IsomerAPI

Isomer Emission Identification Utility for the LISE^{++} Package

Technical Description, Legacy Context, and Integration Plan

Current software snapshot: Version 1.4.3 | 23-AUG-2026

Hudson Miltner | Oleg B. Tarasov | Daniel Kaloyanov

Facility for Rare Isotope Beams, Michigan State University

Document status

IsomerAPI is already distributed inside the LISE package and can be launched from LISE, but it is not yet connected to the LISE calculation/data flow. The planned next step is to replace the legacy LISE isomer-database access with reusable source-level functions derived from IsomerAPI, while preserving the established LISE gamma-yield and coincidence-analysis workflow.

Prepared from the current IsomerAPI screenshots and project documentation, together with historical LISE^{++} isomer documentation and the published LISE^{++} description.

Contents

Version note
The current screenshots identify IsomerAPI 1.4.3 (23-AUG-2026). The detailed developer modification document supplied with this report covers changes through version 1.4.2. Therefore, implementation details below are grounded in the 1.4.2 documentation unless they are directly visible in the 1.4.3 screenshots or explicitly stated as current project status.

1. Executive summary

Isomeric gamma-ray tagging provides a direct nuclear-structure signature for fragments implanted after in-flight separation. LISE^{++} historically connected its calculated fragment rates to a compact internal isomer database so that the program could estimate observable isomeric gamma-ray yields, generate an isomeric gamma spectrum, and simulate particle-identification plots in coincidence with gamma radiation [1,2].

The historical implementation was highly useful, but its database was intentionally compact. A 2006 LISE^{++} snapshot contained 1,854 gamma-ray records, mostly selected from NNDC under restricted energy and half-life conditions [2]. In contrast, the current IsomerAPI database snapshot contains 79,569 gamma-line records for 923 isotopes. This is roughly 43 times the number of gamma-line records in the 2006 internal database snapshot, while also preserving explicit level and transition relationships needed for modern decay-scheme handling.

IsomerAPI is a Qt/SQLite application for browsing, filtering, and visualizing this expanded information. It supports separate isomer and gamma views, range and exact searches in A and Z, filters in level energy, gamma energy, half-life and data source, summary statistics, and graphical level-scheme construction. The project documentation also provides a decay/transmission export path and an internal isotope -> level -> transition representation that is directly useful for future LISE integration.

At present, IsomerAPI is included in the LISE distribution and can be started from LISE, but it remains functionally independent. The intended migration is not to make LISE call the IsomerAPI executable. Instead, LISE will reuse the relevant IsomerAPI source functions and data structures so the old isomer-database functionality can be redirected to the new NNDC-based data layer.

Area	Current status (23-AUG-2026)	Purpose
Database	79,569 gamma lines; 923 isotopes	Expanded source for isomer levels and gamma transitions
Filtering / sorting	Operational	Select gamma energy, level energy, half-life, source, A and Z regions
Level-scheme plot	Operational	Visualize the decay structure of selected isotopes/isomers

Area	Current status (23-AUG-2026)	Purpose
Standalone utility in LISE package	Operational	Can be launched from the LISE distribution
Direct LISE data connection	Not yet implemented	Future replacement of the legacy isomer database
Reuse of IsomerAPI source in LISE	Planned	Direct in-process access rather than invoking the standalone GUI

2. Why LISE needs an isomeric gamma-ray database

LISE^{++} is primarily a rare-isotope production and separator-simulation code. It calculates which fragments are produced, their rates, their transmission through an in-flight separator, and the experimental particle-identification observables. For implanted-fragment experiments, however, a fragment rate alone does not tell the user which delayed gamma-ray lines can be used to confirm the identity of that fragment.

An isomer database supplies the nuclear-structure information needed to convert a calculated implanted-fragment rate into experimentally useful gamma-ray signatures. The historical LISE^{++} implementation used the isomeric population ratio, gamma intensity, half-life, level energy, gamma energy and detector/gate conditions to estimate delayed gamma-ray yields [1,2].

Scientific role of the database
The database is the bridge between separator physics and nuclear spectroscopy: LISE predicts how many fragments reach the implantation detector; the isomer information predicts which delayed gamma rays can be observed in coincidence with those fragments.

The historical LISE^{++} gamma-yield calculation can be written schematically as:

$$Y\gamma,\text{isomer} = \text{Im\_ratio} \times I\gamma \times \text{Yfragment} \times \varepsilon\text{AcqGate} \times \varepsilon\text{detect}$$

Here Yfragment is the implanted-fragment rate calculated by LISE; Im_ratio is the isomeric population/transition ratio; Iγ is the gamma-ray intensity; εAcqGate is the probability that the decay occurs inside the acquisition time gate; and εdetect is the detector efficiency. The acquisition-gate term depends on the isomer half-life, fragment velocity/energy, flight length, and the gamma-acquisition delay and width [1,2].

