Module sui::versioned

A wrapper type that supports versioning of the inner type. The inner type is a dynamic field of the Versioned object, and is keyed using version. User of this type could load the inner object using corresponding type based on the version. You can also upgrade the inner object to a new type version. If you want to support lazy upgrade of the inner type, one caveat is that all APIs would have to use mutable reference even if it's a read-only API.

Represents a hot potato object generated when we take out the dynamic field. This is to make sure that we always put a new value back.

Failed to upgrade the inner object due to invalid capability or new version.

Create a new Versioned object that contains a initial value of type T with an initial version.

Get the current version of the inner type.

Load the inner value based on the current version. Caller specifies an expected type T. If the type mismatch, the load will fail.

Similar to load_value, but return a mutable reference.

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

Destroy this Versioned container, and return the inner object.

Struct

A wrapper type that supports versioning of the inner type. The inner type is a dynamic field of the Versioned object, and is keyed using version. User of this type could load the inner object using corresponding type based on the version. You can also upgrade the inner object to a new type version. If you want to support lazy upgrade of the inner type, one caveat is that all APIs would have to use mutable reference even if it's a read-only API.

```bash

```

Represents a hot potato object generated when we take out the dynamic field. This is to make sure that we always put a new value back.

```bash

```

Failed to upgrade the inner object due to invalid capability or new version.

```bash

```

Create a new Versioned object that contains a initial value of type T with an initial version.

```bash

```

```bash

```

Get the current version of the inner type.

```bash

```
```

```bash
```

Load the inner value based on the current version. Caller specifies an expected type T. If the type mismatch, the load will fail.

```bash
```

```bash
```

Similar to load_value, but return a mutable reference.

```bash
```

```bash
```

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Struct

Represents a hot potato object generated when we take out the dynamic field. This is to make sure that we always put a new value back.

```bash
```

Failed to upgrade the inner object due to invalid capability or new version.

```bash
```

Create a new Versioned object that contains a initial value of type T with an initial version.

```bash
```

```bash
```

Get the current version of the inner type.

```bash
```

```bash
```

Load the inner value based on the current version. Caller specifies an expected type T. If the type mismatch, the load will fail.

```bash
```

```bash
```

Similar to load_value, but return a mutable reference.

```bash
```

```bash
```

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Constants

Failed to upgrade the inner object due to invalid capability or new version.

```bash
```

Create a new Versioned object that contains a initial value of type T with an initial version.

```bash
```

```bash
```

Get the current version of the inner type.

```bash
```

```bash
```

Load the inner value based on the current version. Caller specifies an expected type T. If the type mismatch, the load will fail.

```bash
```

```bash
```

Similar to load_value, but return a mutable reference.

```bash
```

```bash
```

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

```
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Function

Create a new Versioned object that contains a initial value of type T with an initial version.

```bash
```

```bash
```

Get the current version of the inner type.

```bash
```

```bash
```

Load the inner value based on the current version. Caller specifies an expected type T. If the type mismatch, the load will fail.

```bash
```

```bash
```

Similar to load_value, but return a mutable reference.

```bash
```

```bash
```

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Function

Get the current version of the inner type.

```bash
```

```bash
```

Load the inner value based on the current version. Caller specifies an expected type T. If the type mismatch, the load will fail.

```bash
```

```bash
```

Similar to load_value, but return a mutable reference.

```bash
```

```bash
```

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Function

Load the inner value based on the current version. Caller specifies an expected type T. If the type mismatch, the load will fail.

```bash
```

```bash
```

Similar to load_value, but return a mutable reference.

```bash
```

```bash
```

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Function

Similar to load_value, but return a mutable reference.

```bash
```

```bash
```

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Function

Take the inner object out for upgrade. To ensure we always upgrade properly, a capability object is returned and must be used when we upgrade.

```bash
```

```bash
```

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Function

Upgrade the inner object with a new version and new value. Must use the capability returned by calling remove_value_for_upgrade.

```bash
```

```bash
```

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```

Function

Destroy this Versioned container, and return the inner object.

```bash
```

```bash
```