

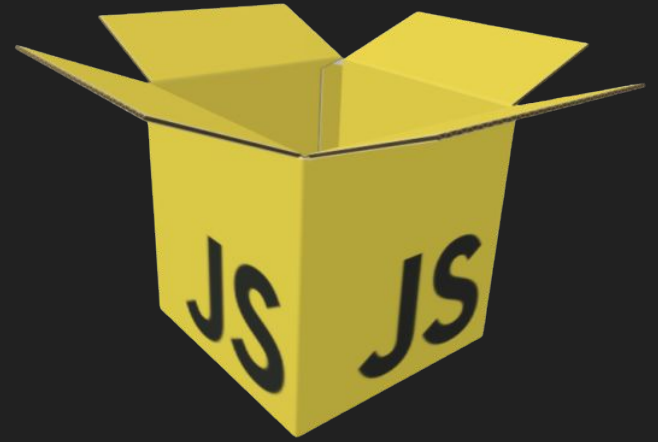
Record & Tuple

Stage 2 Update @ TC39 Tokyo 🤔

Robin Ricard & Rick Button
Bloomberg

*Advisor: Daniel Ehrenberg
Igalia*

Box



Boxmaker, Boxmaker....

Boxes - a refresher

```
const object = {  
  bar: () => "hello world",  
};
```

```
const record = #{  
  foo: Box(object),  
};
```

```
assert("hello world" === record.foo.unbox().bar());
```

Boxes - viable!

Initially concerned that Box violates membrane requirements.

Champion group worked with SES to determine how Record/Tuple work with membranes.

Conclusion: not a security issue.

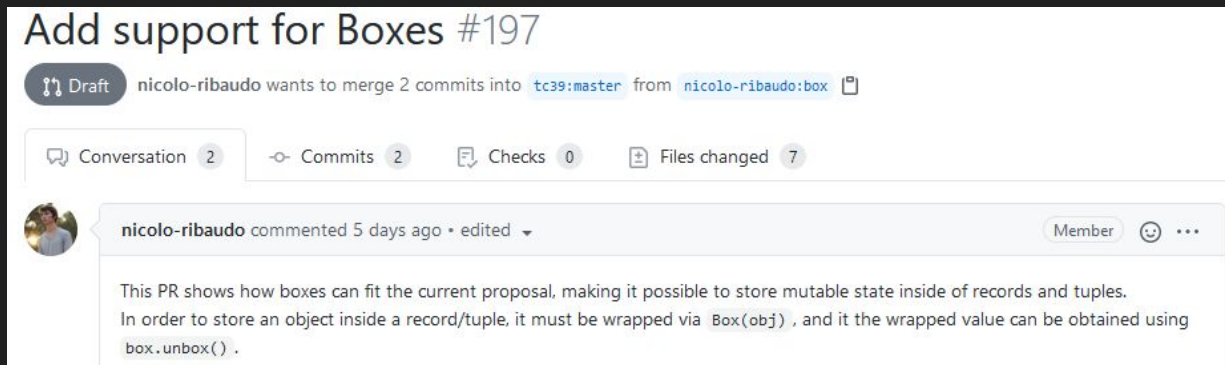
Discussion: [Issue #200](#)

Boxes - viable!

Initially concerned that Box violates membrane requirements.

Champion group worked with SES to determine how Record/Tuple work with membranes.

Conclusion:



The screenshot shows a GitHub pull request interface. At the top, the title is "Add support for Boxes #197". Below the title, it says "Draft" and "nicolo-ribaudo wants to merge 2 commits into tc39:master from nicolo-ribaudo:box". There are statistics for the pull request: "Conversation 2", "Commits 2", "Checks 0", and "Files changed 7". A comment from "nicolo-ribaudo" is visible, dated "5 days ago". The comment text reads: "This PR shows how boxes can fit the current proposal, making it possible to store mutable state inside of records and tuples. In order to store an object inside a record/tuple, it must be wrapped via `Box(obj)`, and it the wrapped value can be obtained using `box.unbox()`."

[Issue #200](#)

Primitives vs Objects

Primitives vs Objects

Currently, Records and Tuples are primitives, i.e. “not objects”.

Primitives don't have identity, are observably equivalent to other values with the same type and “contents”.

Objects have identity; you can observe this via `==/===/Object.is`.

Identity-less objects?

- What if Records and Tuples were “identity-less” objects:
 - Compared by contents
 - No observable identity
 - Always frozen
 - Can't be in Weak{Map/Set/Ref}
 - Can't contain private fields.
- This means:
 - Surface API stays the same
 - No wrapper objects.
 - `[[Prototype]] == {Record/Tuple}.prototype`

Discussion: [Issue #201](#)

Record & Tuple in the TC39 Research Call

Interview scenario under review in the TC39 Research call

Broad topics for research:

- Syntax
- Boxing vs Auto-Boxing vs No-Boxing
- Ergonomics of updating Records & Tuples

Interviews will shape future surveys

Call for Participation: Monthly Record & Tuple Call

Discussion of Record and Tuple proposal and related work. We are interested in feedback from potential users, library authors, and implementers. Please attend if interested!

Sign up via Doodle, link will be posted in Reflector.

Started SpiderMonkey implementation work

Very early, much work to be done.

S/O to **Nicolò Ribaud** for the awesome work! 🚀🚀🚀

<https://phabricator.services.mozilla.com/D87232>

<https://phabricator.services.mozilla.com/D87272>

<https://phabricator.services.mozilla.com/D87586>

<https://phabricator.services.mozilla.com/D87737>

<https://phabricator.services.mozilla.com/D87762>

Questions?