

Angular 19 i18n Configuration & Concept Summary

Concept Summary: Angular 19 i18n (Internationalization)

- Angular 19 uses build-time localization with message extraction.
- All translations are extracted to a single messages.xlf file.
- Per-locale translation files must be created manually (e.g., messages.fr.xlf).
- Deprecated CLI options like `--i18n-locale`, `i18nFile`, `i18nLocale`, and `i18nFormat` should NOT be used.
- Angular 19 uses a centralized `i18n` block in `angular.json` for locale mappings.
- Locale-specific builds are defined using the `localize` array in `build.configurations`.

Correct i18n Block in `angular.json`:

```
{
  "i18n": {
    "sourceLocale": "en-US",
    "locales": {
      "fr": "src/locale/messages.fr.xlf"
    }
  }
}
```

Locale-Specific Build Configuration:

Placed inside:

`"projects" > "angular-internationalization" > "architect" > "build" > "configurations"`

```
"fr": {
  "localize": ["fr"],
  "outputPath": "dist/angular-internationalization/fr",
  "i18nMissingTranslation": "warning"
}
```

Serve Configuration for French:

Placed inside:

"projects" > "angular-internationalization" > "architect" > "serve" > "configurations"

```
"fr": {  
  "browserTarget": "angular-internationalization:build:fr"  
}
```

Commands:

- Extract messages: `ng extract-i18n --output-path src/locale`
- Copy and rename file: `cp src/locale/messages.xlf src/locale/messages.fr.xlf`
- Build French locale: `ng build --configuration=fr`
- Serve French locale: `ng serve --configuration=fr`

Deprecated and Removed:

- `ng xi18n`
- `--i18n-locale`
- `i18nFile` / `i18nFormat` / `i18nLocale` inside `build.configurations`

All locale mapping now must be done via the top-level `i18n` block.