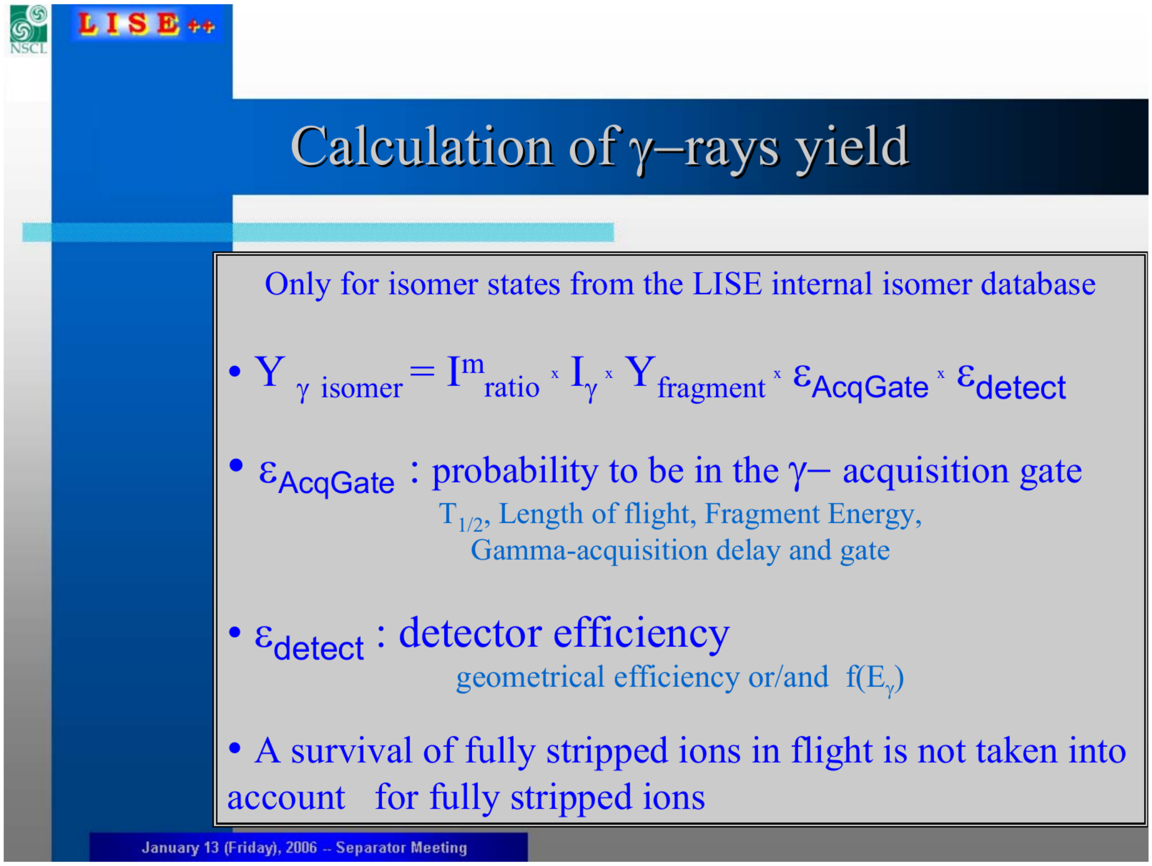

Figure 1. Historical LISE++ gamma-yield formulation (2006). The original isomer database supplied the nuclear quantities needed to combine fragment yield, acquisition-gate probability and detector efficiency. Source: "Isomers in LISE++" [2].

This capability is important in at least three practical situations: experiment planning, where the user needs to estimate whether a known isomeric line is measurable; online particle identification, where delayed gamma rays can confirm a fragment assignment; and post-implantation detector design, where half-life and gamma-energy ranges determine useful time gates and detector-efficiency requirements.

3. The original LISE++ isomer database

3.1 Historical implementation

The isomer functionality was introduced into LISE++ in the mid-2000s. The 2006 documentation describes an internal dBASE3 file, bin/isomers.dbf, with record sorting based on A, Z and gamma energy. The same documentation reports 1,854 gamma-ray records in that snapshot, including 1,752 selected from NNDC and smaller contributions from a thesis, the GANIL database and published sources [2].

The NNDC selection used in that historical snapshot was deliberately restricted: 100 < Elevel < 10,000 keV, 100 < Egamma < 4,000 keV, and 1 ns < T1/2 < 100 microseconds. The 2008 LISE++ paper later summarized the database as containing information for about 2,000 short-lived isomeric states from NNDC, the GANIL isomer database and other sources [1].

