International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

An Evidence-Aware Framework for Cross-Artefact Windows Browser Forensic Investigation and User Activity Reconstruction

Aisha Rabiu Ibrahim[1], Akash Kumar Singh[1], Abdulrahim Magaji[1]

[1] Department of Forensic Science, Vivekananda Global University, Jaipur, India
24wbsc3fs003@vgu.ac.in

Abstract - Web browsers are a principal source of digital evidence on Windows systems, yet records are distributed across multiple browsers, profiles, storage formats and operating-system artefacts. The reviewed literature addresses artefact analysis, acquisition constraints and general timeline reconstruction, but an integrated treatment of evidence completeness, corroboration, potentially inconsistent evidence and traceability from findings to source evidence is not evident for Windows browser evidence. This paper proposes an evidence-aware framework for Windows browser forensic investigation and user activity reconstruction. The framework organizes the investigation as a layered lifecycle from case management, discovery, acquisition and preservation, through normalization and cross-browser and cross-artefact correlation, to a forensic intelligence module, activity reconstruction, visualization, traceable findings and audit. A unified evidence model links evidence items, artefacts, events, relationships, activities and findings. The forensic intelligence module comprises evidence-quality assessment, corroboration, contradiction flagging, traceability, anti-forensic review, reconciliation and hypothesis testing; its outputs support investigator judgement and do not determine truth, intent, identity, authorship or legal responsibility. The principal contribution is integrating these functions into a coherent, testable architecture. An evaluation methodology based on ten controlled ground-truth scenarios and measurable dimensions is specified. No empirical results are reported; implementation and empirical validation are required before effectiveness can be claimed.

Key Words: Windows browser forensics, digital evidence, cross-artefact correlation, user activity reconstruction, evidence provenance, forensic intelligence.

1. INTRODUCTION

Browser records of navigation, downloads and stored web data have long been used as evidence of user activity, and early work established methods for collecting and analyzing them [1], [2]. On Windows systems, a single machine may host several browsers, each with several profiles, under several user accounts. Browser data are held in heterogeneous structures: relational databases for history [2], key–value stores for web storage such as IndexedDB [3], and time values whose formats differ between browsers [1]. Protection mechanisms add further variation; for example, Chrome 127 introduced application-bound encryption for cookies on Windows [4].

Consequently, evidence of one user action, such as a search, a visit and a download, may be fragmented across artefacts, profiles, browsers and operating-system sources. Analysis of a single browser or a single log file is inadequate when several browsers are used [1]. Private browsing modes aim to limit local persistence, yet residual artefacts have been reported in files, host artefacts and memory [5]–[9], so the availability of evidence varies with browser, version and condition of use.

Relating browser evidence to other Windows sources is therefore necessary but difficult. Guidance for forensic practice recognizes that evidence comes from different data sources such as files, operating systems and applications [10], and timeline research has shown how very large numbers of low-level events can be abstracted to higher-level events while retaining their provenance [11]. Investigators must also judge how complete the acquired evidence is, which records support one another, and which appear inconsistent. Error, uncertainty and loss are recognized properties of digital evidence [12], approaches to expressing the strength of digital evidence have been proposed [13], [14], and tools exist for detecting inconsistency in general computer activity timelines [15], [16]. Timeline-based event reconstruction is nevertheless described as fragmented in terminology and method [17].

The reviewed literature therefore indicates that the elements needed for investigating Windows browser evidence exist separately, whereas a single architecture that integrates acquisition-coverage reporting, correlation, corroboration, contradiction flagging and finding-level traceability for this evidence is not evident in the sources examined (Sec. 3). This absence in the examined sources does not prove absence in the wider literature.

This paper proposes such a framework. Its objectives are: (O1) a unified, provenance-preserving evidence model; (O2) cross-browser and cross-artefact correlation dimensions; (O3) evidence-quality, corroboration and contradiction functions that support investigator judgement; (O4) traceable activity reconstruction and reporting; and (O5) an empirical evaluation methodology based on controlled ground truth. The corresponding research questions are: **RQ1** How can heterogeneous Windows browser evidence be represented in a unified, provenance-preserving model? **RQ2** How can cross-browser and cross-artefact relationships be defined and labelled? **RQ3** How can quality, corroboration and potential contradiction be exposed without implying determination of truth? **RQ4** How can activities and findings remain traceable to preserved evidence? **RQ5** How should the framework be evaluated empirically?

The contributions are conceptual and methodological: an evidence-aware architecture, a unified evidence model with eight correlation dimensions, a forensic intelligence module with cautious semantics, and a proposed evaluation methodology. The components draw on established concepts; the principal contribution is their integration. Section 2 reviews related work, Sec. 3 derives gaps and requirements, Sec. 4 presents the framework, Sec. 5 proposes the evaluation, and Secs. 6 to 9 present the discussion, limitations, future research and conclusion.

2. BACKGROUND AND RELATED WORK

The literature is synthesized by theme. For each, established knowledge, existing approaches, limitations and the resulting requirement (FR1–FR20, defined in Table 2) are indicated.

2.1 Browser Artefacts, Acquisition, Profiles and Persistence

Browser history, downloads and search terms are standard evidence sources, and methods exist for integrated timeline analysis, search-word analysis and recovery of deleted records across several browsers [1]. Database recovery for Firefox history has been demonstrated, including deleted records [2], and the internal structure of Chromium's key–value storage has been studied for forensic use [3]. Acquisition guidance emphasizes identification, collection, acquisition and preservation of potential evidence [18]. Protection mechanisms such as application-bound encryption restrict which processes can decrypt Chrome cookie data on Windows [4], so recoverability depends on browser version and conditions. Existing studies typically address one browser, format or tool, and do not report case-level coverage of what was and was not acquired. *Requirement:* documented acquisition, preservation, hashing, and explicit recording of unavailable sources (FR2–FR6, FR10).

