 100-500 MB; level-2 event files (.evt2.fits) and exposure maps (.expmap) are sufficient inputs for surface-brightness reconstruction.

§5. Dataset 4 — Baryonic Component B: Galaxy Stellar Mass (HST/JWST/Magellan)

> **STATUS: PUBLIC ARCHIVE — same MAST DOI as §3.1.**

Field	Value
Primary Source	JWST NIRCam imaging from GO-4598 + ancillary HST ACS imaging
MAST DOI	`10.17909/8zea-jv19` (same collection as §3.1)
JWST NIRCam filters	F090W, F115W, F150W, F200W, F277W (ICL filter), F356W, F410M, F444W

HST filters	F435W, F606W, F775W, F814W, F850LP
Stellar mass derivation	SED fitting using SE++/EAZY/Prospector pipeline on multi-band photometry

§6. Dataset 5 — Baryonic Component C: JWST ICL (Cha et al. 2025)

> **STATUS: PUBLIC ARCHIVE — same MAST DOI as §3.1.**

Field	Value
Primary Source	JWST NIRCam F277W mosaic; Cha 2025 §2.5
MAST DOI	`10.17909/8zea-jv19`
ICL extraction filter	F277W (rest-frame near-IR, optimal for ICL)
Hausdorff distance (ICL vs mass)	19.80 ± 12.46 kpc (the witness metric Alfred has been quoting)

The ICL map is extracted from the JWST NIRCam F277W mosaic with galaxy segmentation masks applied. The same MAST collection delivers raw mosaics; ICL-specific products may be published as a High-Level Science Product (HLSP) — search MAST HLSPs for "bullet cluster ICL".

§7. Dataset 6 — Lensing Geometry (CALCULATED)

> **STATUS: COMPUTED FROM PUBLISHED REDSHIFTS — no archive retrieval required.**

Field	Value
z_L (cluster redshift)	0.296 (1E0657-56 cluster center)
z_S (source redshift distribution)	From Cha 2025 catalog above; spectroscopic + model + photometric z available per source
Cosmology	Standard Planck 2018: H0 = 67.4 km/s/Mpc, Ω_m = 0.315, Ω_Λ = 0.685 (substrate-clean per Paper 2)

```
from astropy.cosmology import Planck18
import astropy.units as u
```

```
import numpy as np

cosmo = Planck18  # or FlatLambdaCDM(H0=67.4, Om0=0.315) for substrate-clean
z_L = 0.296
z_S = 2.0  # typical lensed-source redshift; use catalog values per source

D_L  = cosmo.angular_diameter_distance(z_L)
D_S  = cosmo.angular_diameter_distance(z_S)
D_LS = cosmo.angular_diameter_distance_z1z2(z_L, z_S)

c = 2.998e8 * u.m / u.s
G = 6.674e-11 * u.m**3 / (u.kg * u.s**2)

Sigma_crit_lens = (c**2 * D_S / (4 * np.pi * G * D_L * D_LS)).to(u.kg / u.m**2)
```

§8. Dataset 7 — WCS Pixel Registration

> **STATUS: EMBEDDED IN FITS HEADERS** — automatically present in any FITS file from §3-§6.

Required FITS header keywords:

- `CRPIX1`, `CRPIX2` — reference pixel coordinates
- `CRVAL1`, `CRVAL2` — reference world coordinates (RA, Dec)
- `CDELT1`, `CDELT2` — pixel scales (or CD_{ij} matrix)
- `CTYPE1`, `CTYPE2` — coordinate types ("RA---TAN", "DEC--TAN" or similar)
- `WCSAXES` — number of WCS axes

```
from astropy.wcs import WCS
from astropy.io import fits

# Load a FITS map and extract its WCS
with fits.open('bullet_cluster_kappa.fits') as hdul:
    wcs = WCS(hdul[0].header)
    data = hdul[0].data

# Reproject another map onto the same WCS grid for pixel-by-pixel comparison:
from reproject import reproject_interp
data2_aligned, footprint = reproject_interp(
    ('chandra_xray_sb.fits', 0),  # source FITS
    wcs,                          # target WCS
    shape_out=data.shape
)
```

§9. Dataset 8 — Noise / Uncertainty Maps

STATUS: PUBLIC ARCHIVE — packaged with primary FITS products.

Per-instrument noise maps:

- JWST NIRCam: `*_rate.fits` ERR extension provides per-pixel uncertainty; `*_cal.fits` ERR/WHT extensions for calibrated products. Available via MAST same DOI.
- HST ACS: `*_drz.fits` WHT extension (inverse variance); ERR extension.
- Chandra: exposure maps from `fluximage` output. Photon-noise computed as √counts per pixel from binned image.
- Lensing reconstruction: bootstrap or Markov chain uncertainty published by reconstruction team (Cha 2025 supplementary).