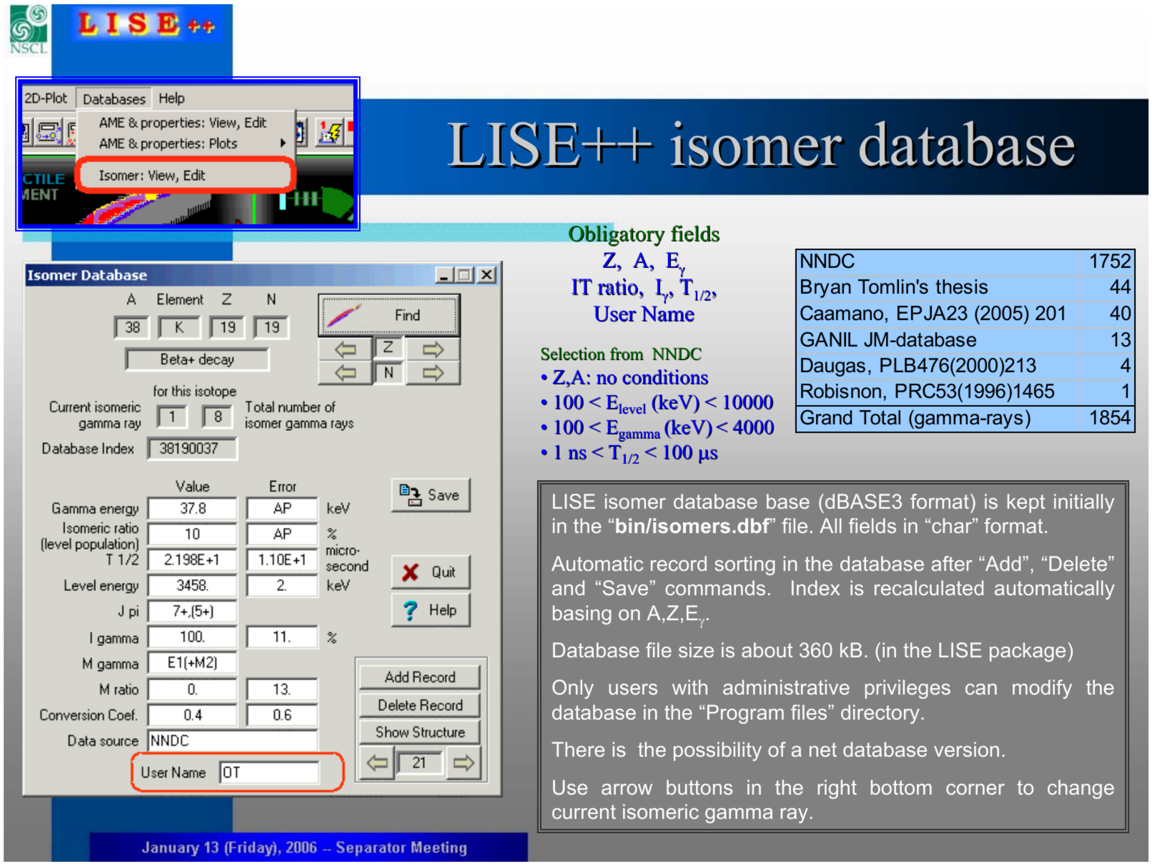

Figure 2. Historical LISE++ isomer database interface and the 2006 NNDC selection used to build the compact internal database. The slide reports 1,854 gamma-ray records. Source: "Isomers in LISE++" [2].

Historical database item	2006 LISE++ snapshot
Storage	dBASE3 file bin/isomers.dbf; fields stored as character data
Primary identity	A, Z, gamma-ray record/index
Key nuclear quantities	Egamma, isomeric ratio, T1/2, level energy, Jpi, Igamma, multipolarity, source
NNDC selection	100 < Elevel < 10,000 keV; 100 < Egamma < 4,000 keV; 1 ns < T1/2 < 100 microseconds
Database size	About 360 kB in the 2006 LISE package
Documented gamma-ray records	1,854 total; 1,752 from NNDC in the 2006 snapshot

LISE development continued after the original dBASE implementation. The LISE++ change history records an isomer-database update and migration of LISE databases from DBF to SQLite in version 16.16.01 (2023) [3]. The current IsomerAPI project follows the same SQLite direction but with a much richer NNDC-derived level/transition data set.

3.2 What the original database enabled

- Calculation of delayed isomeric gamma-ray yields associated with fragments transported to the final focal plane.
- Generation of an isomeric gamma spectrum over a user-selected energy region.

- Simulation of particle-identification plots for fragments observed in coincidence with isomeric gamma rays.
- Fast experiment-planning estimates using the same fragment yields and separator transmission already calculated by LISE++.