2.2 Private, Deleted and Memory-Resident Browsing

Private browsing has stated privacy goals and known weaknesses [5]. Forensic studies report residual artefacts from private and portable browsing [6] and from the Tor Browser on Windows 10 [7]. Memory analysis can recover browsing activity, including from private modes, in Chromium-based browsers [8], and memory and browser artefacts of a web application have been examined across browsers on Windows 10 [9]. Findings are specific to versions, configurations and scenarios, and absence of records does not establish concealment. *Requirement:* review of evidence-disrupting conditions without presuming intent (FR14).

2.3 Windows Corroborating Artefacts and Cross-Artefact Correlation

Windows systems hold execution, file-access and registry records beyond the browser, and forensic guidance covers files, operating systems and applications as data sources [10]. General timeline research relates low-level events, such as file modifications and registry updates, to higher-level events with preserved provenance [11]. Combined browser and memory artefacts have been examined for a specific application [9]. A browser-centred correlation architecture across Windows sources is not reported in these sources, and artefact availability depends on system configuration. *Requirement:* selected Windows artefact classes handled under the same event model and correlated with browser evidence (FR7, FR9).

2.4 Cross-Browser Correlation and Activity Reconstruction

Integrated analysis across browsers requires harmonization of time formats [1]. Event reconstruction was formalized early as an event-based investigation framework that develops and tests hypotheses [19], and recent work classifies the field's terminology and challenges, noting fragmentation [17]. Reconstruction supports inference about past activity but does not by itself establish identity or intent. *Requirement:* common event structure; correlation across browsers and profiles; activity episodes separated from attribution (FR7, FR8, FR15).

2.5 Evidence Provenance and Integrity

Process models and standards emphasize preservation and documentation of evidence [10], [18]–[20]. Provenance from derived events back to raw data has been demonstrated for general timelines [11]. This principle has not been reported, in the sources examined, as combined with browser-specific corroboration and contradiction analysis. *Requirement:* hashing, provenance records and links from every derived object to its source (FR4, FR5, FR13).

2.6 Evidence Completeness, Corroboration and Contradiction

Error, uncertainty and loss are inherent to digital evidence [12]; structured expression of evaluative opinion and of evidence certainty has been proposed [13], [14]. Inconsistency in activity timelines can be detected [15], and implicit timing information can be used to characterize timestamp inconsistencies [16]. These approaches are general and are not specific to browser and Windows evidence. *Requirement:* transparent quality, corroboration and contradiction functions that flag, and do not resolve, relationships for review (FR10–FR12).

2.7 Anti-Forensic Analysis, Visualization and Explainable Analysis

Anti-forensics has been defined as attempts to compromise the availability or usefulness of evidence, with taxonomies of techniques [21], [22]. Taxonomies also include indications of anti-forensic activity, which are not proof of intent [22]. Visualization has been used to present high-level events derived from timelines [11]. Evidence that machine learning provides measurable benefit for browser evidence is not established in the sources examined. *Requirement:* transparent rule-based functions, timeline and graph views linked to evidence, and disruption indicators without attribution (FR14, FR16).

3. RESEARCH GAP AND DESIGN REQUIREMENTS

Table 1 relates each gap to the evidence available in the examined sources. Gaps marked "not established" are design requirements derived from the framework specification and are not claimed as proven deficiencies of the literature.

Table -1: Research gaps, supporting evidence, limitations and framework requirements.

Gap	Evidence from literature (support status)	Limitation	Framework requirement
G1 Fragmented acquisition and storage	Heterogeneous artefacts and storage [1]–[3]; memory sources [8], [9]. *Supported.*	No case-level record of acquired and unavailable sources reported.	FR1–FR6 discovery, acquisition, preservation, hashing, repository
G2 Limited cross-browser integration	Integrated multi-browser analysis proposed for browsers of its period [1]. *Partially supported.*	Current browser and format coverage not established.	FR7–FR8 common event model; cross-browser correlation
G3 Limited cross-artefact correlation	Low- to high-level event abstraction [11]; browser and memory artefacts for one application [9]. *Partially supported.*	Not a general browser-to-Windows correlation architecture.	FR9 browser-to-browser and browser-to-Windows correlation
G4 Limited completeness visibility	*Not established*; conceptually related to loss and uncertainty [12].	Acquisition coverage not reported as an output.	FR10 evidence quality and coverage

Gap	Evidence from literature (support status)	Limitation	Framework requirement
G5 Limited explicit corroboration	*Not established* for browser evidence; strength-of-evidence methods exist generally [13], [14].	Supporting relationships not reported as explicit outputs.	FR11 corroboration matrix
G6 Limited contradiction handling	Timeline inconsistency detection [15], [16]. *Partially supported* (general timelines).	Not browser-specific; not combined with corroboration.	FR12 flagging of potential conflicts
G7 Fragmented reconstruction	Field described as fragmented [17]. *Supported.*	Terminology and methods vary.	FR15 evidence-supported activity episodes
G8 Weak finding traceability	Provenance preserved for general timelines [11]. *Partially supported.*	Not combined with browser corroboration and reporting.	FR13 finding → activity → event → artefact → evidence
G9 Evidence disruption	Residual artefacts [5]–[7]; anti-forensic taxonomies [21], [22]. *Supported.*	Findings are version- and scenario-specific.	FR14 evidence-disruption review
G10 Interpretation burden	Large numbers of low-level events per disk [11]. *Supported* (general).	Manual comparison across artefacts remains.	FR16–FR18 views, reconciliation, hypothesis workspace
G11 Inconsistent validation	*Not established*; studies report individual designs [6], [7].	No common ground-truth approach identified.	FR20 ground-truth evaluation methodology

Note. "Supported" indicates support in the cited source; it does not imply exhaustive confirmation.

