

[Skip to main content](#)

Stack Overflow

1. [About](#)
2. Products
3. [For Teams](#)
4. [Stack Internal Implement a knowledge platform layer to power your enterprise and AI tools.](#)
5. [Stack Data Licensing Get access to top-class technical expertise with trusted & attributed content.](#)
6. [Stack Ads Connect your brand to the world's most trusted technologist communities.](#)
7. [Releases Keep up-to-date on features we add to Stack Overflow and Stack Internal.](#)
8. [About the company Visit the blog](#)



Loading...

[current community](#)

- [Stack Overflow](#)
- [help chat](#)
- [Meta Stack Overflow](#)

your communities

[Sign up](#) or [log in](#) to customize your list.

[more stack exchange communities](#)

- [company blog](#)
 - 3. [Log in](#)
 - 4. [Sign up](#)
- [AI Assist is now on Stack Overflow.](#) Start a chat to get instant answers from across the network. Sign up to save and share your chats.
- [Home](#)
 - [Questions](#)
 - [AI Assist](#)
 - [Tags](#)
 - [Challenges](#)
 - 7. [Chat](#)
 - [Articles](#)
 - [Users](#)
 - [Companies](#)
 - [Collectives](#)

Communities for your favorite technologies. [Explore all Collectives](#)

Stack Internal

Stack Overflow for Teams is now called **Stack Internal**. Bring the best of human thought and AI automation together at your work.

[Try for free](#) [Learn more](#)

[Stack Internal](#)

Bring the best of human thought and AI automation together at your work. [Learn more](#)

Collectives™ on Stack Overflow

Find centralized, trusted content and collaborate around the technologies you use most.

[Learn more about Collectives](#)

[Stack Internal](#)

Knowledge at work

Bring the best of human thought and AI automation together at your work.

[Explore Stack Internal](#)

[Create a Set of Sets](#)

[Ask Question](#)

Asked 3 years, 8 months ago

Modified [3 years, 2 months ago](#)

Viewed 1k times

4

How does one create a set of sets in Rust? Is it necessary to write an `impl` block for every concrete type satisfying `HashSet<HashSet<_>>`?

Minimal failing example:

```
fn main () {
    let a: HashSet<u32> = HashSet::new();
    let c: HashSet<HashSet<u32>> = HashSet::new();
    c.insert(a);
}
```

Error:

```
"insert" method cannot be called on `std::collections::HashSet<std::collections::HashSet<u32>>` due to unsatisfied trait bound
HashSet doesn't satisfy `std::collections::HashSet<u32>: Hash
```

Is it possible to override the fact that `HashSet` is unhashable? I'd like to use a `HashSet` and need my contents to be unique by actual (memory) equality; I don't need to be unique by contents.

- [rust](#)

[Share](#)

[Improve this question](#)

Follow

[edited Sep 26, 2022 at 4:34](#)

[Horohtar](#)

5,66444 gold badges3434 silver badges4444 bronze badges

asked Mar 21, 2022 at 18:01

[Test](#)

1,79833 gold badges2222 silver badges3939 bronze badges

10

4

`HashSet` is not hashable since its iteration order is unpredictable. You can use [BTreeSet](#) instead.

apilat – [apilat](#)

2022-03-21 18:14:54 +00:00

Commented Mar 21, 2022 at 18:14

Does this answer your question? [Why is BTreeMap hashable, and not HashMap?](#)

Aiden4 – [Aiden4](#)

2022-03-21 18:26:34 +00:00

Commented Mar 21, 2022 at 18:26

Yes. Makes perfect sense. It is possible to "override" the behavior: make two different sets unequal even when they have the same contents?

Test – [Test](#)

2022-03-21 18:35:19 +00:00

Commented Mar 21, 2022 at 18:35

I'd like to have a set of sets and want them to be unique by "actual" (memory) equality, not by contents.

Test – [Test](#)

2022-03-21 18:41:27 +00:00

Commented Mar 21, 2022 at 18:41

2

@Test Rust values are not automatically heap-allocated, so they don't have a stable address you could use as "identity", as is the case in, say, Python.

user4815162342 – [user4815162342](#)

2022-03-21 18:42:11 +00:00

Commented Mar 21, 2022 at 18:42

| [Show 5 more comments](#)

2 Answers

Sorted by: [Reset to default](#)

Highest score (default) Trending (recent votes count more) Date modified (newest first) Date created (oldest first)

I'd like to have a set of sets and want them to be unique by "actual" (memory) equality, not by contents.

