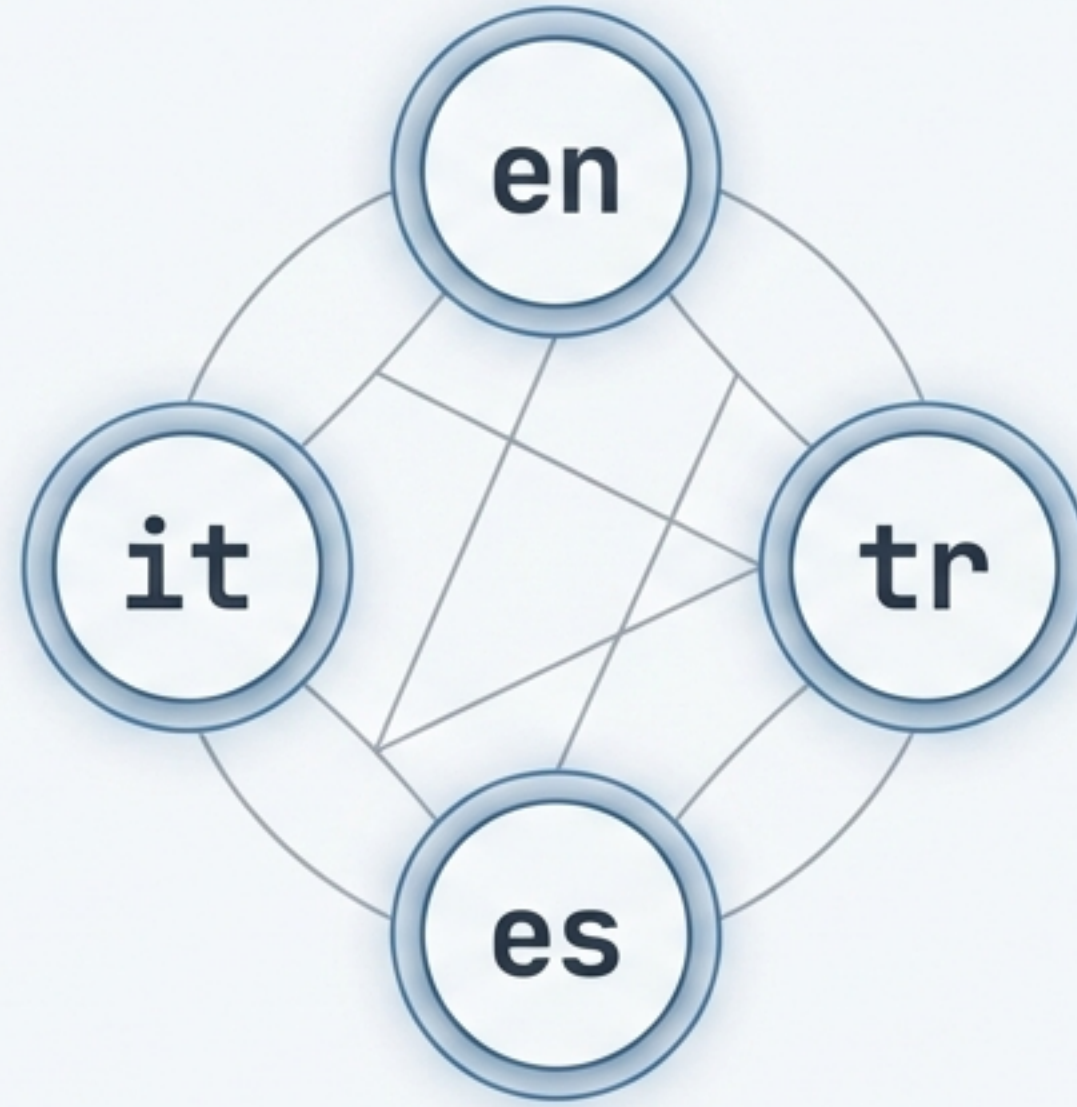


Flutter'da Manuel Uluslararasılařtırma

Harici araçlar olmadan çoklu dil desteęi oluřturma rehberi.

