Documentation Language: Swift

SwiftUI / View / translationTask(_:action:)

Instance Method

translationTask(_:action:)

Adds a task to perform before this view appears or when the translation configuration changes.

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

```
nonisolated
func translationTask(
    _ configuration: TranslationSession.Configuration?,
    action: @escaping (TranslationSession) async -> Void
) -> some View
```

Parameters

`configuration`

A configuration for a `TranslationSession`. When this configuration is non-nil and changes, the `action` runs providing an instance of `TranslationSession` to perform translations.

`action`

This closure runs when the `configuration` is non-nil and changes. If the `configuration` is initially non-nil, it calls the action closure when the view appears and provides a `TranslationSession` instance to perform one or multiple translations.

Discussion

This task provides an instance of `TranslationSession` to use in translations. Whenever `TranslationSession.Configuration` changes and isn't `nil`, the closure `action` runs, providing a `TranslationSession` instance to perform one or more translations.

For example, you can create a `TranslationSession.Configuration` in response to a button press to trigger translation:

```swift
struct ContentView: View {
    @State private var sourceText = "Hallo, Welt!"
    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?.invalidate()
                    return
                }

                 configuration = TranslationSession.Configuration(
                    source: sourceLanguage,
                    target: targetLanguage)
            }
            .translationTask(configuration) { session in
                Task { @MainActor in
                    do {
                        let response = try await session.translate(sourceText)
                        targetText = response.targetText
                    } catch {
                        // code to handle error
                    }
                }
            }
        }
    }
}
```

The system throws a `fatalError` if you use a `TranslationSession` instance after the attached view disappears or if you use it after changing the `configuration`. This causes the `action` closure to provide a new `TranslationSession` instance.

See Also

Showing a translation

```
func translationPresentation(isPresented: Binding<Bool>, text: String, attachmentAnchor: PopoverAttachmentAnchor, arrowEdge: Edge, replacementAction: ((String) -> Void)?) -> some View
```

Presents a translation popover when a given condition is true.

```
func translationTask(source: Locale.Language?, target: Locale.Language?, action: (TranslationSession) async -> Void) -> some View
```

Adds a task to perform before this view appears or when the specified source or target languages change.