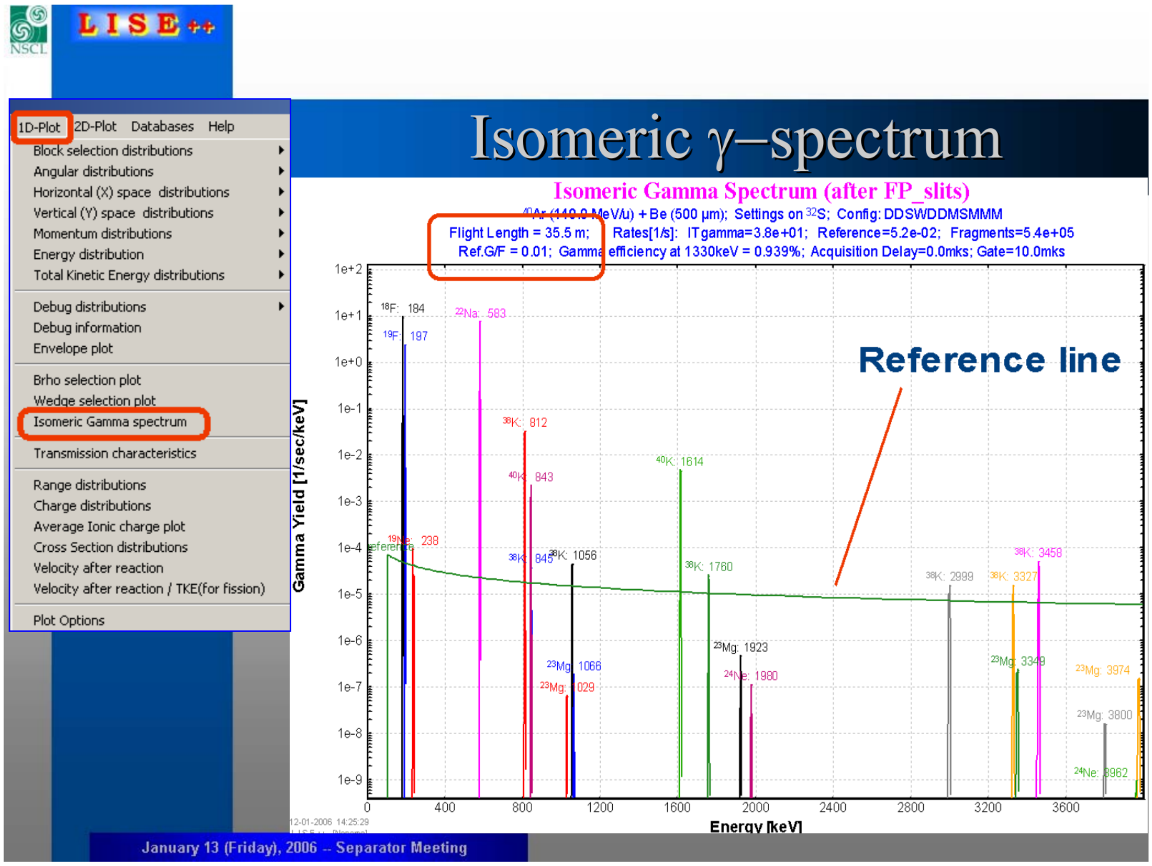

Figure 3. Example of the original LISE++ isomeric gamma-spectrum calculation. The database turns calculated implanted-fragment rates into predicted delayed gamma lines. Source: "Isomers in LISE++" [2].

3.3 Why the legacy data layer should be replaced

The original database was designed for speed, portability and the nuclear-data availability of its time. Its compact selection was entirely appropriate for the first LISE++ implementation, but it is restrictive for current rare-isotope experiments. Modern experiments routinely require broader isotope regions, more complete level schemes, multiple competing transitions, weaker branches, and direct comparison with evaluated NNDC/ENSDF records.

- The historical database snapshot contained only a small selected subset of known gamma transitions.
- The old selection imposed fixed energy and half-life windows during database construction rather than allowing the user to search a much more complete data set at run time.
- The legacy record structure was not designed around an explicit connected decay graph with level and final-level identifiers.
- Manual curation and a small internal database make long-term synchronization with evolving evaluated nuclear data difficult.
- A richer data model is needed to draw complete level schemes and to follow cascades through intermediate levels.

4. IsomerAPI: the new database and user utility

IsomerAPI is a Qt Widgets / SQLite application developed as a self-contained module for isomer-level and gamma-transition data. The current project is designed both as a useful standalone utility and as the source of reusable functionality for later integration into LISE++. The public project description likewise states that it is intended to be integrated with LISE++ and provides a GUI for isomer-level and gamma-line information [4].

Current database snapshot
79,569 gamma-line records for 923 isotopes (23-AUG-2026 project snapshot). The application uses the SQLite database nndc_DB_fullScan.sqlite and organizes filtered information into isotope, level and transition structures.

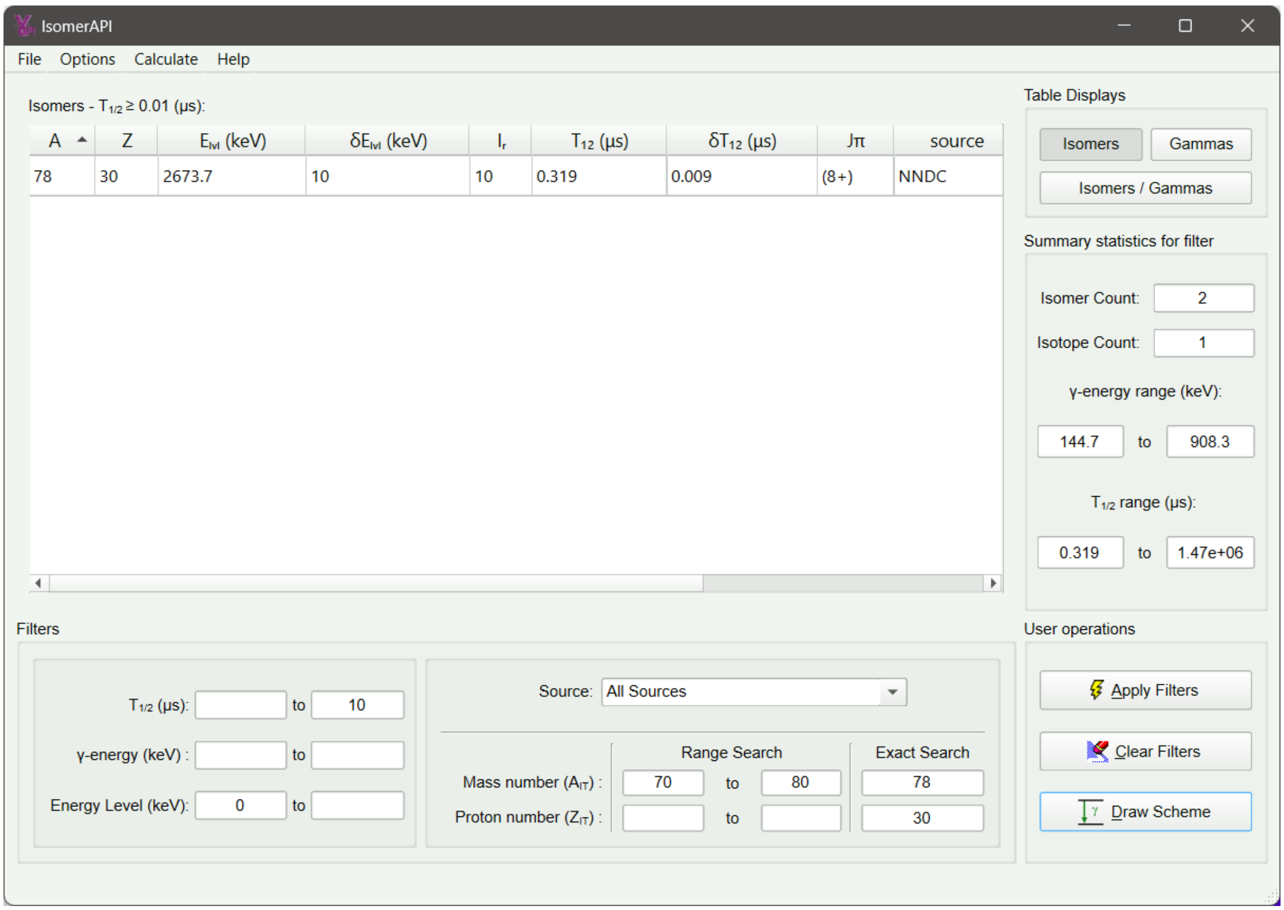

Figure 4. IsomerAPI main window, shown for 78Zn. The interface combines filtered isomer/gamma tables, summary statistics, range and exact A/Z searches, half-life and energy filters, and the Draw Scheme action.