The framework follows these design principles: preserve evidence before interpretation; maintain provenance; verify integrity cryptographically where applicable; isolate cases and keep investigators accountable; use transparent rules; separate evidence from interpretation; handle missing and conflicting evidence explicitly; avoid unsupported attribution and legal conclusions; and keep outputs reproducible and traceable. These are consistent with established guidance [10], [18].

4. PROPOSED FRAMEWORK

4.1 Framework Overview and Architecture

The proposed framework (specified as WinBrowser Forensics) is a conceptual architecture for acquiring, preserving, normalizing, correlating, reconstructing, interpreting and reporting Windows browser evidence and selected related Windows evidence. It is not a browser-history viewer; its proposed contribution is the integration of evidence handling with functions that expose evidence quality, support, potential inconsistency and traceability. It covers multiple browser families, profiles and Windows users. It does not claim universal recovery of deleted or private browsing data, does not determine legal responsibility, guilt, intent or identity, and does not treat machine-generated interpretation as independent evidence. Two pathways enter the same chain: direct acquisition from a Windows system, and import of evidence acquired elsewhere with provenance and integrity metadata.

The architecture is a layered chain from authentication to audit (Fig. 1); each layer has a defined input, processing role and output, and analytical results remain linked to their source evidence. The components, and the functional requirements they satisfy, are summarized in Table 2.

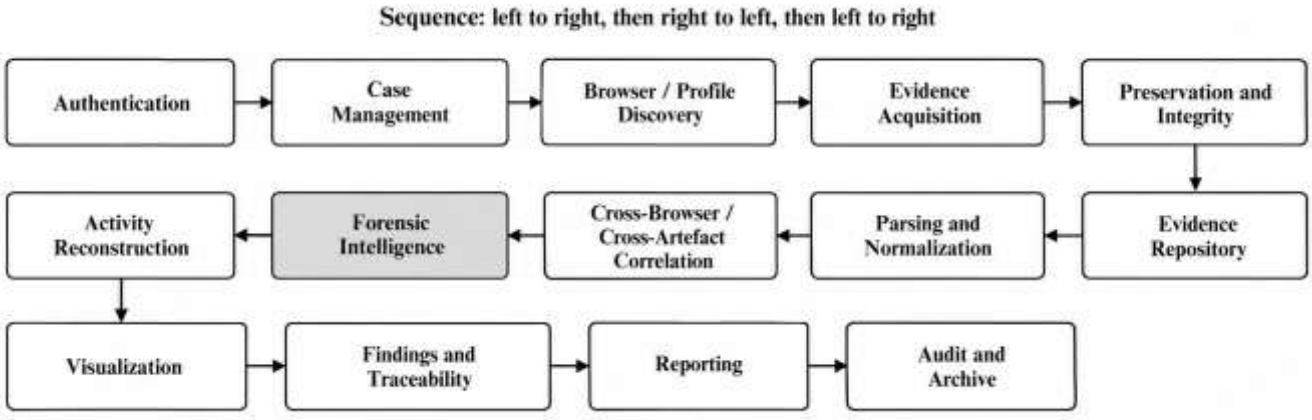


Fig -1: Proposed framework architecture: layered chain from authentication and case management to reporting and audit.

Table -2: Framework components and functional requirements.

ID	Component	Purpose	Principal output
FR1	Authentication and case management	Authenticate investigators; isolate cases	Session identity; case context
FR2	Browser/profile discovery	Identify Windows users, browsers, profiles	Discovery inventory
FR3	Evidence acquisition	Acquire supported browser and Windows artefacts	Acquired evidence
FR4	Evidence preservation	Preserve evidence; record provenance	Provenance records
FR5	Integrity verification	Calculate and verify hashes (e.g., SHA-256)	Integrity records
FR6	Evidence repository	Maintain case evidence and metadata	Evidence vault
FR7	Parsing and normalization	Map records to a common event model	Normalized events
FR8	Cross-browser correlation	Relate events across browsers and profiles	Relationships
FR9	Cross-artefact correlation	Relate browser and Windows artefacts	Relationships
FR10	Evidence quality	Assess integrity, provenance, completeness	Quality and coverage indicators
FR11	Corroboration	Identify evidence consistent with an activity	Corroboration matrix
FR12	Contradiction analysis	Flag potentially inconsistent relationships	Conflict alerts

ID	Component	Purpose	Principal output
FR13	Traceability	Link findings to source evidence	Evidence-to-finding chain
FR14	Anti-forensic review	Review evidence-disrupting conditions	Disruption indicators
FR15	Activity reconstruction	Group related events into episodes	Activities
FR16	Visualization	Timeline, graph and evidence views	Views; dashboard
FR17	Run reconciliation	Align records; retain gaps and conflicts	Reconciled evidence view
FR18	Hypothesis lab	Test investigator propositions	Hypothesis assessment
FR19	Reporting	Generate traceable reports	Forensic report
FR20	Audit and evaluation	Record actions; support evaluation	Audit trail

4.2 Evidence Lifecycle and Unified Evidence Model

The evidence lifecycle comprises thirteen stages: identification, acquisition, preservation, verification, storage, parsing, normalization, correlation, interpretation, reconstruction, validation, reporting and audit. Its early stages follow the identification, collection, acquisition and preservation activities described in standards and process models [18], [20]. The unified evidence model (Fig. 2) connects an evidence item (a preserved file, image, database, memory object or imported artefact) to artefacts, events, relationships, activities, findings and reports, with contextual entities attached through identifiers.

