

```
D4D4D4;--ch-t-background: #1E1E1E;--ch-t-lighter-
inlineBackground: #1e1e1ee6;--ch-t-editor-background:
#1E1E1E;--ch-t-editor-foreground: #D4D4D4;--ch-t-editor-
rangeHighlightBackground: #ffffff0b;--ch-t-editor-
infoForeground: #3794FF;--ch-t-editor-
selectionBackground: #264F78;--ch-t-focusBorder:
#007FD4;--ch-t-tab-activeBackground: #1E1E1E;--ch-t-
tab-activeForeground: #ffffff;--ch-t-tab-
inactiveBackground: #2D2D2D;--ch-t-tab-
inactiveForeground: #ffffff80;--ch-t-tab-border: #252526;--
ch-t-tab-activeBorder: #1E1E1E;--ch-t-editorGroup-
border: #444444;--ch-t-editorGroupHeader-
tabsBackground: #252526;--ch-t-editorLineNumber-
foreground: #858585;--ch-t-input-background: #3C3C3C;-
-ch-t-input-foreground: #D4D4D4;--ch-t-icon-foreground:
#C5C5C5;--ch-t-sideBar-background: #252526;--ch-t-
sideBar-foreground: #D4D4D4;--ch-t-sideBar-border:
#252526;--ch-t-list-activeSelectionBackground: #094771;--
ch-t-list-activeSelectionForeground: #ffffffe;--ch-t-list-
hoverBackground: #2A2D2E; }
```

## Migrate to v1

This guide references the major changes between `safe-core-sdk` and `protocol-kit v1` to help those migrating an existing application.

Note: Follow this guide before migrating to `protocol-kit v2`.

You can remove `@safe-global/safe-core-sdk` from your `package.json` after completing this guide.

### Adding the new dependency

To add the Protocol Kit to your project, run the following:

```
_10 yarn add @safe-global/protocol-kit@1.3.0 If you use the types library, you will need to update to v2.3.0:
```

```
_10 yarn add @safe-global/safe-core-sdk-types@2.3.0
```

### EthAdapter

#### EthersAdapter

(`safe-ethers-lib`)

`EthersAdapter` isn't in a separate package anymore. Now, it's provided inside the `protocol-kit` package.

`protocol-kit v1` only supports `ethers v5`

`_10 // old _10 import EthersAdapter from '@safe-global/safe-ethers-lib' _10 _10 // new _10 import { EthersAdapter } from '@safe-global/protocol-kit' After this change, you can remove @safe-global/safe-ethers-lib from your package.json .`

## Web3Adapter

(safe-web3-lib)

Web3Adapter isn't in a separate package anymore. Now, it's part of the protocol-kit package.

Note: protocol-kit v1 only supports Web3.js v1.

`_10 // old _10 import Web3Adapter from '@safe-global/safe-web3-lib' _10 _10 // new _10 import { Web3Adapter } from '@safe-global/protocol-kit' After this change, you can remove @safe-global/safe-web3-lib from your package.json .`

## Type changes

Type changes are affecting the web3 and ethers adapter libraries.

`getSafeContract`, `getMultisendContract`, `getMultisendCallOnlyContract`, `getCompatibilityFallbackHandlerContract`, `getSafeProxyFactoryContract`, `getSignMessageLibContract` and `getCreateCallContract` don't need the `chainId` parameter anymore, they will use the chain set on the provider. Also, they return a `Promise` now.

`estimateGas` now returns a string instead of a number .

## safeFactory.deploySafe()

`SafeDeploymentConfig` was simplified. If you were using `saltNonce` you should set it like this:

`_16 // old _16 const safeAccountConfig: SafeAccountConfig = { _16 ... _16 } _16 const safeDeploymentConfig: SafeDeploymentConfig = { saltNonce } _16 _16 const safeSdk = await safeFactory.deploySafe({ safeAccountConfig, safeDeploymentConfig }) _16 _16 // new _16 const safeAccountConfig: SafeAccountConfig = { _16 ... _16 } _16 _16 const saltNonce = " _16 _16 const protocolKit = await safeFactory.deploySafe({ safeAccountConfig, saltNonce })`

## getAddress()

The `getAddress()` method now returns a `Promise` .

`_10 // old _10 const safeAddress = safeSdk.getAddress() _10 _10 // new _10 const safeAddress = await protocolKit.getAddress()`

## General type changes

If you set `safeTxGas`, `baseGas`, `orgasPrice`, you must use the type `string` instead of `number` .

[Messages Migrate to v2](#) Was this page helpful?

[Report issue](#)