

Meteor runs most app code within Fibers, which allows keeping track of the context a function is running in.

`Meteor.EnvironmentVariable` works with `Meteor.bindEnvironment`, promises, and many other Meteor API's to preserve the context in async code. Some examples of how it is used in Meteor are to store the current user in methods, and record which arguments have been checked when using `audit-argument-checks`.

```
const currentRequest = new Meteor.EnvironmentVariable();

function log(message) {
  const requestId = currentRequest.get() || 'None';
  console.log(`[${requestId}]`, message);
}

currentRequest.withValue('12345', () => {
  log('Handling request'); // Logs: [12345] Handling request
});
```

{% apibox "Meteor.EnvironmentVariable" %}

{% apibox "Meteor.EnvironmentVariable.get" %}

{% apibox "Meteor.EnvironmentVariable.withValue" %}

{% apibox "Meteor.bindEnvironment" %}