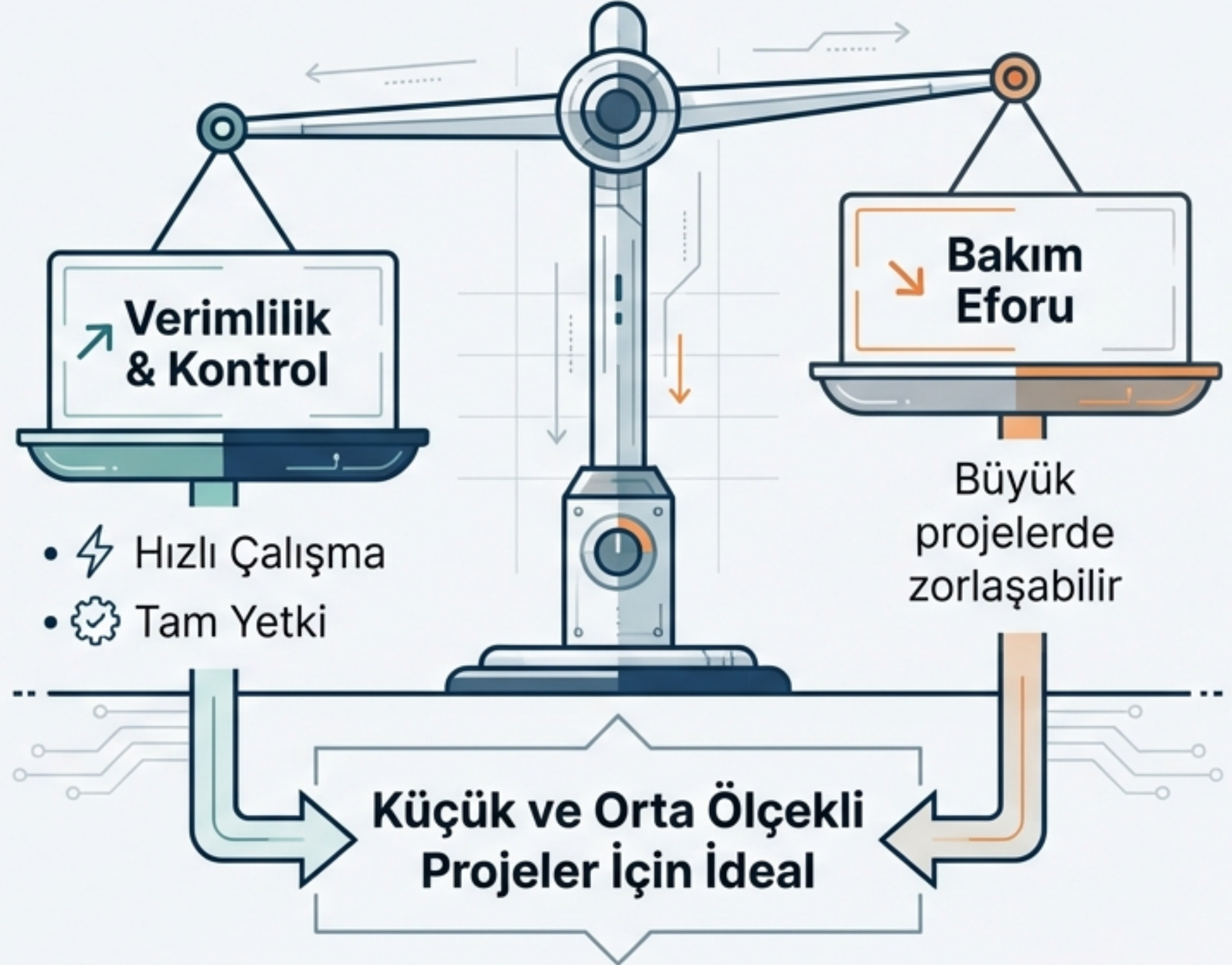


TEKNİK REHBER SERİSİ

Neden Manuel Yöntem?

Tanım

Otomatik araçlara veya harici dosyalara (JSON/ARB) güvenmeden, çevirilerin doğrudan Dart kodu içinde yönetilmesi.



Proje ve Klasör Yapısı

Düzenli bir mimari için önerilen dosya ağacı.

 lib/

 localization/

 app_localization.dart

 localization_delegate.dart

Çeviri veritabanı ve
Temsilci (Delegate)
sınıfları burada bulunur.

 routes/

 main.dart

Temel Sınıf: AppLocalization

```
class AppLocalization {  
    final Locale locale;  
    const AppLocalization(this.locale);
```

```
// Veritabanı (O(1) Erişim)
```

```
static Map<String, Map<String, String>> _db = {
```

```
    "en": {
```

```
        "hello": "Hello!",
```

```
        "hello_world": "Hello world",
```

```
    },
```

```
    "it": {
```

```
        "hello": "Ciao!",
```

```
        "hello_world": "Ciao mondo",
```

```
    },
```

```
    "es": {
```

Veri sadece bir
kez başlatılır.