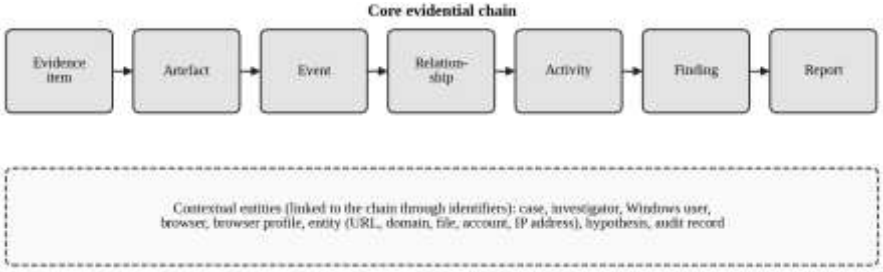


Fig -2: Unified evidence model: core evidential chain and contextual entities.

Let E denote evidence items, A artefacts, V normalized events, R relationships, Y reconstructed activities and F findings. The conceptual pipeline is

$$E \rightarrow A \rightarrow V \rightarrow R \rightarrow Y \rightarrow F \quad (1)$$

A normalized event is a tuple v = (t, b, p, u, k, x, s, a): timestamp in Coordinated Universal Time (UTC) with the original value retained, browser, profile, Windows user, event type, entity, source and artefact type. Relationships are established along the dimensions D = {temporal, browser, profile, user, entity, source, sequence, artefact type}:

$$R \subseteq V \times V \times D \quad (2)$$

Traceability requires that every reportable finding f maps to an activity, events, artefacts and evidence items, so that Trace(f) $\neq \emptyset$. No numerical weighting or probability model is imposed, since no empirical basis for such parameters is established here.

4.3 Evidence Acquisition, Preservation and Normalization

Acquisition and preservation. Evidence is collected by a documented procedure or imported with metadata, stored with minimal modification, and hashed so that integrity can be verified [10], [18]. Active browsers may lock databases and database sidecar files may hold records not yet merged into the main file; the acquisition record therefore states which files were captured and which sources were unavailable, protected or corrupted, for example values protected by application-bound encryption [4]. Artefact classes in scope are history, downloads, searches, cookies, cache, bookmarks, sessions and tabs, web storage, autofill, database sidecars, browser configuration, and selected Windows registry, event-log, Prefetch, link-file, name-resolution, resource-usage and memory artefacts, where available. *Normalization.* Records are parsed and mapped to the event tuple; because time conventions differ between browsers [1], timestamps are converted to UTC while retaining the original value and its source semantics, since artefacts may record different moments (creation, access or completion). Mapping rules must be maintained as browser versions change.

4.4 Cross-Browser and Cross-Artefact Correlation

Events are related along the eight correlation dimensions defined in Table 3 (Fig. 3). Cross-browser correlation applies the browser, profile, user, entity, temporal and sequence dimensions to events from different browsers; cross-artefact correlation applies the source and artefact-type dimensions to relate events within browser data and between browser and Windows data. For example, a search, a visit and a download, followed where available by a Windows file-creation or execution record, may be related as an ordered sequence. Each relationship records the dimension(s) on which it rests, so that criteria are transparent and open to review. Broad criteria can relate unrelated events; relationships are therefore candidate relationships.

Table -3: Correlation dimensions.

Dimension	Question	Illustrative example
Temporal	Do events occur within a relevant time relationship?	Events at 10:10, 10:12, 10:15
Browser	Do records come from the same or different browsers?	Chrome and Edge
Profile	Do records belong to the same browser profile?	Chrome “Default” profile
Windows user	Are events associated with the same operating-system context?	User A
Entity	Do records refer to the same URL, domain, file or account?	example.com

Dimension	Question	Illustrative example
Source	Do records originate from distinct sources?	History and download database
Sequence	Do events form an ordered sequence?	Search → visit → download
Artefact type	Do different artefact types describe the same activity?	History, cookie and download records

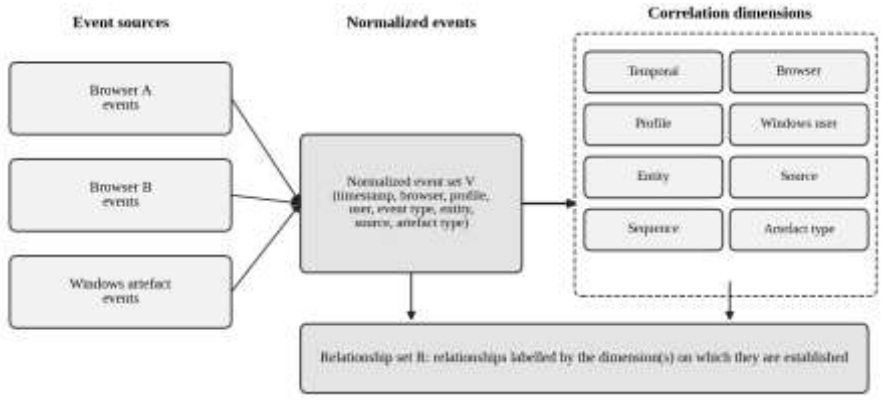


Fig -3: Cross-browser and cross-artefact correlation model.

4.5 Forensic Intelligence Module

The forensic intelligence module (Fig. 4, Table 4) is the analytical layer between correlated evidence and investigator-facing findings. "Intelligence" denotes evidence-aware analytical functions that support interpretation, not autonomous reasoning or legal conclusions. Each function is described below by purpose, input, analytical process, output, forensic value and limitations.