4.1 Main user functions

Function	What the user can do	Scientific use
Isomer view	Display one representative row per isomer level after filtering	Identify candidate metastable levels and half-lives
Gamma view	Display gamma-transition rows	Search observable gamma energies and intensities
Isomers / Gammas	Use a split display of level and gamma information	Compare the isomer level with its associated transitions
A and Z filters	Apply exact or range searches; Z can also be entered by element symbol	Restrict the data set to a selected isotope or separator region

Function	What the user can do	Scientific use
Energy filters	Filter level energy and gamma energy	Match detector windows and physics-of-interest regions
T1/2 filter	Restrict half-life range	Match implantation, flight-time and acquisition-gate conditions
Source filter	Select all or specific data sources	Trace evaluated data provenance
Summary statistics	Show counts and min/max gamma energy and half-life	Check the filtered data set before plotting or export
Draw Scheme	Create a graphical level/transition scheme	Inspect decay paths and compare with NNDC records

4.2 Level-scheme visualization

The filtered SQL rows are converted into a structured isotope map containing levels and transitions. The level-scheme window can then draw energy levels and gamma transitions for the selected isotope. The project documentation states that transition-line thickness is proportional to gamma intensity, and the scheme window provides links to NNDC records for comparison.

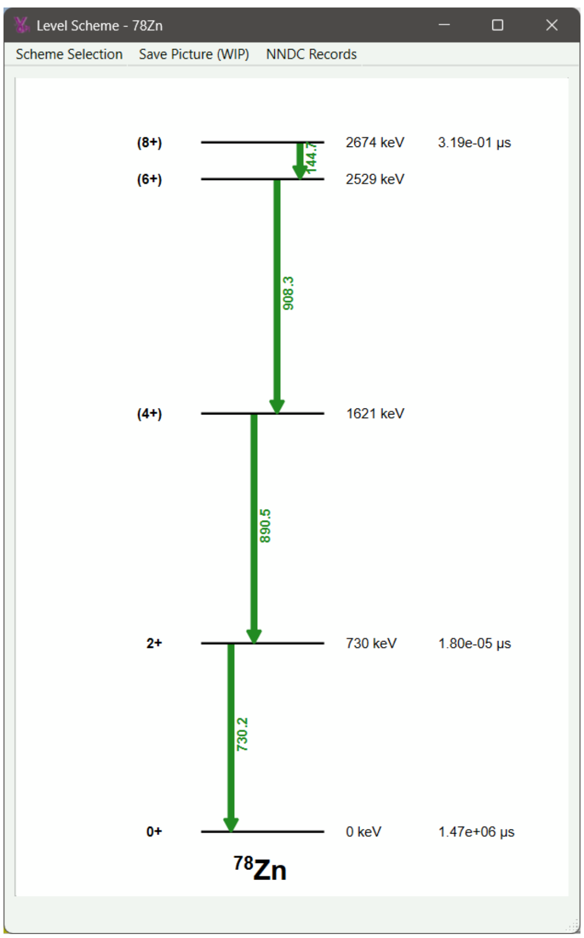

Figure 5. Level scheme for 78Zn.

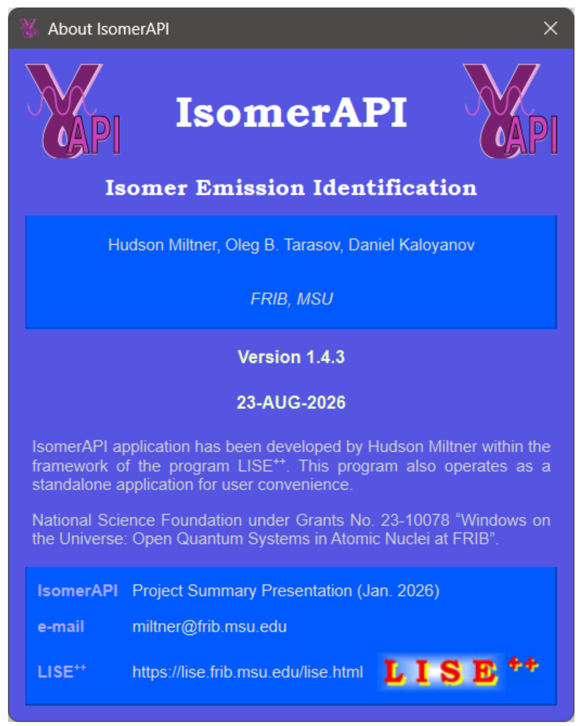

Figure 6. About dialog for IsomerAPI 1.4.3 (23-AUG-2026).

4.3 Why the expanded data set matters

The expanded database changes the workflow from "look up a small curated list of lines" to "search a broad evaluated level/transition data set and apply experiment-specific cuts at run time." That distinction is important:

detector energy range, separator flight time, implantation gate, and isotope region vary from experiment to experiment, so those constraints should be user-selectable rather than permanently embedded in the database construction.