To do so you first need to box the hashset so that it has a stable memory address. For example:

```
struct Set<T>(Box<HashSet<T>>);
```

To make your `Set` hashable, you'll need to implement `Hash` and `Eq`:

```
impl<T> Set<T> {
    fn as_addr(&self) -> usize {
        // as_ref() gives the reference to the heap-allocated contents
        // inside the Box, which is stable; convert that reference to a
        // pointer and then to usize, and use it for hashing and equality.
        self.0.as_ref() as *const _ as usize
    }
}

impl<T> Hash for Set<T> {
    fn hash<H: Hasher>(&self, state: &mut H) {
        self.as_addr().hash(state);
    }
}

impl<T> Eq for Set<T> {}

impl<T> PartialEq for Set<T> {
    fn eq(&self, other: &Self) -> bool {
        self.as_addr() == other.as_addr()
    }
}
```

Finally, you'll need to add some set-like methods and a constructor to make it usable:

```
impl<T: Hash + Eq> Set<T> {
    pub fn new() -> Self {
        Set(Box::new(HashSet::new()))
    }

    pub fn insert(&mut self, value: T) {
        self.0.insert(value);
    }

    pub fn contains(&mut self, value: &T) -> bool {
        self.0.contains(value)
    }
}
```

Now your code will work, with the additional use of `Rc` so that you have the original `Set` available for lookup after you insert it:

```
fn main() {
    let mut a: Set<u32> = Set::new();
    a.insert(1);
    let a = Rc::new(a);
    let mut c: HashSet<_> = HashSet::new();
    c.insert(Rc::clone(&a));
    assert!(c.contains(&a));
}
```

[Playground](#)

[Share](#)

[Improve this answer](#)

Follow

[edited Mar 21, 2022 at 19:11](#)

answered Mar 21, 2022 at 18:57

[user4815162342](#)

159k2222 gold badges350350 silver badges418418 bronze badges

Sign up to request clarification or add additional context in comments.

4 Comments

Add a comment

Test

[Test Over a year ago](#)

This is very nice, thank you. Does it not require a pin? I broke out the specifically question about Hash/Eq, but now I'm a bit confused.

2022-03-21T23:31:10.517Z+00:00

0

Reply

- Copy link

Test

[Test Over a year ago](#)

Could one create a struct with the hashset as one property and the Box as a separate property? The hashset could be used without a lookup, and the box serve only for equality.

2022-03-21T23:49:23.983Z+00:00

0

Reply

- Copy link

user4815162342

[user4815162342 Over a year ago](#)

@Test Re 1st comment: It doesn't require a pin, it's actually the box that makes an object stable in memory. A pin is only needed when you must expose a mutable reference to the box and need to prevent someone from replacing it with another box using `std::mem::replace()` or equivalent. That cannot happen here because the box is private to the object, and it's ok if replacing the whole object changes its identity.

2022-03-22T07:22:39.64Z+00:00

0

Reply

- Copy link

user4815162342

[user4815162342 Over a year ago](#)

@Test Re 2nd comment: yes, but in that case you don't need to box, an ordinary counter will suffice. That approach is covered in [my answer to your other question](#).

2022-03-22T07:23:48.017Z+00:00

0

Reply

- Copy link

Add a comment

-1

As pointed out helpfully in the comments, it's not possible to hash sets because they have no fixed address. An effective, if inelegant, solution, is to wrap them in a specialized struct:

```
struct HashableHashSet<T> {
    hash: ...
    hashset: HashSet<T>
}
```

And then hash the struct by memory equality.

[Share](#)

[Improve this answer](#)

Follow

[edited Mar 21, 2022 at 18:50](#)

answered Mar 21, 2022 at 18:45

[Test](#)

1,79833 gold badges2222 silver badges3939 bronze badges

1 Comment

Add a comment

user4815162342

[user4815162342 Over a year ago](#)

As also pointed out in the comments (and in the other answer), the `hashset` field needs to be something like `Box<HashSet<T>>` to make its address stable as `HashableHashSet` gets moved.

2022-03-21T19:05:22.493Z+00:00

1

Reply

- Copy link

Your Answer

Thanks for contributing an answer to Stack Overflow!

- Please be sure to *answer the question*. Provide details and share your research!

But avoid ...

