dortmund university

Work in Parallel, Without Fear

Conflict Tolerance for Local-First Collaboration

Jonas Schürmann

Local-First Conf 2026, Lab Day, Berlin, 2026-07-14

Chair for Programming Systems, TU Dortmund University

Motivation

- Live collaboration is awesome, but sometimes you want to work alone to...
  - ...avoid interruptions
  - ...work offline
  - ...keep ideas to myself

Motivation

- Live collaboration is awesome, but sometimes you want to work alone to...
  - ...avoid interruptions
  - ...work offline
  - ...keep ideas to myself
- Working alone means concurrent edits pile up

Motivation

- Live collaboration is awesome, but sometimes you want to work alone to...
  - ...avoid interruptions
  - ...work offline
  - ...keep ideas to myself
- Working alone means concurrent edits pile up
- When CRDTs eventually merge them, automatic conflict resolution can silently discard my edits

- Branches provide isolated workspaces

- Branches provide isolated workspaces
- But: Upon merging, conflict resolution is a huge pain

- Branches provide isolated workspaces
- But: Upon merging, conflict resolution is a huge pain
- If you are the unlucky one to hit conflicts you have to resolve them...
  - ...immediately
  - ...alone
  - ...without context

- Branches provide isolated workspaces
- But: Upon merging, conflict resolution is a huge pain
- If you are the unlucky one to hit conflicts you have to resolve them...
  - ...immediately
  - ...alone
  - ...without context
- The result:
  - Interruptions
  - Premature resolution decisions
  - Dismissal of others' work
  - Introduction of errors

- Branches provide isolated workspaces
- But: Upon merging, conflict resolution is a huge pain
- If you are the unlucky one to hit conflicts you have to resolve them...
  - ...immediately
  - ...alone
  - ...without context
- The result:
  - Interruptions
  - Premature resolution decisions
  - Dismissal of others' work
  - Introduction of errors

$\Rightarrow$ Despite its usefulness, branching & merging is rare in local-first apps

Conflict Tolerance

- Conflicts are first-class objects in the document, so you can
  - Defer the resolution decision
  - Share conflicts with others
  - Resolve conflicts collaboratively

[1] https://www.inkandswitch.com/universal-version-control/

- Conflicts are first-class objects in the document, so you can
  - Defer the resolution decision
  - Share conflicts with others
  - Resolve conflicts collaboratively
- Benefits [3]:
  - Preserve all contributions
  - Uncover disagreements
  - Make deliberate and well-informed resolution decisions

[1] https://www.inkandswitch.com/universal-version-control/

Conflict Tolerance

- Conflicts are first-class objects in the document, so you can
  - Defer the resolution decision
  - Share conflicts with others
  - Resolve conflicts collaboratively
- Benefits [3]:
  - Preserve all contributions
  - Uncover disagreements
  - Make deliberate and well-informed resolution decisions
- Lazy merging: Integrate syntax, history, and conflicts in one CRDT [1]

[1] https://www.inkandswitch.com/universal-version-control/

- Conflicts are first-class objects in the document, so you can
  - Defer the resolution decision
  - Share conflicts with others
  - Resolve conflicts collaboratively
- Benefits [3]:
  - Preserve all contributions
  - Uncover disagreements
  - Make deliberate and well-informed resolution decisions
- Lazy merging: Integrate syntax, history, and conflicts in one CRDT [1]

$\Rightarrow$ Is conflict tolerance one of the missing pieces for universal version control[1]?

[1] https://www.inkandswitch.com/universal-version-control/

Version Control

Lazy Merging in Cinco Cloud [2]

Version Control

Conflict Representation

```
...
targetID: {
  "tag": "choice",
  "options": ["fed6", "2d4f"]
}
...
```

Cinco lazy merge driver

Lazy Merging in Cinco Cloud [2]

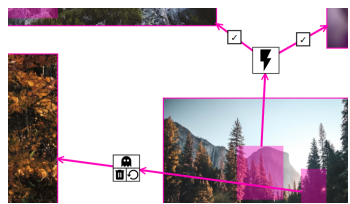
Version Control

Conflict Representation

```
...
targetID: {
  "tag": "choice",
  "options": ["fed6", "2d4f"]
}
...
```

Cinco lazy merge driver

Conflict Visualization



Conflict projection in the Cinco model editor

DEMO

Version Control

Automerge

Version Control

Automerge

Conflict Representation

```
Automerge.getConflicts(
    node.inputs,
    inputName
)
              ||
{
    "1@a1b2": "silhouette",
    "2@c3d4": "checkerboard"
}
```

Conflicts Object

Version Control

Automerge

Conflict Representation

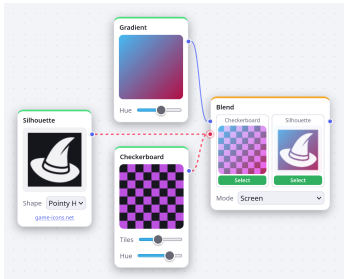
```
Automerge.getConflicts(
    node.inputs,
    inputName
)
```

‖

```
{
    "1@a1b2": "silhouette",
    "2@c3d4": "checkerboard"
}
```

Conflicts Object

Conflict Visualization



Conflict projection in React Flow

How to develop a conflict-tolerant local-first app

1. Implement a conflict-tolerant CRDT

2. Make the rest of the app conflict-tolerant

Challenges

- Conflict tolerance is a pervasive cross-cutting concern
- Similar to error handling

[2] https://www.inkandswitch.com/project/ambsheets/live25/

Challenges

- Conflict tolerance is a pervasive cross-cutting concern
- Similar to error handling
- How can we make it easy?
  - Conflict-tolerant UI frameworks
  - Conflict-tolerant databases/query languages
  - Conflict-tolerant APIs/programming languages (see Ambsheets[2])

[2] https://www.inkandswitch.com/project/ambsheets/live25/

Thank You!

dortmund university

Slides & Links

https://jonas-schuermann.name/talks/2026-local-first-conf/

References

[1] Jonas Schürmann and Bernhard Steffen. **"Lazy Merging: From a Potential of Universes to a Universe of Potentials".** en. In: *Electronic Communications of the EASST* Volume 82: 11th International Symposium on Leveraging Applications of Formal Methods (2023). DOI: 10.14279/tuj.eceasst.82.1226.

[2] Jonas Schürmann and Bernhard Steffen. **"Lazy Three-Way Model Merging in Cinco Cloud: A Lattice-Theoretic Approach".** In: *Formal Methods for Industrial Critical Systems: 30th International Conference, FMICS 2026, Liverpool, United Kingdom, September 03-04, 2026, Proceedings*. Ed. by Anne Remke and Bernhard Steffen. [to appear]. Springer Nature Switzerland, 2026.

[3] Konrad Wieland et al. **"Turning Conflicts into Collaboration".** In: *Computer Supported Cooperative Work (CSCW)* 22.2-3 (Sept. 2012), pp. 181–240. ISSN: 1573-7551. DOI: 10.1007/s10606-012-9172-4.