Spine Sorter v5.51

Automated Asset Optimization & Integrity Tool

- Technical Artist / Animator Workflow
- Runtime Optimization (VRAM/Disk)
- Automated QC & Integrity Verification

Pipeline Objective

- Automates pre-export asset processing.

- Optimizes build size via intelligent texture sorting (JPEG vs PNG).

- Enforces build integrity by verifying internal export flags.

- Standardizes output structure for engineering handoff.

Workflow (Standard Operation)

1. Target source directory containing **.spine** project files.

2. Select specific skeleton/project file from queue.

3. Execute processing pipeline ('Run Selected File').

4. Review generated build report and error logs.

Feature: Intelligent Compression Sorting

JPG Pipeline: Automatically routes fully opaque textures and 'Additive' blend mode slots to JPEG to reduce specific disk footprint.

PNG Pipeline: Preserves Alpha channel for transparent/translucent assets and soft-edge gradients.

Ref Isolation: Segregates non-production reference images from the final build package.

QC: Export Flag Validation

[WARNING] Export Disabled (Skeleton)

Issue: Entire Skeleton (e.g., 'Reference') exists in project but missing from export.

Impact: Major runtime crash (Missing Component).

[WARNING] Export Disabled (Asset)

Issue: Identifies active attachments/animations with disabled 'Export' flags.

Impact: Prevents 'missing asset' runtime errors.

Action: Flagged assets must be re-enabled in the Spine Tree view.

QC: Asset Integrity

[WARNING] Alpha Channel Conflict

Detection: Assets in the 'opaque' pipeline detected with non-zero alpha pixels.

Auto-Correction: Automatically reroutes to PNG pipeline to prevent white-halo artifacts at runtime.

[WARNING] Broken References

Detection: Validates internal JSON path references against local file system availability.

New: Animation Sync Validation

The Issue: Silent Export Failures

• Spine's 'Export' checkbox for animations is local to the project file but critical for the runtime skeleton data.

• Accidental unchecking results in successful builds but missing logic in-game (e.g., 'Frozen' idle states).

Solution: CLI-Based Verification

Logic: Source-to-Binary Comparison

Introspection: Queries the active Spine Editor CLI for the authoritative 'Source of Truth' animation list.

Analysis: Parses the exported skeleton data (JSON).

Verification: Diffs the lists to ensure 1:1 parity between Source and Build.

Detection Heuristics

- Filtration of duplicates and system noise.

- Multi-Version Support: Spine 3.8, 4.0, 4.1, 4.2+.

- Deep Scan: Validates multiple skeletons per project (e.g., Reference vs Main).

- Parser: Handles both legacy indented and modern comma-separated CLI output formats.

Build Report

Attachment Density: Total slot/attachment count.

Texture Atlas Paths: Validated file paths processed for packing.

Animation Parity: Source vs. Export match count (e.g., 18/18).

Compression Ratio: Final breakdown of JPEG vs PNG texture count.

Configuration Profiles

Post-Process Launch: Auto-load project in Editor for visual verification.

Aggressive Compression: Force opaque assets to JPEG (Recommended for Mobile/WebGL optimization).

Legacy Mode: Bypass sorting logic for maintenance of older project structures.