The explicit level and transition identifiers also make the new data suitable for decay-cascade traversal. In particular, the current schema includes LEVEL_ID, GAMMA_ID and FINAL_LEVEL_ID, allowing software to associate a transition with its parent and final levels rather than treating gamma lines as independent records.

5. IsomerAPI software architecture

The current implementation separates database access, filtering, model views, derived isotope structures, and user actions. The detailed 1.4.2 developer documentation records substantial work to prevent unbounded loading of the full NNDC scan, improve proxy-model performance, keep drawing windows under correct ownership, and split the main implementation into focused filtering, drawing and transmission source files.

- SQLite / Qt SQL layer: QSqlDatabase and QSqlTableModel operate on the Isomers table in lisecfg/nndc_DB_fullScan.sqlite.
- Filtering layer: applyFilters() builds SQL WHERE expressions; the source filter supports all sources or selected source sets.
- Proxy models: the level proxy keeps isomer-level rows, while the gamma proxy keeps rows containing gamma-energy information.
- Derived data: prepData() converts filtered database rows into an Isotope -> Levels -> Transitions representation.
- User actions: table selection, drawing, NNDC record access, decay/transmission text export, and summary statistics.
- Performance safeguards: bounded startup behavior, an empty-result path for Clear Filters, and a cached first-occurrence lookup for duplicate level rows in the 1.4.2 implementation.

IsomerAPI block scheme

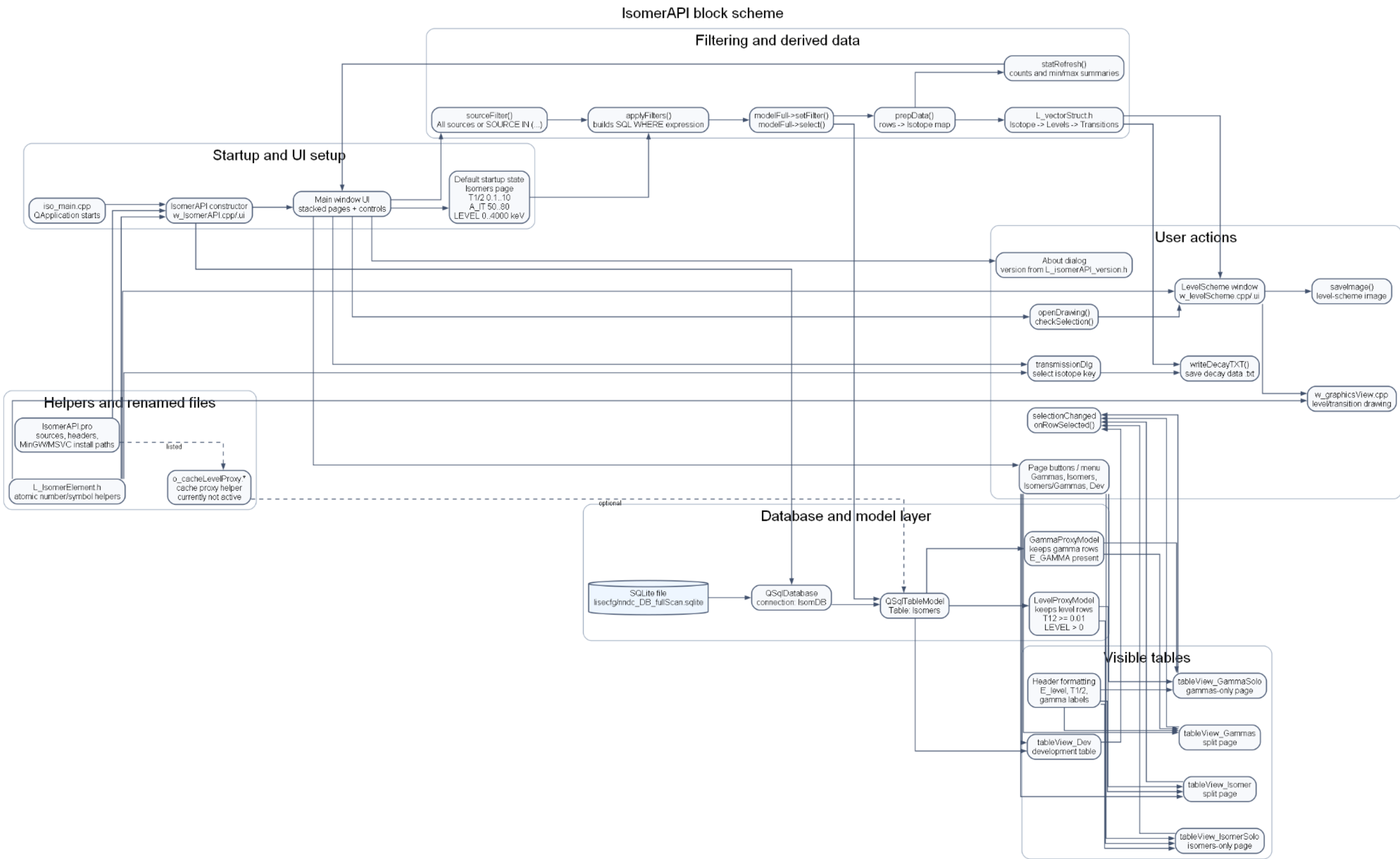

Figure 7. Current IsomerAPI block scheme: startup and UI setup, SQLite/model layer, filtering and derived data, visible tables, user actions, and helper/source files.