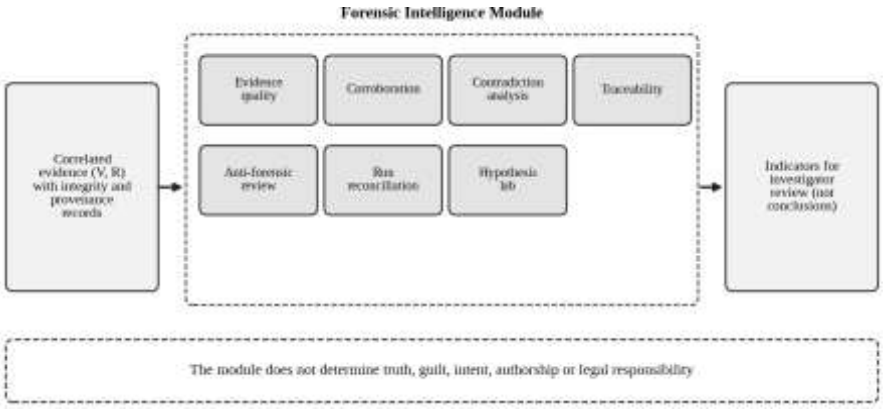


Fig -4: Forensic intelligence module: seven functions operating on correlated evidence and producing indicators for investigator review.

Table -4: Forensic intelligence functions.

Function	Question addressed	Output	Principal limitation
Evidence quality	What is the condition and provenance of this evidence?	Quality and coverage indicators	Condition, not accuracy of content
Corroboration	What other evidence is consistent with this activity?	Corroboration matrix and rationale	Consistency does not prove occurrence
Contradiction analysis	What evidence does not fit the current interpretation?	Conflict alerts	May flag legitimate differences
Traceability	Where did this finding come from?	Evidence-to-finding chain	Shows derivation, not correctness
Anti-forensic review	Which conditions may have reduced availability?	Disruption indicators	Absence is not proof of intent
Run reconciliation	Where are records aligned, missing or in conflict?	Reconciled evidence view	Depends on alignment criteria
Hypothesis lab	What supports, conflicts with or is missing for a proposition?	Hypothesis assessment	Decision support only

4.5.1 Evidence Quality (FR10)

Input: integrity, provenance, acquisition and parsing records. Process: each item and source receives indicators (integrity verified or not; acquired, partial or unavailable; parsed or not; timestamp available or not), compared with the discovery inventory so that expected but missing sources are listed. Output: quality and coverage indicators. Value: makes limits visible [12]. Limitation: indicators describe condition, not the accuracy of content.

4.5.2 Corroboration (FR11)

Input: relationships and quality indicators. Process: six forms of support are distinguished: single-source evidence; multi-artefact corroboration (distinct artefacts of one browser or profile); cross-browser corroboration; temporal, entity and

sequence corroboration. Records are described as potentially independent sources only where they derive from distinct artefact sources, since several records may stem from one user action or software event. Output: a corroboration matrix with rationale. Limitation: terms such as "supporting", "consistent" and "corroborating" evidence are used deliberately; any summary value, such as the share of supporting dimensions, is an ordinal indicator and must not be read as a probability that an activity occurred unless calibrated by validation. Methods for expressing evidential strength exist in general [13], [14] and are not replaced by this indicator.

4.5.3 Contradiction Analysis (FR12)

Input: relationships, quality indicators, provenance. Process: potential contradictions are identified along seven dimensions: temporal (an implausible order or interval), user or profile (related records attributed to different contexts), browser, entity (mismatching attributes such as name or hash), source (a record present in one source but absent where expected), sequence, and provenance (evidence whose integrity check failed or whose origin is unclear), in the spirit of timeline-inconsistency detection [15], [16]. Output: conflict alerts naming the evidence and the rule triggered. Limitation: the framework flags potential contradictions and the investigator evaluates them; conflicts are not resolved automatically, and legitimate differences (timestamp semantics, clock changes, time zones) may cause false flags.

4.5.4 Traceability (FR13)

Input: all derived objects and their links. Process: links are recorded at each transformation so that a finding connects to an activity, events, artefacts, evidence items and integrity and provenance records (Fig. 5(b)); a finding without such a chain is not reportable. Output: the evidence-to-finding chain. Value: transparency, auditability, reproducibility and clearer interpretation of the strength of each link; provenance-preserving derivation has been demonstrated for general timelines [11]. Limitation: traceability shows how a finding was derived, not that it is correct.

4.5.5 Anti-Forensic Review (FR14)

Input: coverage indicators, artefact status, relationships. Process: indicators of deletion, history or cache clearing, private browsing, profile removal and expected-but-missing artefacts are recorded with their basis, using anti-forensic categories as a reference [21], [22]. Output: potential disruption indicators. Limitation: missing evidence is not interpreted as intentional destruction.

4.5.6 Run Reconciliation (FR17)

An analysis run denotes processing of the evidence of a case. Records are aligned by time, entity, browser, profile, user and sequence; aligned records, gaps and conflicts are retained side by side, not merged, giving a reconciled view that includes what is missing. Limitation: quality depends on alignment criteria and timestamp semantics.

4.5.7 Hypothesis Lab (FR18)

The investigator formulates a proposition; relevant events are selected; supporting, conflicting and missing evidence is identified; and the proposition receives one of five categories: Supported, Partially Supported, Contradicted, Insufficient Evidence or Requires Review. Testing explicit propositions follows the hypothesis-based character of event-based investigation [19]. The investigator reviews and documents the decision. Limitation: the categories are decision support, their semantics must be defined and validated at implementation, and they do not establish truth, guilt, intent or authorship.

4.6 User-Activity Reconstruction

Low-level records are converted into activity episodes through the sequence raw artefact, normalized event, related events, activity episode and finding (Fig. 5(a)). Events are grouped into one episode when they are related on defined dimensions, for example common browser, profile and Windows user, a common entity and a plausible temporal and sequential relationship; for instance, a search, a visit to a result page and a download recorded at 10:10, 10:12 and 10:15 in one profile may form one episode with each supporting record listed. Grouping thresholds (permissible intervals, inactivity thresholds) are configurable parameters to be defined, justified and evaluated at implementation; specific values are not established in the examined sources. A reconstructed activity describes what the evidence indicates occurred on the system and is not equated with intent, identity, authorship or criminal responsibility unless independently established.

