

Position Paper: RCT Repaired Change Tracking for ODF 1.2

Dennis E. Hamilton
Interoperability Architect
4401 44th Ave SW
Seattle, WA 98116, USA
+1-206-779-9430
dennis.hamilton@acm.org

ABSTRACT

Repaired Change-Tracking (RCT) is an extension for OpenDocument that remains compliant to the OASIS ODF 1.2 Standard. This position paper relates the design point and benefits, especially interoperability trade-offs, behind the design of RCT.

Categories and Subject Descriptors

H.4.1 [Information Systems Applications]: Office Automation – *word processing*; D.2.7 [Software Engineering]: Distribution, Maintenance, and Enhancement – *portability; restructuring, reverse engineering, re-engineering*; D.2.9 [Software Engineering]: Management – *life cycle*; H.1.2 [Models and Principles] User/Machine Systems – *human factors*, I.7.1 [Document and Text Processing]: Document and Text Editing – *version control, document management*

General Terms

Design, Management, Standardization

Keywords

OpenDocument, RCT, change tracking, fitness, WYSIWYG.

1. INTRODUCTION

Repaired Change Tracking (RCT) is an extension profile for change-tracking provisions in OpenDocument format (ODF) [1].

The distinguishing features of the approach are set forth in terms of design point (section 2), benefits (section 3), and challenges (section 4).

The author is wary of systems that present changes as operations on the starting document rather than operations that can be made to revert the ending document to earlier forms. Those concerns are presented for contrasting consideration (section 5).

2. DESIGN POINT

With RCT, just as for current OpenDocument documents, the ODF format represents a changed document, d_n , with accompanying tracked-changes, -deltas, that can be manifest as the current document with in-line presentation of changes. Rejection of individual changes revert portions of the document to some also-presented earlier form. Rejection of all changes reduces the document to d_0 , the original starting document.

This is in contrast to systems that represent, still using ODF representation, the starting document, d_0 , along with separate +deltas. In that case, acceptance of changes bring portions of d_0 to later forms. Acceptance of all changes delivers d_n .

Manifestation of d_n tracked changes is the default behavior of prevalent office-productivity software. It is the normal starting point for further work on an in-progress document. It is the expected starting point for review and acceptance/rejection of changes. RCT satisfies that expectation by design.¹

3. RCT BENEFITS

3.1 OpenDocument Version 1.2 Compliance

RCT-profiled documents are OpenDocument Extended Documents. There are no breaking changes with respect to ODF 1.2 [2]. There is no change to ODF 1.2 schemas.

The RCT Profile constrains producers in how ODF 1.2 change-tracking is produced, using extended attributes that can be completely or entirely ignored by an OpenDocument Consumer.

RCT is built on what already works to the extent that it already works. It is an improvement of what is already specified.

3.2 Interoperability

Interoperability with ODF-native applications using tracked-change documents is not changed. Interchange with consumers supporting some level of RCT profile is enhanced.

Interchange with down-level consumers loses RCT-enabled coherence in the changes. Tracked changes are not lost, just seen as more granular. Some requirements around healing markup beyond the boundaries of insertions and deletions will not be satisfied. Down-level defects in treating those changes, if at all, are whatever they already are. RCT-aware applications receiving down-level OpenDocument documents operate as well as they can, just as down-level consumers now must.

There are already-successful, however-limited implementations of conversion to other tracked-change file formats. RCT can enhance how well documents are exported and may also improve how tracked-changes in other forms are brought into OpenDocument form. RCT is positioned to enhance inter-conversion with Office Open XML (OOXML) tracked-changes as much as possible within the constraints of OpenDocument.

3.3 Robustness

Defects in tracked-change information, including untracked modifications or damage to the d_n representation, have local effects on manifestation of the document

Degradation that arises when importing imperfectly-produced tracked-change documents tends toward preserving as much as possible. Degradation is not brittle nor more unacceptable than it

¹ <http://nfoWorks.org/rct/>

already is. There is a rather Hippocratic “do no more harm” condition.

The resiliency of in-stream markers connecting d_n to its $-$ deltas is inestimable. The connection is immutable and insensitive to movement of the marker position as the result of further changes.

Markers and the deltas can also be used as tombstones for preserving provenance and change history of accepted and rejected changes in possible extension beyond essential RCT.

It is an invariant of all approaches that implementations could have more arrangements to assist users in noticing conflicts and degradation at a point where meaningful corrective-action is available to the user.

3.4 Deployment

RCT buys time. For many situations, it may be good enough for a very long time. The practical advantage of incremental, tested, and interoperability-confirmed upgrades is protection of the growing corpus of OpenDocument documents. Interoperability preservation respects the diversity of software and platforms at all levels that are already used for producing and accepting OpenDocument documents. There is no requirement to force changes or obsolete software. The RCT benefits are elective where and when appropriate implementations are available.

4. CHALLENGES

The basic challenge for RCT is the limited headroom that there may be for repairs and extended cases while retaining compliance with the OpenDocument format and the relaxed conformance requirements of the ODF standard. To achieve greater function and interoperable integrity, more aggressive departures into extensions are likely required.

5. THE $-$ DELTA $+$ DELTA TRADE-OFF

Differences that arise with an ODF 1.2 tracked-change model that is incompatible with the current d_n $-$ deltas model can be surmised by recasting the RCT design point and claimed benefits in terms of the alternative model. That’s straightforward

There is an additional consideration that was paramount in clinging to d_n $-$ delta for RCT. That is the brittleness of d_0 $+$ delta schemes that do not, practically cannot, use markers in the d_0

document XML representation to which they apply. This author is fearful of the related complexities and their fragility.

- Pointing from $+$ deltas into specific places in the XML representation is very difficult to accomplish in a manner that does not depend on some structural aspects of the path to the target being immutable. Consequently, many $+$ deltas must be updated correctly when new ones are introduced that break that assumption.
- The $+$ deltas have a partial ordering that has some of them apply to d_i , $0 < i < n$, that have to be reconstructed reliably as the $-$ deltas are dealt with. And those changes impact how other $+$ deltas locate where they have the correct effect.
- Once $+$ delta tracking has started, it cannot be suspended and resumed without having to adjust the impacted deltas. It mustn’t completely be suspended.
- There’s almost no practical defense against a defect introduced into d_0 being caught, and the consequences on the $+$ deltas can be global and catastrophic, though perhaps not so exciting as when a progressive decompression algorithm encounters a defect in its stream.

The correct, independent implementation of algorithms for these operations requires great skill and determination to ensure correctness. It would be fascinating to have a computationally appealing robust solution in hand. RCT is based on not waiting.

6. REFERENCES

- [1] Dennis E. Hamilton. Tracked Changes: Navigating the Document-Format Anti-Pattern. DChanges 2014, September 16, 2014, Fort Collins, Colorado, USA. To Appear.
- [2] OASIS. *Open Document Format for Office Applications (OpenDocument) Version 1.2 Part 1: OpenDocument Schema*. 29 September 2011 OASIS Standard. Available at <http://docs.oasis-open.org/office/v1.2/os/>