**CRDT**

|  |  |  |
| --- | --- | --- |
|  | yjs | automerge |
| Licenses | MIT | MIT |
| Created | 7 years | 5 Years |
| Modified | 3 days ago | 15 days ago |
| Maintainers | 1 | 2 |
| Dependencies | 1 | 3 |
| Open Issues | 40 | 71 |
| Daily Downloads | 16460 | 825 |
| Weekly Downloads | 85961 | 4463 |
| Monthly Downloads | 357920 | 18048 |
|  | WINNER | LOSER |

Aboveinformation gather based on below reputed website links

[**https://www.libhunt.com/compare-automerge-vs-yjs**](https://www.libhunt.com/compare-automerge-vs-yjs)

**https://npmcompare.com/compare/automerge,gulf,ot,scuttlebutt,yjs**

**YJS**

**Pros**

* Own web socket and wrtc library – y-websocket, etc.,
* Own local storage / db – y-indexedDB, etc
* Faster result set compare to other CRDT frameworks
* Avg. latency 0.015ms - 0.021ms
* Yjs provides predefined binding editors like quill and many editors
* Code Maintained frequently by the community

**Cons**

* No much developer communities but also far better than other CRDT frameworks

**Auto merge**

**Pros**

Very simple framework compare to other CRDT frameworks example: uses only change, merge, save & load methods.

**Cons**

* Had to depend on many 3rd party libraries like websocket, db
* Shared editing is more complex implementation
* Slower result set on share programming becoz of 3rd party websocket
* Avg. latency 0.20s to 0.30s
* No Editor components like yjs
* No much developer communities
* Code maintaining is not that frequent

**Correct Scenario**

One node - > one editor -> N no of users

**Example**: Git, Svn, Online peer to peer programming editors etc., May be using CRDT

**Wrong Scenario**

Two Nodes -> two editor -> N no of users

**Real-Time Example**: Messenger, Whatsapp etc.

**Why?**

1. Core Concept of CRDT -> (Conflict Free Replicated Data Type ) – To merge the data in same node or instance
2. Users from 2 node’s and 2 different editor’s then needed common place to hold the data or Update Changes like Database, Web socket