

# A Collaborative Visual Database

*by*

Imed Adel

A document submitted in partial fulfillment of the requirements for the

degree of

*Technical Report*

at

MISKATONIC UNIVERSITY



# **I      Conceptual study**

The deciding factor for the success of one application and the failure of another is the architecture. In order to achieve the requirements we set in the previous chapter, along with solving the issues faced by the existing solutions, we need to carefully design our systems.

In this chapter, we will first dive into the realm of real-time collaboration and read into the existing theoretical research in the field and its practical applications in the real world. Then, we will explore the different architectures for building and scaling our application. Finally, we will present the sequence and class diagrams to better define the implementation of our ideas and the trajectory of our work in the following chapter.

## **1.1   Real-time collaboration**

Real-time collaboration is a type of collaboration used in editors and web applications with the goal of enabling multiple users on different computers or mobile devices to modify the same document with automatic and nearly instantaneous merging of their edits. The document could either be a computer file, stored locally, or a cloud-stored data shared over the internet, such as an online spreadsheet, a word processing document, a database, or a presentation.

Multiple web applications support real-time collaboration under various names. Microsoft, for example, refers to it as “co-authoring” and offers it as part of its

Microsoft Office bundle, including Word, Excel, and PowerPoint [2]. Google Docs is another notorious contender in the space of collaborative editing, with products such as Google Docs and Google Sheets.

The interest in collaborative software has seen a resurgence since 2020, mainly due to the move to remote work, with companies like Microsoft offering ready-to-use APIs to enable this feature.

Real-time collaboration is different from other offline or delayed collaborative approaches, such as Git. While real-time editing performs automatic, frequent, or even instantaneous synchronization of data between all the connected users, offline editing requires manual submission, merging, and resolution of editing conflicts.

### 1.1.1 History

In 1968, Douglas Engelbart introduced the first collaborative real-time editor in a presentation named “The Mother of All Demos”, which also demonstrated many other fundamental elements of modern personal computing including windows, hypertext, graphics, video conferencing, the computer mouse, word processing, and revision control [3].

It took many decades for collaborative software to become mainstream. One of the first editors that offered collaborative capabilities was Writely, a collaborative word processor launched in 2005 [1]. It was later acquired by Google and renamed to Google Docs [5]. Another Google-backed product from the same era was Google Wave, a collaborative email software. However, less than a year later, it was discontinued due to a lack of users [4].

Since 2010, the number of web-based collaborative software has been exploding with successful examples such as Figma and Notion. Figure 1.1 shows the resurgence of products with real-time collaboration.

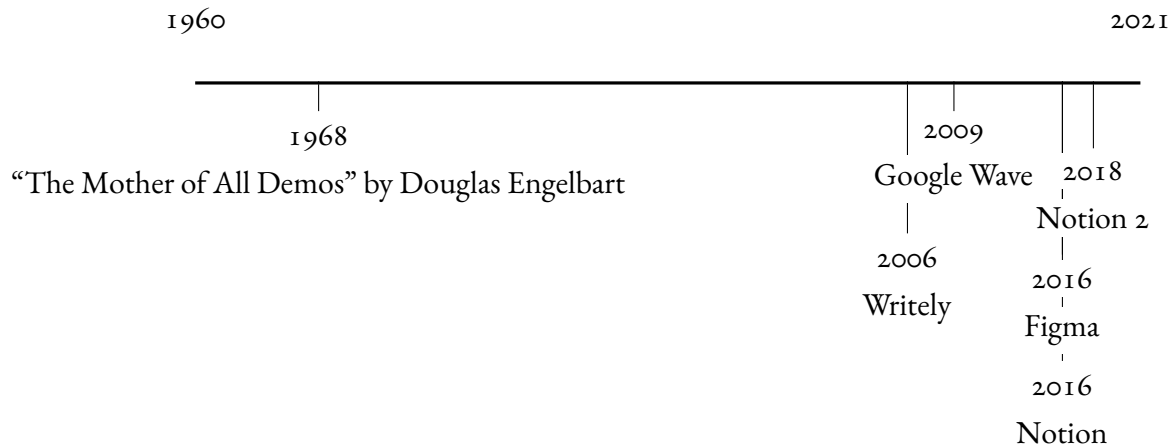


Figure 1.1: Timeline of real-time collaboration

### 1.1.2 Theoretical grounds

The main challenge of real-time collaboration is keeping multiple clients in sync. An algorithm has to be employed to determine how to apply—often conflicting—edits from the different remote users. Network latency is the main culprit for such a dilemma as modifications can reach the other clients with a certain amount of delay. Figure 1.2 shows a common example of conflicting changes by two users caused mainly by network latency.

