RSW-0013 — Versioning and Updates

Revision 0.01 | Protocol rsw/1 | Format 1 | Experimental proposal

Developed by Pablo Murad — November 23, 2026 — pablo@pablomurad.com — https://rsw.works

Experimental Project Specification. This document is an experimental specification of the Really Small Web project. It is a work in progress and has not been published as an RFC or approved as an Internet Standard.

License: specifications CC0 1.0 (public domain); future reference implementation and official tools MIT.

RSW-0013 — Versioning and Updates

Revision 0.01 | Protocol rsw/1 | Format 1

MVP — experimental normative proposal

Developed by Pablo Murad

Pablo Murad — November 23, 2026

pablo@pablomurad.com

MUST and MUST NOT indicate requirements of the described profile; SHOULD indicates a recommendation; MAY indicates an option. Experimental document, not yet ratified or implemented. The rules of RSW-0001 through RSW-0003 remain applicable.

01. Scope

Mandatory for the MVP. Defines immutable versions, the local publication head, rollback, conflicts, and history. A stable identity is (A, name) from RSW-0003. version is the RSW-0002 integer; H_png identifies exact bytes. There is no global consensus, reliable clock, or automatic identity transfer.

02. Editions and Equivalence

First edition version=1, without previous_release. An edition v>1 includes previous_release = H_png of the declared previous edition. The producer SHOULD generate v+1 referencing the exact object of the chosen previous edition, without overwriting a published PNG. The accepted head records (version, H_png, SHA-256(JCS(manifest)), chain state). Multiple packagings of the same edition can have the same canonical manifest and a valid signature, but a distinct H_png. Classify as EQUIVALENT_EDITION if identity, version, and C are identical and both are fully valid; store the hashes separately, do not silently replace the already chosen head. Because previous_release is an exact hash, equivalence does not replace the predecessor object in chain-of-custody proof.

If the same identity and version have distinct C, classify as VERSION_CONFLICT, even if the apparent title or content is equal. Preserve the evidence and the already accepted head; if there is no head, require an explicit object choice for reading, without automatically promoting one. Do not decide by timestamp, lexicographically smaller hash, or first relay. A historical choice does not resolve a global conflict.

03. Promotion Algorithm

Verify all RSW-0001..0003 rules, identity, and expected H_png. Unsigned objects do not participate in an authenticated head. For a first observation without a head, accept the valid version indicated by an accepted index or explicit import; if v>1, report an unverified chain, without claiming it started at 1. If candidate v < largest accepted, classify as ROLLBACK: it may be archived/opened explicitly, not promoted. Equal v: idempotent, equivalent, or conflict per section 02. Greater v: if the predecessor is available, it must have the same identity and version=v-1; otherwise INVALID_PREDECESSOR. Traverse only the local chain, with a recommended limit of 256 steps and cycle detection; each edge compares the exact H_png. If it reaches the known head through a contiguous chain without conflict, promote. If there is a gap/unavailable predecessor, do not download automatically; present UNVERIFIED_CHAIN and require explicit confirmation to promote from an authenticated index or explicit import. If the known chain diverges from the head (including another packaging hash of an equivalent edition), present BRANCH_CONFLICT and require an explicit decision;

never present the chain as continuous. A confirmation may accept a new higher-version head, but must preserve an indication of an unproven chain or conflict and the previous history. Without confirmation, the head remains.

04. Local State and Indexes

Store the largest accepted version per identity atomically and monotonically, even if the user opens or removes a historical object from the cache. Persist the metadata needed to detect regression, without necessarily keeping all bytes. Cache clearing does not erase the high-water mark; an explicit identity reset warns about loss of protection. An index by domain has its own sequence per RSW-0006 and does not redefine the publication order. An entry removed from the index does not revoke copies. A key change is another identity (RSW-0003/0015). Do not promote an index newer than an edition it references as a smaller head.

05. Historical Navigation

The URI ?v=n (RSW-0004) requires edition n; in conflict, ask for H_png. The SHA-256 URI requires the exact object. Opening an old copy does not reduce the high-water mark, does not automatically restore permissions, and does not require network. Indicate version, hash, last index consultation, known chain, and conflicts. Offline mode presents only the known state. Do not claim "latest version" without qualification. Sharing a historical edition must preserve the original bytes.

06. Vectors and Tests

Cases: v1; v2 with v1 predecessor; v3 with a gap; v3 with a predecessor from another key; the same edition repackaged; the same edition with a different manifest; v1 rollback after v2; index with a higher sequence but older edition; absent historical file; cleared cache with high-water mark; key change; two concurrent sources. Test interruption during writing, concurrency, and atomic recovery. Pending: implement the conflict policy, real fixtures, and validate the UX. References: RSW-0002/0003/0004/0005/0006/0010/0015.

License: CC0 1.0 (public domain dedication; see CC0-1.0.txt in the downloads area). Developed by Pablo Murad | pablo@pablomurad.com END — RSW-0013

Author's Address

Pablo Murad
Independent
pablo@pablomurad.com
https://pablomurad.com
Official project website: https://rsw.works

Small files. Independent worlds.