6. Operation inside the LISE package

IsomerAPI is currently included in the LISE package as an auxiliary code. In the current LISE user interface it can be started directly from the IsomerAPI toolbar action or through Menu -> Utilities -> Codes -> IsomerAPI. This makes the new database and plotting utility immediately available to LISE users without requiring a separate installation.

> **Important current limitation**
> Launching IsomerAPI from LISE does not yet establish a calculation or data connection between the two programs. LISE does not currently pass the selected fragment, calculated implantation rate, separator flight information, or detector-gate settings into IsomerAPI, and IsomerAPI does not yet replace the legacy LISE isomer routines.

6.1 Typical IsomerAPI workflow today

1. Launch IsomerAPI from the LISE package toolbar or from Utilities -> Codes.
2. Select the Isomers, Gammas, or Isomers / Gammas table display.
3. Restrict the isotope region using A and Z exact/range filters.
4. Apply half-life, level-energy, gamma-energy and source filters appropriate to the experiment.
5. Inspect the summary statistics to verify the selected region.
6. Select an isotope/isomer and use Draw Scheme to inspect its level and gamma-transition structure.
7. When needed, use the existing decay/transmission text-export path as a development interface to the level/transition information.

6.2 Example: 78Zn

The supplied screenshots illustrate a simple use case for 78Zn (A=78, Z=30). The isomer table identifies the approximately 2674-keV (8+) isomer with T1/2 = 0.319 microseconds. The level-scheme display shows the cascade through the 2529, 1621 and 730-keV levels to the ground state, with gamma energies labeled on the transition arrows. This is exactly the kind of structure that the legacy compact gamma-line list could not present as naturally as the new connected level/transition model.

7. Planned replacement of the legacy LISE isomer database

7.1 Integration principle

The planned LISE integration should reuse IsomerAPI source functions and data structures in-process rather than invoking the IsomerAPI executable as a separate calculator. The standalone IsomerAPI GUI remains valuable for browsing and validation, but LISE needs direct access to the same database logic while it is calculating fragment rates and experimental observables.

> **Design goal**
> One nuclear-data implementation, two interfaces: IsomerAPI provides the dedicated browsing/visualization GUI, while LISE calls the same underlying data-access and decay-processing functions for production calculations and gamma-coincidence simulations.

7.2 Proposed LISE -> isomer-data flow

LISE fragment calculation	->	**Reusable IsomerAPI data layer**	->	**Gamma candidates and decay paths**	->	**LISE gamma spectrum / coincidence plots**

For each fragment/isotope considered by LISE, the calculation layer can pass at minimum A, Z and the calculated implanted-fragment rate. For a full gamma-yield estimate, LISE also provides the fragment time/energy or flight-time information and the gamma-acquisition/detector settings. The new data layer returns candidate isomer levels, transition energies and intensities, half-lives, branching/IT information and connected decay paths. LISE then applies its established gamma-yield and identification-plot calculations using the richer nuclear-data source.

7.3 IsomerAPI components most suitable for reuse

Component / concept	Reuse in LISE	Reason
SQLite data access	Yes	Provides a single maintained source for the expanded NNDC-based data set
A/Z and physics filters	Yes, adapted to LISE calls	LISE can request only the isotope/half-life/energy region needed for a calculation
Isotope -> Level -> Transition structures	Yes	Natural in-memory representation for cascades and gamma candidates
Decay traversal using final-level links	Yes	Supports cascades through intermediate levels and future transmission/gate logic
Summary/validation utilities	Useful	Allow consistency checks and diagnostics during migration
Qt table widgets / standalone main window	No for the LISE calculation path	The LISE GUI already provides the experiment/separator context
Level-scheme window	Optional shared utility	Can remain available as a dedicated visualization tool from either application context

7.4 Functions to preserve from the original LISE workflow

Replacing the data source should not remove the physics workflow that made the original LISE isomer feature useful. The following user-facing capabilities should be preserved or improved:

- Estimate delayed gamma-ray yield from the LISE-calculated implanted-fragment rate.
- Include the isomer half-life and the flight/acquisition gate in the probability that the decay is recorded.
- Include detector efficiency, including an energy-dependent efficiency function when selected.
- Generate an isomeric gamma spectrum for the calculated fragment distribution.
- Generate particle-identification plots in coincidence with isomeric gamma radiation.
- Allow a user to inspect the source level scheme and evaluated nuclear-data record for any line used in the calculation.

7.5 Recommended staged migration

Stage	Work	Acceptance condition
1. Data access	Create a small reusable C++ data-access layer around the current SQLite schema; no GUI dependency.	LISE can query levels/transitions for a specified A,Z and reproduce an IsomerAPI query.
2. Legacy compatibility	Map the returned data to the quantities used by the existing LISE gamma-yield routine.	For lines present in the old database, new and old calculations agree when the same nuclear inputs are used.
3. Rich decay paths	Use LEVEL_ID / FINAL_LEVEL_ID to follow cascades and choose transitions from the selected isomer level.	Known test isotopes produce the expected level scheme and gamma list.
4. LISE UI connection	Reconnect the existing LISE isomer spectrum / coincidence controls to the new data layer.	Users can run the old LISE workflow without opening the standalone IsomerAPI window.

Stage	Work	Acceptance condition
5. Validation and retirement	Benchmark old vs new database coverage and document differences; deprecate the legacy database.	No legacy-only calculation path remains except an optional compatibility mode.

8. Validation and acceptance criteria

The migration should be validated at both the database and physics levels. Because the new data set is much larger, simple record-count agreement with the old database is neither expected nor desirable. Validation should instead separate identical legacy cases from newly available transitions.