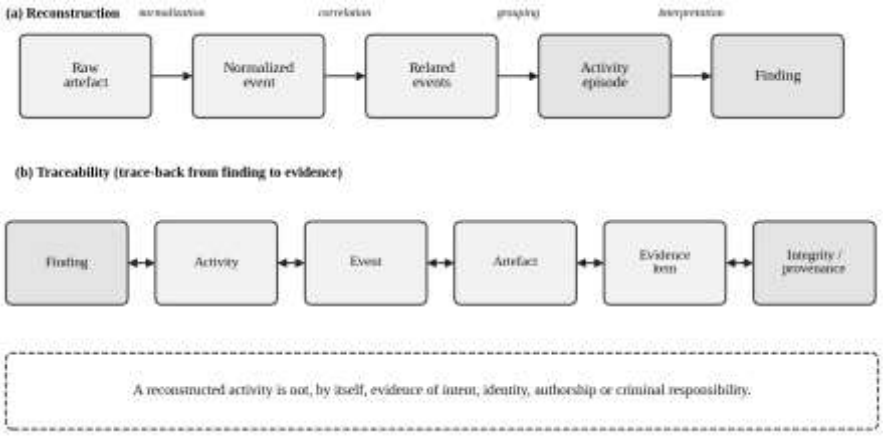


Fig -5: (a) User-activity reconstruction process and (b) evidence-to-finding traceability model.

4.7 Visualization and Reporting

Timeline, graph and evidence views present events, relationships and activities with links back to evidence. Reports carry, for each finding, its evidence chain, quality indicators, corroboration, conflict alerts and disruption indicators, and state unavailable evidence. Framework outputs are the evidence inventory and coverage status, integrity and provenance records, normalized events, relationships, corroboration matrix, conflict list, disruption review, reconciled view, activities, hypothesis assessments, traceability chains, the report and the audit trail.

5. PROPOSED EVALUATION METHODOLOGY

This section specifies how the framework should be evaluated; no experimental results are reported. The procedure is shown in Fig. 6 and the scenarios and measures in Table 5.

5.1 Objectives, Environment and Ground Truth

The evaluation assesses acquisition completeness, parsing correctness, correlation, reconstruction accuracy, corroboration and contradiction detection, traceability, reproducibility, usability and performance. A controlled environment of clean, snapshot-restorable Windows test systems is proposed, with browser families, versions, profiles and user accounts recorded before each scenario, system clocks synchronized and documented, and the configuration of Windows corroborating sources recorded. Ground truth is produced by scripted user actions with an independent action log, withheld from analysis until framework output is produced. Expected evidence is confirmed by manual examination, not assumed, because persistence varies with browser version and configuration; some scenarios should be designed without reference to the framework's correlation rules to limit evaluation bias.

5.2 Scenarios, Metrics and Validation Procedure

Table -5: Proposed evaluation scenarios and measures.

ID	Scenario (example)	Evaluation purpose	Primary measures
S1	Normal browsing (search → visit)	Baseline evidence, parsing	Parsing correctness; completeness
S2	Multiple browsers	Cross-browser correlation	Correlation precision/recall
S3	Multiple profiles	Profile separation	Correlation precision/recall
S4	Download	History–download correlation	Correlation precision/recall
S5	Cookie interaction	Multi-artefact corroboration	Corroboration precision
S6	Private browsing	Evidence availability	Acquisition completeness
S7	History deletion	Evidence disruption	Disruption indicator detection
S8	Cache clearing	Persistence gaps	Acquisition completeness
S9	Profile deletion	Residual evidence	Acquisition completeness
S10	Mixed activity	End-to-end reconstruction	Reconstruction accuracy; traceability completeness; contradiction detection; time; reproducibility; usability

Note. Measures follow the specification; no target values are proposed.

Precision is the proportion of reported items present in the ground truth, and recall the proportion of ground-truth items reported. Acquisition completeness is the proportion of expected artefacts acquired; traceability completeness is the proportion of findings with a complete chain to preserved evidence; reproducibility is the proportion of repeated runs over the same evidence yielding identical output; usability is assessed by task time, errors and user assessment. For each scenario the framework is executed, its output compared with ground truth, and errors analyzed (parsing gaps, over-broad criteria, timestamp semantics). Results should be compared with a manual baseline and with established tools on the same evidence, true- and false-positive definitions fixed before analysis, verification performed by more than one examiner with agreement reported, and repetitions and statistical analysis (for example confidence intervals) specified in advance. Usability studies require ethical approval and informed consent.

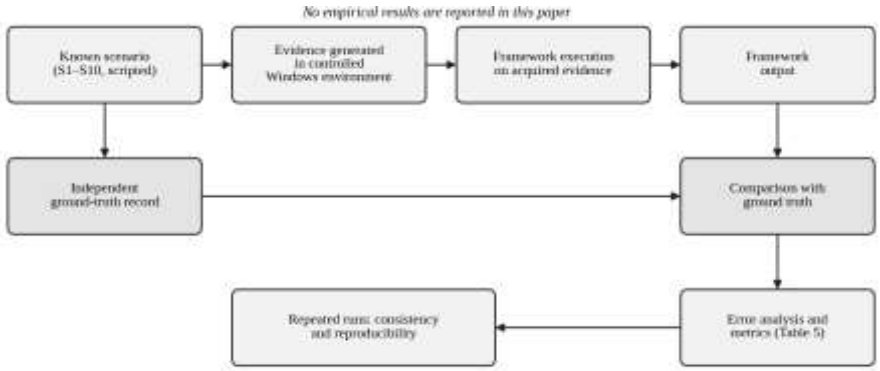


Fig -6: Proposed evaluation methodology. No empirical results are reported in this paper.