Varsayılan dil.

İpucu: String Yerine Enum Kullanımı

Tip güvenliği (Type Safety) için anahtarları sabitleyin.

Riskli Yöntem

```
"en": {  
    "hello": "Hello!"  
}
```

String anahtarlar yazım hatalarına (typo) açıktır.

Güvenli Yöntem

```
enum _LockKeys { hello }  
// ...  
"en": {  
    _LockKeys.hello: "Hello!"  
}
```

Enum kullanımı derleme zamanı güvenliği sağlar.

» _LockKeys private yapılarak dış erişim kısıtlanmalıdır.

Modern Erişim: Extension Metodu

BuildContext üzerine yazılan bir eklenti ile verilere temiz erişim.

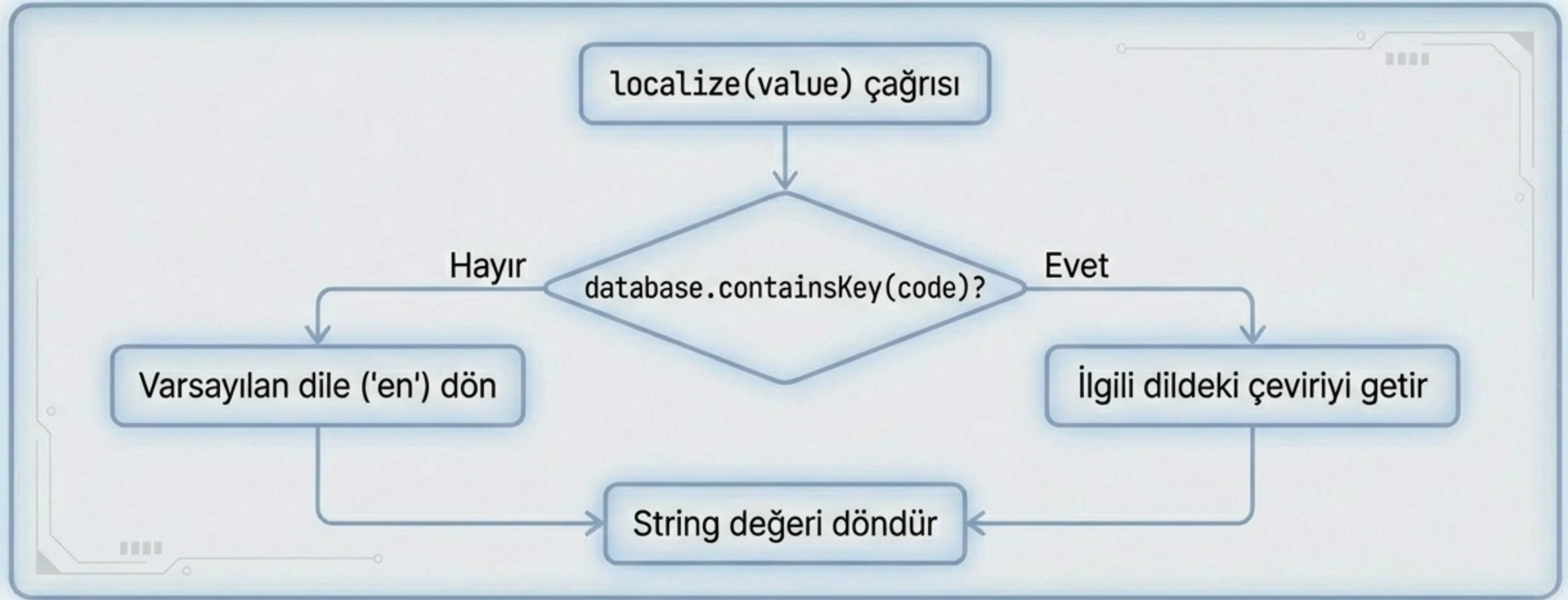
```
extension LocalizationExt on BuildContext {  
  String localize(String value) {  
    // Cihazın dil kodunu al veya varsayılan (en) kullan  
    final code = AppLocalization.of(this)?.locale.languageCode ?? "en";  
    // ... lookup logic ...  
  }  
}
```

Eski: `AppLocalization.of(context).localize("key")`

Yeni: **`context.localize("key")`**

Mantık Akışı: Fallback Mekanizması

Desteklenmeyen dillerde çökme nasıl engellenir?