- Asking for help, clarification, or responding to other answers.
- Making statements based on opinion; back them up with references or personal experience.

To learn more, see our [tips on writing great answers](#).

[Sign up](#) or [log in](#)

Sign up using Google

Sign up using Email and Password

Submit

Post as a guest

Name

Email

Required, but never shown

Post as a guest

Name

Email

Required, but never shown

Post Your Answer Discard

By clicking "Post Your Answer", you agree to our [terms of service](#) and acknowledge you have read our [privacy policy](#).

Start asking to get answers

Find the answer to your question by asking.

[Ask question](#)

Explore related questions

- [rust](#)

See similar questions with these tags.

- The Overflow Blog
 - [Introducing Stack Overflow AI Assist—a tool for the modern developer](#)
 - [Treating your agents like microservices](#)
- Featured on Meta
 - [Chat room owners can now establish room guidelines](#)
 - [AI Assist is now available on Stack Overflow](#)
 - [Policy: Generative AI \(e.g., ChatGPT\) is banned](#)

Linked

1

[Rust: Implement Hash/Eq](#)

2

[Why is BTreemap hashable, and not HashMap?](#)

Related

16

[Is there a quick way to create a set?](#)

34

[How do I create a heterogeneous collection of objects?](#)

11

[How to get every subset of a vector in Rust?](#)

1

[How to repeatedly loop over subsets of a collection determined by an attribute of the items?](#)

5

[How do I intersect two HashSet while moving values in common into a new set?](#)

33

[How can I insert all values of one HashSet into another HashSet?](#)

2

[How to create instances from Array values](#)

1

[How can I create a HashSet from a borrowed array of a generic type in Rust?](#)

1

[How to implement a HashSet of pairs that prevents inserting overlapping values?](#)

5

[Iterating over all possible partitions of a given set](#)

Hot Network Questions

- [When did "doubt" meaning question or point of uncertainty fall out of use outside of Indian English?](#)
- [Best Welding Method to Fix a Damaged Steel Surly Frame](#)
- [How do we define time-ordering operations in QFT?](#)
- [How to check if two tables are identical?](#)
- [Displaying features in a layer of geometrytype = GeometryCollection25D](#)
- [I can't add a loop cut on one face](#)
- ["Save as webpage, HTML Only" Fail while offline, Firefox](#)
- [is it possible to keep the space when using url in latex](#)
- [Finding real variable where a 2x2 complex matrix is singular in Mathematica](#)
- [Looking for a proper formal substitute for "quick fix" for formal letter](#)
- [Do something and "win"](#)
- [Polygons \(and some stars\)](#)
- [Why doesn't sleep mode \(S3\) turn on in the BIOS on my Lecoo Pro 14 laptop \(N155A, Lenovo sub-brand\) with Windows 11?](#)
- [King defense setups](#)
- [Installing root certificates by government](#)
- [I2C bus problems when using two distance sensors in Arduino](#)
- [Explain this seeming contradiction in Euclid Book 1 Proposition 16](#)
- [How to protect 100-year-old wooden floors for the next 100 years?](#)

- [Is "We will review application starting from..." a deadline?](#)
- [2 ways of resolution of vectors](#)
- [Critiques of the real vs. nominal definition distinction](#)
- [When should I use tied notes vs. the whole note duration?](#)
- [What's the point of swapping polarity between mod slots?](#)
- [Is the phrase "murder will out" from Chaucer?](#)

[more hot questions](#)

[Question feed](#)

Subscribe to RSS

[Question feed](#)

To subscribe to this RSS feed, copy and paste this URL into your RSS reader.

[Stack Overflow](#)

- [Questions](#)
- [Help](#)
- [Chat](#)

[Business](#)

- [Stack Internal](#)
- [Stack Data Licensing](#)
- [Stack Ads](#)

[Company](#)

- [About](#)
- [Press](#)
- [Work Here](#)
- [Legal](#)
- [Privacy Policy](#)
- [Terms of Service](#)
- [Contact Us](#)
- [Cookie Settings](#)
- [Cookie Policy](#)

[Stack Exchange Network](#)

- [Technology](#)
- [Culture & recreation](#)
- [Life & arts](#)
- [Science](#)
- [Professional](#)
- [Business](#)
- [API](#)
- [Data](#)
- [Blog](#)
- [Facebook](#)
- [Twitter](#)
- [LinkedIn](#)
- [Instagram](#)