- Database identity: verify A, Z, level energy, half-life, gamma energy, gamma intensity, multipolarity and source against NNDC records for a representative isotope set.
- Legacy regression: for gamma lines common to both databases, run the same LISE fragment rate, gate and detector efficiency and verify the same gamma-yield formula produces consistent results.
- Cascade validation: check isotopes with multi-step cascades (such as the demonstrated 78Zn case) to confirm parent/final-level links and plotted energy ordering.
- Performance: ensure LISE queries are bounded by A/Z and physics cuts so the full 79,569-line data set is not fetched unnecessarily for every update.
- User reproducibility: show the data source and selected transition details for every gamma line used in a LISE calculation.
- Package validation: confirm both MinGW and MSVC LISE builds locate the SQLite database correctly in the deployed lisecfg structure.

Recommended first benchmark set
Use several isotopes already represented in the historical LISE database plus 78Zn from the current screenshots. For each case, compare (1) legacy LISE gamma lines, (2) IsomerAPI lines after applying equivalent cuts, (3) calculated gamma yields with identical gate/detector settings, and (4) the plotted decay scheme against NNDC.

9. Summary

The original LISE++ isomer database was an important extension of fragment-separator simulation: it allowed calculated fragment yields to be translated into delayed gamma-ray observables and coincidence-identification predictions. Its historical compact data set was well matched to the computing and nuclear-data environment of the mid-2000s, but it is too limited for a modern, maintainable implementation.

IsomerAPI provides the natural replacement path. It already offers a modern SQLite data layer, a greatly expanded gamma-transition set, flexible run-time filtering, an isotope/level/transition representation, NNDC-linked level-scheme visualization, and a standalone GUI that is distributed inside the LISE package. The remaining work is the connection layer: LISE should call the relevant IsomerAPI-derived source functions directly and use the new data to feed the established gamma-yield, spectrum and coincidence-analysis calculations.

This approach preserves the successful physics model and user workflow of the original LISE implementation while replacing the old database with a more complete, inspectable and maintainable nuclear-structure data source.

Appendix A. Data model and current implementation notes

A.1 Representative SQLite fields used by IsomerAPI

Category	Representative fields	Role
Isotope identity	A_IT, Z_IT	Mass and proton number; used for exact/range selection and isotope keys
Level data	LEVEL, D_LEVEL, JPI, T12, D_T12, IT_RATIO, LEVEL_ID	Excitation energy, uncertainty, spin/parity, half-life, isomeric ratio and unique level identity
Gamma-transition data	E_GAMMA, D_EG, I_GAMMA, D_IG, M_GAMMA, GAMMA_ID, FINAL_LEVEL_ID	Transition energy/intensity/multipolarity and connection to the final level
Provenance	SOURCE	Evaluated/source record used for filtering and traceability

A.2 Current code organization (v1.4.2 documentation)

Area	Representative implementation
Main window	w_IsomerAPI.cpp/.ui - construction, high-level UI wiring and actions
Filtering/data preparation	w_IsomerAPI_filters.cpp - SQL filter construction, source handling, summary statistics, prepData()
Drawing	w_IsomerAPI_drawing.cpp plus level-scheme/graphics classes - selection and scheme window path
Transmission/export	w_IsomerAPI_transmission.cpp - decay/transmission text export and decay traversal
Level proxy	L_levelProxyModel.* - first-occurrence cache for duplicate-level suppression
Project/deployment	IsomerAPI.pro - source list, build metadata, MinGW/MSVC support and debug trace switch

A.3 Implementation cautions

- Do not reintroduce unfiltered full-database fetches during LISE integration; A/Z and physics constraints should be applied at the SQL/query boundary.
- Keep the data-access and decay-processing layer independent of the standalone IsomerAPI main window so LISE can reuse it without creating a second GUI.
- Treat the current v1.4.3 screenshot as the software-status reference and the v1.4.2 developer document as the detailed implementation reference until a 1.4.3 developer change log is available.
- The screenshot labels Save Picture as WIP; image/export features should be documented as complete only after they are explicitly verified in the release being distributed.

References

[1] O. B. Tarasov and D. Bazin, "LISE^{++}: Radioactive beam production with in-flight separators," Nuclear Instruments and Methods in Physics Research B 266 (2008) 4657-4664. DOI: 10.1016/j.nimb.2008.05.110. https://lise.frib.msu.edu/doc/lisepp_nim_2008.pdf

[2] O. B. Tarasov, "Isomers in LISE^{++}," LISE^{++} separator-meeting presentation, 2006. Historical description of the LISE isomer database, gamma-yield calculation, isomeric gamma spectrum, and gamma-coincidence identification plots. https://lise.frib.msu.edu/paper/isomers.pdf

[3] LISE^{++} software change history. Version 16.16.01 records an update of the isomer database and migration of LISE databases from DBF to SQLite (2023). https://lise.frib.msu.edu/changes.html

[4] LISEcute/IsomerAPI public repository description: self-contained Qt/C++ project intended for integration with LISE^{++}, with GUI access to isomer-level and gamma-line information. https://github.com/LISEcute/IsomerAPI

[5] IsomerAPI Manual, project documentation, 23-AUG-2026.

[6] IsomerAPI First Steps, project user manual, 23-AUG-2026.

[7] IsomerAPI modifications since version 1.3.23, developer documentation through v1.4.2, 22-AUG-2026.

[8] IsomerAPI block scheme, project architecture diagram, 23-AUG-2026.

End of document