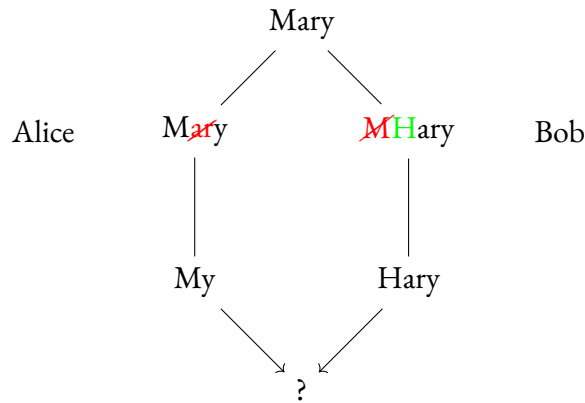


Figure 1.2: Example of a common problem in real-time collaboration

Over the years, two main algorithms emerged for dealing with real-time collaboration, Conflict-free Replicated Data Type (CRDT) and Operational Transformation (OT). Both have their benefits and drawbacks, and each one has numerous variations and implementations.

### Operational Transformation (OT)

Operational Transformation was invented for supporting real-time co-editors in the late 1980s and has evolved to become a collection of core techniques widely used in today's working co-editors and adopted in major industrial products [7]. Google Docs has been using OT since at least 2009 [8].

The model of Operational Transformation works by calculating all possible transformations for a text and using them to transform received changes according to the local state, thus eliminating any possible mistakes of synchronization. Figures 1.3 and 1.4 explain the algorithm followed by OT for resolving conflicts. Figure 1.5 illustrates the solution proposed by OT for the problem proposed in 1.2.

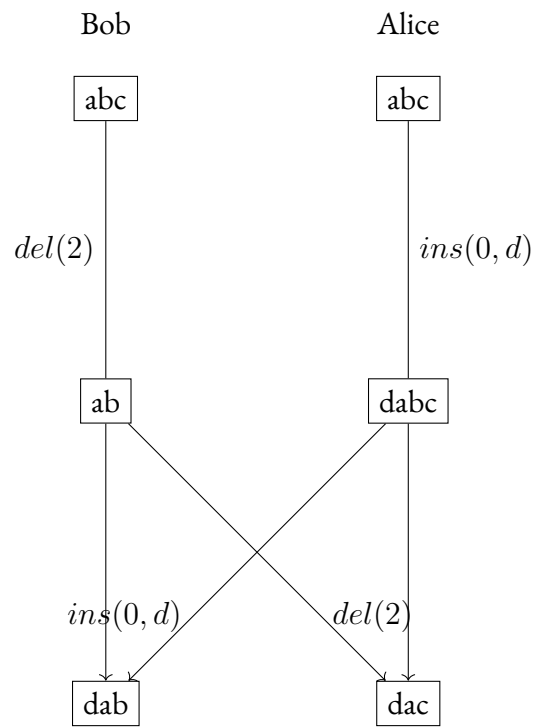


Figure 1.3: Without OT

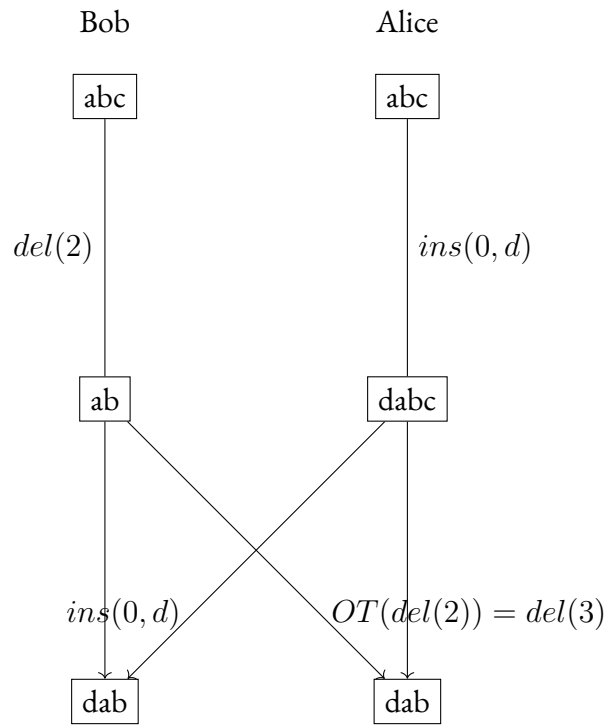


Figure 1.4: With OT

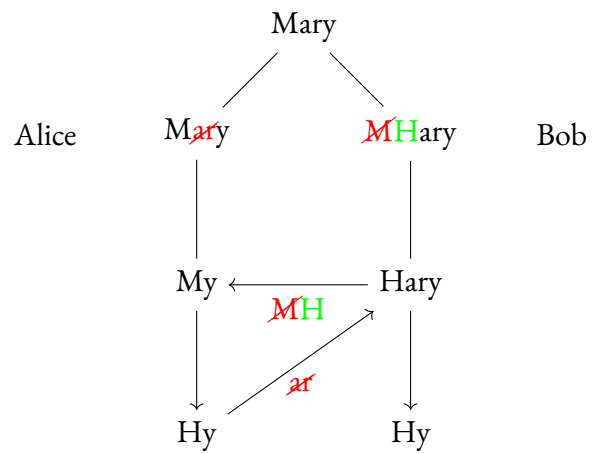


Figure 1.5: OT's solution for the problem in figure 1.2



As seen in the previous examples, OT can be hard and costly to set up, even for text-based documents. When it comes to more complicated and nested data structures, OT is rarely the first choice.

