

Catching Errors

Catching Errors in Hardhat

There are some scenarios where handling errors in hardhat is not as straightforward as it seems.

Generally this simple ethers client would suffice to catch errors inside a try block:

```
try
```

```
{ await contract . method ( params ) ; }
```

```
catch
```

```
( error )
```

```
{ console . log (error! ) ; }
```

 However, if a contract calls a fails only on the commit of a transaction and not in the preceding gas estimation, then this will not raise an error. This is because the transaction will be successfully added on-chain, but the result will be a failure.

The reason this happens is that during gas estimation the FHE operations are not actually performed, but rather the gas is estimated based on the size of the encrypted data.

Instead, when calling contracts that perform FHE operations, we recommend checking for the status of the transaction:

```
try
```

```
{ let tx =
```

```
await contract . method ( params ) ; let receipt =
```

```
await tx . wait ( ) ; if
```

```
( receipt ?. status ===
```

```
0 )
```

```
{ throw
```

```
Error ( Transaction failed! ) } }
```

```
catch
```

```
( error )
```

```
{ console . log (error! ) ; }
```

 note This type of behaviour might be client and framework specific, and might change in the future - we're putting it here for now because we've seen this behaviour in hardhat. We'll update in the future if this is only hardhat specific, ethers specific, or if it's a general behaviour. [Edit this page](#)

[Previous](#) [Network Keys](#) [Next](#) [Templates](#)