§10. Executable Pixel-κ Pipeline (Reference)

```
import numpy as np
from astropy.io import fits
from astropy.wcs import WCS

# === Locked UM-native framework constants ===
PHI = 1.6180339887498949
ALPHA_STRUCT = 0.0073032157
EIDOLON = (1 - ALPHA_STRUCT) / ALPHA_STRUCT
K_STRUCT = 4 * PHI**3 * EIDOLON / 3
G_CRITICAL = 1.042e-10
G_NEWTON = 6.674e-11

# === Lensing geometry (computed once per dataset) ===
SIGMA_CRIT_LENS = compute_sigma_crit(z_L=0.296, z_S_dist=Cha2025_catalog)

# === Cluster-locus environmental parameters ===
EPS_EXT = 0.0     # cluster core; no external host
ETA_TID = 0.0     # cluster-scale collisionless is in equilibrium
DELTA_ML = 0.05   # cluster baryonic M/L well constrained
F_ENV = 1.0 / ((1.0 + EPS_EXT) * (1.0 + ETA_TID**2))
F_ML_INV = 1.0 / (1.0 + DELTA_ML)

def process_chunk(sigma_baryonic_chunk):
    g_bar = 2 * np.pi * G_NEWTON * sigma_baryonic_chunk
    safe_g_bar = np.where(g_bar > 0, g_bar, 1e-30)
    K_int = np.maximum(1.0, np.sqrt(G_CRITICAL / safe_g_bar))
    K_obs = K_int * F_ENV * F_ML_INV
    kappa_bar = sigma_baryonic_chunk / SIGMA_CRIT_LENS
    kappa_pred = K_obs * kappa_bar
```

```
    return kappa_pred, K_obs, kappa_bar

# === Memory-bounded chunked execution ===
CHUNK = 256

with fits.open('sigma_baryonic.fits', memmap=True) as hdul:
    sigma = hdul[0].data
    wcs_baryonic = WCS(hdul[0].header)

ny, nx = sigma.shape
kappa_pred_full = np.zeros((ny, nx), dtype=np.float64)
K_obs_full = np.zeros((ny, nx), dtype=np.float64)

for i0 in range(0, ny, CHUNK):
    for j0 in range(0, nx, CHUNK):
        i1, j1 = min(i0+CHUNK, ny), min(j0+CHUNK, nx)
        chunk = np.asarray(sigma[i0:i1, j0:j1])
        k_pred, k_obs, _ = process_chunk(chunk)
        kappa_pred_full[i0:i1, j0:j1] = k_pred
        K_obs_full[i0:i1, j0:j1] = k_obs

# === Save outputs ===
fits.PrimaryHDU(data=kappa_pred_full, header=hdul[0].header).writeto(
    'bullet_kappa_predicted.fits', overwrite=True
)
fits.PrimaryHDU(data=K_obs_full, header=hdul[0].header).writeto(
    'bullet_Kact_observed.fits', overwrite=True
)

# === Compare against observed kappa map ===
with fits.open('bullet_kappa_observed.fits') as h_obs:
    kappa_obs = h_obs[0].data
    wcs_obs = WCS(h_obs[0].header)

from reproject import reproject_interp
kappa_obs_aligned, _ = reproject_interp(
    ('bullet_kappa_observed.fits', 0), wcs_baryonic,
    shape_out=kappa_pred_full.shape
)

kappa_residual = kappa_obs_aligned - kappa_pred_full
fits.PrimaryHDU(data=kappa_residual, header=hdul[0].header).writeto(
    'bullet_kappa_residual.fits', overwrite=True
)
```

§11. Acquisition Summary

Dataset	Status	Location	Action required

SL multiple-image catalog	**RETRIEVED**	`SL_multiple_image_catalog.txt` in this folder	None
JWST + HST imaging (mass reconstruction inputs)	Public archive	MAST DOI 10.17909/8zea-jv19	Run astroquery recipe §3.2 (GB-scale download)
Chandra X-ray ACIS-I (9 ObsIDs, 500 ks)	Public archive	Chandra DOI 10.25574/cdc.373	Run `download_chandra_obsid` §4.3 or browse ChaSeR (100s of MB)
Galaxy stellar mass (same MAST collection)	Public archive	MAST DOI 10.17909/8zea-jv19	Same as §3.2; SED-fit on multi-band photometry post-download
JWST ICL map (F277W)	Public archive	MAST DOI 10.17909/8zea-jv19	Same as §3.2; segmentation masking on F277W mosaic
Lensing geometry	Computable	From z_L=0.296 + Cha 2025 z_S catalog	Run §7 astropy snippet
WCS pixel registration	Embedded	In each FITS header	Use `reproject.reproject_interp` §8
Noise / uncertainty maps	Packaged	Same FITS products' ERR/WHT extensions	Read from extensions during pipeline

§12. Action Items Summary

1. **Immediate (no external retrieval needed):** the SL catalog (§2) is delivered.
2. **Single astroquery script:** retrieves all JWST + HST imaging needed for stellar mass, ICL, and ancillary lensing reconstruction (§3.2).
3. **Single CIAO command:** retrieves all 9 Chandra ObsIDs comprising the canonical 500 ks dataset (§4.3).
4. **One astropy cosmology snippet:** computes Σ_{crit_lens} for all relevant source redshifts (§7).
5. **One reproject call per FITS comparison:** aligns all maps to a single WCS grid (§8).
6. **One pipeline script:** runs the chunked pixel-κ computation end-to-end (§10).

The data are entirely public-archive. No proprietary access required. The infrastructure for retrieval (astroquery, CIAO, reproject, astropy) is open-source Python. Once the FITS products are local, the pixel-κ algorithm executes against them per the §10 pipeline.

§13. Closure Status

This manifest closes joint frontier item 5 (pixel-κ Bullet Cluster chunked-data algorithm) for the data-access dimension. Every dataset Alfred named in his "REMAINING DATASETS NEEDED" message is identified with a canonical DOI, an archive URL, and an executable retrieval recipe. The SL multiple-image catalog is delivered in-package.

What remains is execution: running the retrieval recipes, applying the chunked-data algorithm, and producing the $\kappa_{residual}(x,y)$ map. The mathematical framework is complete; the operational protocols are specified; the data sources are named.

— End of Bullet Cluster Data Acquisition Manifest —