

[Translation](#) / [TranslationSession](#) / [TranslationSession.Configuration](#)

Structure

TranslationSession.Configuration

A type containing the information to use when performing a translation.

iOS 18.0+ | iPadOS 18.0+ | Mac Catalyst 26.0+ | macOS 15.0+

```
struct Configuration
```

Overview

Specify the source and target languages to use in a translation session with this object. Initialize an instance of this type using the [`init\(source:target:\)`](#) and passing in the source and target languages. When you pass this configuration into the [`translationTask\(_ :action:\)`](#) function, the framework uses the languages you specify for translation.

To re-run a translation, store the configuration object as state in your SwiftUI view by using the `State` property wrapper. Then change one of the configuration properties (such as the source or target language) to re-run the translation on a new pair of languages. You can also call [`invalidate\(\)`](#) on the configuration instance to re-run the translation using the same languages with new content to translate. When you do, the action closure of [`translationTask\(_ :action:\)`](#) triggers and the framework translates the text.

The following example demonstrates how to trigger a new translation from a button press:

```
struct TranslationExample: View {
    var sourceText: String
    var sourceLanguage: Locale.Language?
    var targetLanguage: Locale.Language?

    @State private var targetText: String?
    @State private var configuration: TranslationSession.Configuration?
```

```
var body: some View {
    VStack {
        Text(targetText ?? sourceText)
        Button("Translate") {
            guard configuration != nil else {
                configuration = TranslationSession.Configuration(
                    source: sourceLanguage,
                    target: targetLanguage)
            }
            self.configuration.invalidate()
        }
    }
    .translationTask(configuration) { session in
        do {
            let response = try await session.translate(sourceText)
            targetText = response.targetText
        } catch {
            // Handle error.
        }
    }
}
```

Topics

Initializers

`init(source: Locale.Language?, target: Locale.Language?)`

Creates a configuration from a source and target language.

Instance Properties

`var source: Locale.Language?`

The language to translate content from.

`var target: Locale.Language?`

The language to translate content into.

`var version: Int`

A value the equals function uses to represent change in the configuration instance.

Instance Methods

`func invalidate()`

Invalidate the current translation session and re-run it with new content.

Relationships

Conforms To

Equatable

See Also

Preparing for translation

`func prepareTranslation() async throws`

Asks for permission to download translation languages without doing any translations.