5.3 Reproducibility

Browser, operating-system and tool versions, scenario scripts, framework configuration and correlation parameters should be recorded and, where legally and ethically permissible, published with hashes of the evidence and ground-truth records. Rules should be deterministic where possible, the audit trail of each run preserved, and test data free of real personal data.

6. DISCUSSION

6.1 Interpretation

The framework addresses the fragmentation of browser evidence across browsers, profiles, artefacts and Windows sources, and the manual effort of judging completeness, support and conflict. It responds to the gaps in Table 1 by integrating functions that are usually treated separately, and consistently favours transparency: relationships carry their basis, support is described qualitatively, conflicts are flagged and not resolved, and reconstruction is separated from attribution. What remains unvalidated is considerable: that the correlation criteria give acceptable precision and recall, that contradiction rules avoid

excessive false flags, that corroboration indicators help rather than mislead, and that effort is reduced. The framework is therefore a hypothesis to be tested by the methodology in Sec. 5.

6.2 Relationship to Existing Approaches

Table 6 relates the framework's capabilities to the examined literature. It reflects only these sources, is not a systematic comparison, and supports no claim of superiority. Several capabilities are partly addressed in earlier work, such as integrated multi-browser analysis [1], provenance-preserving event abstraction [11] and timeline-inconsistency detection [15]; the framework extends these ideas by combining them for Windows browser evidence.

Table -6: Comparison dimensions with existing approaches.

Capability	Evidence in examined sources (status)
Multi-browser	Addressed for browsers of its period [1]; browsers on Windows 10 for one application [9]
Cross-artefact	General event abstraction [11]; not browser-to-Windows architecture
Acquisition and normalization	Described [1], [10], [18]; time formats harmonized [1]
Completeness; corroboration	Not established for browser evidence; general uncertainty and strength-of-evidence work [12]–[14]
Contradiction	Partly, for general timelines [15], [16]
Traceability; visualization	Partly, for general timelines [11]
Activity reconstruction	Addressed; field fragmented [17]
Anti-forensics	Taxonomies [21], [22]; residual artefacts [6], [7]
Hypothesis analysis	General hypothesis-based investigation [19]; browser-specific not established
Validation	Individual experimental designs [6], [7]; no common benchmark identified

6.3 Contributions

(i) *Conceptual:* an evidence-aware architecture and unified evidence model. (ii) *Methodological:* an eight-dimension correlation model, qualitative corroboration and contradiction semantics, a hypothesis workflow and an evaluation methodology. (iii) *Forensic:* explicit treatment of coverage, unavailable sources and evidence-disrupting conditions, and a traceability requirement that separates interpretation from evidence. (iv) *Practical:* a specification to guide implementation and a common basis for comparing future tools. None is validated, and components are not claimed to be novel in isolation.

6.4 Practical, Forensic and Ethical Implications

If implemented and validated, the framework could reduce manual cross-referencing and make coverage and uncertainty explicit in reports; this requires empirical confirmation. Preservation, hashing, provenance and audit components support documentation of handling consistent with established guidance [10], [18], while adherence to a specific procedure remains the practitioner's responsibility. Browser data are highly personal and may concern third parties, so acquisition and analysis should be limited to what is authorized and relevant, with case isolation and restricted access. Requirements for admissibility and lawful acquisition differ between jurisdictions; no jurisdiction-specific claim is made. Corroboration indicators, conflict alerts and hypothesis assessments are decision support, and the investigator remains responsible for interpretation.

7. LIMITATIONS AND THREATS TO VALIDITY

The principal limitation is that the framework has not been implemented or evaluated. Further limitations are summarized in Table 7.

Table -7: Limitations and threats to validity.

Limitation or threat	Implication	Proposed mitigation
No implementation or validation	Effectiveness and benefit unknown	Prototype; apply Sec. 5
Browser version changes; artefact heterogeneity	Parsers and mappings may become outdated	Version-aware parsers; recorded versions
Encryption; evidence loss; private or deleted browsing	Evidence may be absent or partial [4], [6], [7]	Record unavailable sources; no inference of intent from absence
Timestamp differences	Artefacts record different moments and conventions [1]	Retain original values and semantics
False correlations; false contradiction flags	Unrelated events related; legitimate differences flagged	Transparent criteria; measure precision; investigator review
Ground-truth limits; evaluation bias	Scenarios may not reflect real-world complexity	Independent logs; broader scenarios; ethical case studies
Limited coverage of Windows artefact semantics in cited sources	Artefact-specific behaviour not established here	Verify with primary sources at implementation
Framework-level security not specified	Repository and audit tamper-resistance undefined	Define at implementation

8. FUTURE RESEARCH

Future work comprises implementing a prototype; building controlled, reproducible benchmark datasets; validating correlation and reconstruction against ground truth; evaluating and calibrating corroboration and contradiction functions; usability, performance and scalability studies; extending browser and Windows artefact coverage; comparison with established forensic tools; controlled case studies under ethical safeguards; and investigating explainable machine learning only where it provides measurable forensic value.

9. CONCLUSION

Windows browser evidence is fragmented across browsers, profiles, artefact types and operating-system sources, and isolated analysis can leave completeness, support, inconsistency and traceability unaddressed. This paper proposed an evidence-aware framework that carries evidence from acquisition and preservation, through a unified evidence model and cross-browser and cross-artefact correlation, to a forensic intelligence module, activity reconstruction and traceable reporting, without determining truth, intent, identity or authorship. Its principal contribution is the integration of these functions into a coherent, testable architecture that responds to the gaps supported by the examined evidence, and a controlled ground-truth evaluation methodology was specified. Because no implementation or empirical evaluation has been performed, effectiveness remains unproven and empirical validation is necessary.