## **CRDT**

Conflict-free Replicated Data Type is a data structure that was first proposed around 2006 under the name of WithOut Operational Transformation (WOOT) [7]. It has the benefit of being able to be duplicated over numerous computers in a network, where the replicas can be updated independently and concurrently without requiring coordination, and where inconsistencies can always be resolved mathematically [6].

In 2011, CRDT was formally defined [6], and while it was initially developed for collaborative text editing, it was eventually adopted for multiple other use cases such as online chat systems and distributed databases.

CRDT is further subdivided into two types based on its implementation—Commutative Replicated Data Type (CmRDT) and Convergent Replicated Data Type (CvRDT). Both of them offer the same real-time capabilities but differ in their design and approach to the concept.

**CmRDT** Commutative Replicated Data Type or Operation-based CRDT transmits the local state only by sending the update operations required to reach that state. Upon receiving these operations, remote clients must apply them to become in sync with the sender. The transmission server has to avoid duplicating the operations as this algorithm is not idempotent.

**CvRDT** Convergent Replicated Data Type or State-based CRDT sends over the whole local state to be merged with the receiving clients' states. This method tends to be slow due to the need of sending large amounts of data instantaneously.

neously over the network. However, the infrastructure does not have to deal with deduplication as it is the case with CmRDTs.

### 1.1.3 Practical examples

In practice, web applications do not adhere to one algorithm. Instead, they draw from different concepts to a solution that perfectly fits their needs.

#### **Figma**

Figma is one of the first web design tools with real-time collaboration. It was released in 2016 and since then it has been hailed as one of the best examples of performant and collaborative software.

#### **Excalidraw**

Excalidraw is an open-source white-board and wireframing web application.

#### **Automerger**

Automerger is a JavaScript data structure designed specifically for collaborative software.

#### **Centige**

Centige is a discontinued collaborative web app builder by the author. It was mainly a learning project, started in late 2019 and developed throughout 2020.

**1.1.4 Conclusion**

**1.2 Design patterns**

**1.2.1 Common patterns**

**1.2.2 Comparison**

**1.2.3 Conclusion**

**1.3 Detailed architecture**

**1.4 Class diagrams**

**1.5 Sequence diagrams**

**1.6 Activity diagrams**

**1.7 Conclusion**

## **2** Implementation

Researching the current solutions led us to formulate a set of requirements to ensure that Merebase offers the best experience.

# Acronyms

**CmRDT** Commutative Replicated Data Type

**CMS** Content Management Software

**CRDT** Conflict-free Replicated Data Type

**CvRDT** Convergent Replicated Data Type

**FDD** Feature-Driven Development

**OT** Operational Transformation

**RBAC** Role-Based Access Control

**SaaS** Software as a Service

**UI** User Interface

**UX** User Experience

**WOOT** WithOut Operational Transformation

# Bibliography

- [1] E. Chang. *eHub Interviews Writely*. English. Oct. 2005. URL: <https://web.archive.org/web/20110722190058/http://emilychang.com/ehub/app/ehub-interviews-writely/> (visited on 06/04/2021).
- [2] *Document collaboration and co-authoring*. en-US. URL: <https://support.microsoft.com/en-us/office/document-collaboration-and-co-authoring-ee1509b4-1f6e-401e-b04a-782d26f564a4> (visited on 06/04/2021).
- [3] *Firsts: The Demo - Doug Engelbart Institute*. URL: <https://www.dougenelbart.org/content/view/209/448/> (visited on 06/04/2021).
- [4] I. Fried. *Google pulls plug on Google Wave*. en. URL: <https://www.cnet.com/news/google-pulls-plug-on-google-wave/> (visited on 06/04/2021).
- [5] *Google acquires online word processor, Writely*. en-US. Mar. 2006. URL: <https://venturebeat.com/2006/03/09/google-acquires-online-word-processor-writely/> (visited on 06/04/2021).
- [6] M. Shapiro, N. Preguiça, C. Baquero, and M. Zawirski. “Conflict-Free Replicated Data Types.” en. In: *Stabilization, Safety, and Security of Distributed Systems*. Ed. by X. Défago, F. Petit, and V. Villain. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2011, pp. 386–400. ISBN: 9783642245503. DOI: 10.1007/978-3-642-24550-3\_29.

## *Bibliography*

- [7] C. Sun, D. Sun, A. Ng, W. Cai, and B. Cho. “Real Differences between OT and CRDT under a General Transformation Framework for Consistency Maintenance in Co-Editors.” en. In: *Proceedings of the ACM on Human-Computer Interaction* 4.GROUP (Jan. 2020), pp. 1–26. ISSN: 2573-0142. DOI: 10.1145/3375186. URL: <https://dl.acm.org/doi/10.1145/3375186> (visited on 06/04/2021).
- [8] *What’s different about the new Google Docs: Conflict resolution.* en. URL: [https://drive.googleblog.com/2010/09/whats-different-about-new-google-docs\\_22.html](https://drive.googleblog.com/2010/09/whats-different-about-new-google-docs_22.html) (visited on 06/04/2021).