Köprü: LocalizationDelegate

Temsilci sınıfı, cihazın diline göre çevirileri üretir.

```
class AppLocalizationDelegate extends  
  LocalizationsDelegate<AppLocalization>  
{  
  const AppLocalizationDelegate();  
  
  // ... override methods  
}
```

1. **isSupported:** "Uygulama bu dili destekliyor mu?"
2. **load:** "Çevirileri yükle."
3. **shouldReload:** "Yeniden yükleme gerekli mi? (Genelde false)"

Yükleme Stratejisi

Temsilci sınıfı çevirileri nasıl yükler?

```
bool isSupported(Locale locale) => ["en", "it", "es"].contains(locale.languageCode);
```

```
@override  
Future<AppLocalization> load(Locale locale) =>  
  SynchronousFuture(AppLocalization(locale));
```

Çeviriler bellekte (statik map) hazır olduğu için bekleme gerekmez. Standart Future yerine **SynchronousFuture** kullanılır.

```
bool shouldReload(LocalizationsDelegate d) => false;
```

Entegrasyon: main.dart

Delegelerin MaterialApp içine kaydedilmesi.

```
return MaterialApp(  
  // ...  
  localizationsDelegates: [  
    const AppLocalizationDelegate(), // Bizim sınıfımız  
    GlobalMaterialLocalizations.delegate,  
    GlobalCupertinoLocalizations.delegate,  
    GlobalWidgetsLocalizations.delegate,  
  ],  
  supportedLocales: const [  
    Locale("en"),  
    Locale("it"),  
    Locale("es"),  
  ],  
);
```

Kritik Uyarı: Dil Tutarlılığı

Delegate ve MaterialApp listeleri eşleşmelidir.

Delegate

```
bool isSupported(Locale locale)  
=> ["en", "it", "es"]  
    .contains(locale.  
    .languageCode);
```

Birebir Eşleşmeli

MaterialApp

```
supportedLocales: const [  
  Locale("en"),  
  Locale("it"),  
  Locale("es"),  
],
```

⚠ Uyumsuzluk durumunda Flutter'ın dil arama (lookup) stratejisi çalışmaz.

Geriye Dönük Uyumluluk

Extension desteği olmayan (Dart < 2.6) durumlar.

Temiz ve Kısa

```
Text(context.localize("hello"))
```

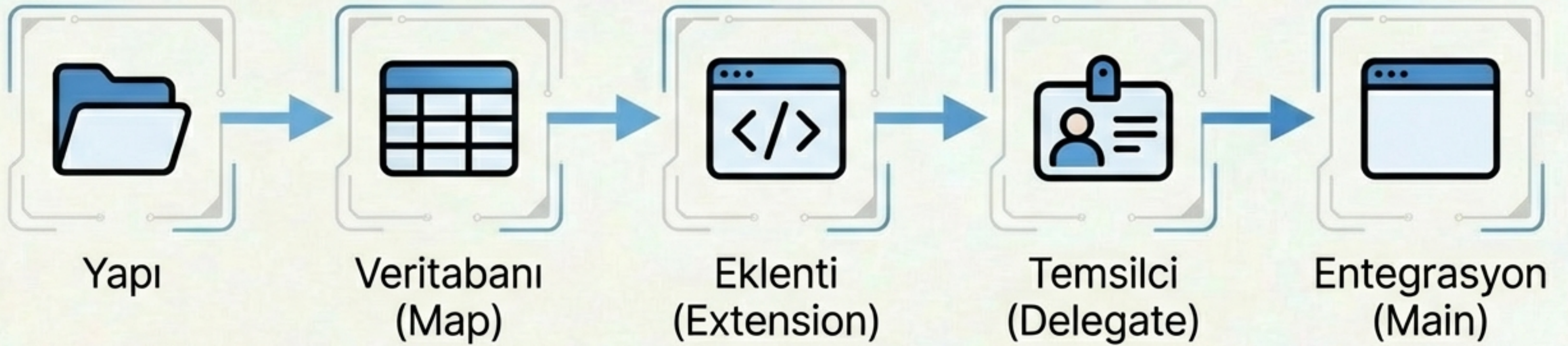
Daha Uzun (Verbose)

```
Text(AppLocalization.of(context).localize("hello"))
```



Dart 2.10+ Null-safety ile eski yöntem `?` ve `??` kontrolleri gerektirir.

Özet ve Sonuç



Küçük ve orta ölçekli projeler için hızlı, dışa bağımlılığı olmayan, güvenli ve tamamen **kontrol** edilebilir bir yerelleştirme çözümü.