08\_FilesOnDisk

## 📁 Files on Disk (Preliminary Reference)

This document outlines the purpose, structure, and naming conventions of files generated or used by the Mango Workbench and its cryptographic tools. It serves both as internal reference and the foundation for a formal publication section.

---

## 🧬 File Naming Convention Breakdown

A large number of files follow this structured pattern:

<Name>,-L<N>-P<P>-D<T>-M<M>-S<S>.<ext>

🔤 Components Explained:

* - Name → The file type or purpose (e.g., Contenders, MungeFailDB, BTRFailDB, etc.)
* - L<N> → The Munge Level, where L1 to L5 refers to the sequence depth tested.
* - P<P> → The PassCount threshold — minimum number of metrics that must pass for a sequence to be retained as a contender.
* - D<T> → The InputType / DataType:
* - DC → Combined
* - DN → Natural
* - DR → Random
* - DS → Sequence
* - M<M> → The Evaluation Mode:
* - C → Cryptographic (standard, most common)
* - E → Exploratory (used for alternative scoring exploration)
* - S<S> → The Scoring Format:
* - SF → Practical Scoring (default; ComputePracticalScore)
* - ST → Theoretical/Metric Scoring (ComputeMetricScore)

These tags are always joined by dashes and commas as delimiters.

---

## 📄 Representative Files and Their Purpose

## 🔹 Contender Files

Contenders,-L<N>-P6-D<T>-MC-SF.txt

* - Stores the top-ranked sequences discovered at Munge Level L<N>.
* - One per data type (DC, DN, etc.).
* - Sorted by descending score. Top sequence is the winner.

## 🔹 State Files

State,-L5-P6-D<T>-MC-SF.json

* - Periodic snapshot used to resume a Munge in case of power outage or user interruption (e.g., Ctrl-C).
* - Resumption is automatic. Simply rerunning the Munge will detect the snapshot and continue from where it left off.

## 🔹 Fail Databases

MungeFailDB,-P6-D<T>-MC-SF.db

BTRFailDB,-P0-D<T>-MC-SF.db

* - SQLite databases that log sequences which failed to meet minimum pass-count thresholds.
* - Prevents re-testing of poor performers.

## 🔹 Benchmark Results

TransformBenchmarkResults.json

TransformBenchmarkResults.txt

TransformProfileResults.json

* - Outputs from performance profiling tools.
* - Includes metrics like average execution time per transform.

## 🔹 Input Data and Supporting Files

Frankenstein.txt / .bin

randoms.bin

* - randoms.bin — Canned randomized input data for profiling. Used to ensure test reproducibility.
* - Frankenstein.\* — Text sourced from published books, used to populate the Natural InputType.
* - Sequence data is generated dynamically.
* - Combined InputType is a structured blend of Natural, Random, and Sequence.

## 🔹 Workbench State Files

ConsoleState.json

GlobalSettings.json

MangoConfig.txt

* - ConsoleState.json — Stores command history from the interactive shell.
* - GlobalSettings.json — Persists workbench configuration (Rounds, Mode, PassCount, etc.).
* - MangoConfig.txt — Runtime marker indicating what Mango is currently executing. Useful for long-running Munge operations.

---

This is a living document. As additional tooling and export formats are finalized, more entries will be appended or structured accordingly.

✍️ Final draft will include screenshots, flow diagrams, and usage examples.