DECLARATIONS

Ethics: no human participants or personal data. **Competing interests:** none declared. **Funding:** none reported. **Data availability:** no datasets were generated or analyzed. **AI assistance:** [to be completed by the author per journal policy].

REFERENCES

[1] J. Oh, S. Lee, and S. Lee, "Advanced evidence collection and analysis of web browser activity," *Digit. Investig.*, vol. 8, suppl., pp. S62–S70, 2011, doi: 10.1016/j.diin.2011.05.008.

[2] M. T. Pereira, "Forensic analysis of the Firefox 3 Internet history and recovery of deleted SQLite records," *Digit. Investig.*, vol. 5, no. 3–4, pp. 93–103, 2009, doi: 10.1016/j.diin.2009.01.003.

[3] B. Jeong, S. Lee, and J. Park, "MIC: Memory analysis of IndexedDB data on Chromium-based applications," *Forensic Sci. Int., Digit. Investig.*, vol. 50, Art. no. 301809, 2024.

[4] Google Security Blog, "Improving the security of Chrome cookies on Windows," Jul. 30, 2024. [Online]. Available: https://security.googleblog.com/2024/07/improving-security-of-chrome-cookies-on.html

[5] G. Aggarwal, E. Bursztein, C. Jackson, and D. Boneh, "An analysis of private browsing modes in modern browsers," in *Proc. 19th USENIX Security Symp.*, Washington, DC, USA, Aug. 2010.

[6] D. J. Ohana and N. Shashidhar, "Do private and portable web browsers leave incriminating evidence? A forensic analysis of residual artifacts from private and portable web browsing sessions," *EURASIP J. Inf. Secur.*, vol. 2013, Art. no. 6, 2013, doi: 10.1186/1687-417X-2013-6.

[7] M. Muir, P. Leimich, and W. J. Buchanan, "A forensic audit of the Tor Browser Bundle," *Digit. Investig.*, vol. 29, pp. 118–128, 2019, doi: 10.1016/j.diin.2019.03.009.

[8] G.-Y. Choi, J. Bang, S. Lee, and J. Park, "Chracer: Memory analysis of Chromium-based browsers," *Forensic Sci. Int., Digit. Investig.*, vol. 46, suppl., Art. no. 301613, 2023, doi: 10.1016/j.fsidi.2023.301613.

[9] F. Iqbal, Z. Khalid, A. Marrington, B. Shah, and P. C. K. Hung, "Forensic investigation of Google Meet for memory and browser artifacts," *Forensic Sci. Int., Digit. Investig.*, vol. 42, Art. no. 301448, 2022, doi: 10.1016/j.fsidi.2022.301448.

[10] K. Kent, S. Chevalier, T. Grance, and H. Dang, "Guide to integrating forensic techniques into incident response," National Institute of Standards and Technology, Gaithersburg, MD, USA, NIST Spec. Publ. 800-86, 2006, doi: 10.6028/NIST.SP.800-86.

[11] C. Hargreaves and J. Patterson, "An automated timeline reconstruction approach for digital forensic investigations," *Digit. Investig.*, vol. 9, suppl., pp. S69–S79, 2012, doi: 10.1016/j.diin.2012.05.006.

[12] E. Casey, "Error, uncertainty, and loss in digital evidence," *Int. J. Digit. Evid.*, vol. 1, no. 2, pp. 1–45, 2002.

[13] E. Casey, "Standardization of forming and expressing preliminary evaluative opinions on digital evidence," *Forensic Sci. Int., Digit. Investig.*, vol. 32, Art. no. 200888, 2020.

[14] G. Horsman, "Digital evidence certainty descriptors (DECDs)," *Forensic Sci. Int., Digit. Investig.*, vol. 32, Art. no. 200896, 2020.

[15] A. Marrington, I. Baggili, G. Mohay, and A. Clark, "CAT Detect (Computer Activity Timeline Detection): A tool for detecting inconsistency in computer activity timelines," *Digit. Investig.*, vol. 8, suppl., pp. S52–S61, 2011.

[16] L. M. Dreier, C. Vanini, C. J. Hargreaves, F. Breitinger, and F. Freiling, "Beyond timestamps: Integrating implicit timing information into digital forensic timelines," *Forensic Sci. Int., Digit. Investig.*, vol. 49, Art. no. 301755, 2024, doi: 10.1016/j.fsidi.2024.301755.

[17] F. Breitinger, H. Studiawan, and C. Hargreaves, "SoK: Timeline based event reconstruction for digital forensics: Terminology, methodology, and current challenges," *Forensic Sci. Int., Digit. Investig.*, vol. 53, Art. no. 301932, 2025, doi: 10.1016/j.fsidi.2025.301932.

[18] ISO/IEC 27037:2012, "Information technology—Security techniques—Guidelines for identification, collection, acquisition and preservation of digital evidence," International Organization for Standardization, Geneva, Switzerland, 2012.

[19] B. Carrier and E. H. Spafford, "An event-based digital forensic investigation framework," in *Proc. Digit. Forensic Res. Workshop (DFRWS)*, 2004.

[20] B. Carrier and E. H. Spafford, "Getting physical with the digital investigation process," *Int. J. Digit. Evid.*, vol. 2, no. 2, 2003.

[21] R. Harris, "Arriving at an anti-forensics consensus: Examining how to define and control the anti-forensics problem," *Digit. Investig.*, vol. 3, suppl., pp. S44–S49, 2006, doi: 10.1016/j.diin.2006.06.005.

[22] K. Conlan, I. Baggili, and F. Breitinger, "Anti-forensics: Furthering digital forensic science through a new extended, granular taxonomy," *Digit. Investig.*, vol. 18, suppl., pp. S66–S75, 2016, doi: 10.1016/j.diin.2016.04.006.