

# Concept i18n mixed trees and multi-url support



+

Your Rating: Results: 96 rates

GREY

First, unfinished draft. Many points missing or open for discussion.

GREY

- [Rationale](#)
- [Examples](#)
- [Current status](#)
- [Problem analysis](#)
- [Proposed Solution](#)
  - [solution 1](#)
  - [solution 2](#)
  - [solution 3](#)
- [implementation details](#)
  - [Affected components](#)
  - [Missing components](#)
  - [Multisite support](#)
  - [Language views](#)
  - [\(De\)Activation of languages in different times](#)
  - [Configuration](#)
  - [Editors support](#)

## Rationale

More and more clients have need for solution that is easily maintainable and allows urls customized for target language while maintaining easy translation synchronization typical for single document tree solution. More sites also need to allow certain pages to exist only in specified languages.

## Examples

Michelin has same content structure in all countries, but some products are available only in Brasil while other are available everywhere Similarly Playstation releases some variations of the digital comics for some languages and regions first and some others only later.

## Current status

Current workaround employed by customers have been to use multiple trees (one per lang/locale) and try to keep them somehow in sync. This is cumbersome and leads to a lot of copy paste for content that doesn't need to be translated, leads to mistakes and inconsistencies caused by C&P and by humans.

Another kind of solution implemented by customers (Sony) is based on heavily customized templates rendering content from other language trees when content is not available in current tree. This removes need to duplicate the content but structure is very fragile and sensitive to changes.

## Problem analysis

i18n support is ad-hoc and is limited to i18n-ized properties. what is missing is

- activation by lang support
- security/editing rights by lang
- different urls per language
- language based tree browsing and preview
- inheritance would be affected as well
- possibility to define different components or whole areas per language

## Proposed Solution

### solution 1

- extra info in new section
- no default (w/o suffix) values for props (helps when switching lang later)

```
myPage [mgnl:page]
- main [mgnl:area]
  - 00 [mgnl:component]
- i18n
  - en [mgnl:i18n]
    - actDate=1.2.2012
    - uri=SamplePage
  - de [mgnl:i18n]
    - actDate=1.5.2012
    - uti=ProbeSeite
```

### solution 2

- change page structure
- extra level
- no suffix marking lang anymore
- again no default w/o lang values

```
myPage [mgnl:page]
  en [mgnl:i18n] <== i18n wrapper should make this level transparent
  main [mgnl:area]
    00 [mgnl:component]
  de [mgnl:i18n]
    mgnl:uri = meineSeite
    mgnl:actDate = 1.5.2012
  main [mgnl:area]
    00 [mgnl:component]
```

### solution 3

- do not set i18n on server/site level only, but allow override on page level and subpages
  - can we make this easy configurable and understandable for editors?
  - can it be changed at runtime from one setting to another? (the other 2 solutions are totally transparent)

## implementation details

### Affected components

- activation
- links
- tree view
- virtual uri mappings
- security

**Missing components**

**Multisite support**

**Language views**

**(De)Activation of languages in different times**

**Configuration**

**Editors support**