PLONE
SOLUTIONS

www.plonesolutions.com

Geir Bækholt

LinguaPlone

LinguaPlone is an extension to Plone for

making and managing multilingual web-sites

LinguaPlone is for managing multilingual

Content

4 audiences

- Site manager/content editor
- Site visitor
- Product developer
- Translator

Site manager/ content editor

- Create and manage translations of content
- Easily work and navigate with different language versions of the same content

Site visitor

- Provide content in a language the user understands
- Allow the user to override the automatic choice
- Present the entire site in one language

Product developer

- Make it easy to integrate custom content types
- Hooks for writing workflows for managing translations

Enable multilingual support

```
from Products.Archetypes.public import *
```

Enable multilingual support

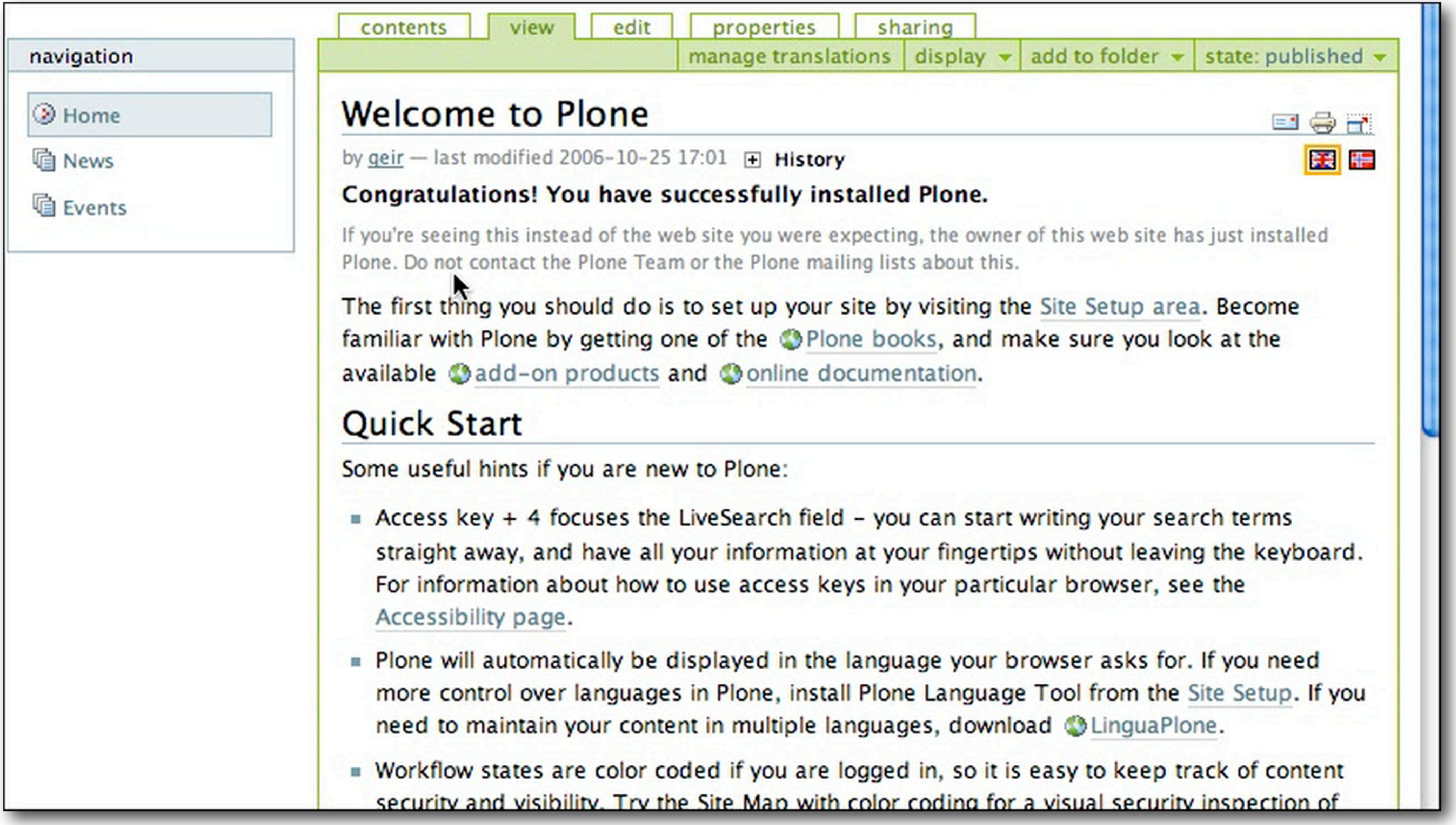
```
try:
    from Products.LinguaPlone.public import *
except ImportError:
    # No multilingual support
    from Products.Archetypes.public import *
```

Translator

- LP has simple user-interface for translating content in small sites
- Hooks for making translation workflows
- For larger systems; XLIFF

XLIFF

- For larger deployments / External translations
- XLIFF XML standard for translation tool
- Translation agencies
- Translate using MS Word or dedicated translation tool like TRADOS
- Import / exports of subtrees or full sites



Welcome to Plone

by geir — last modified 2006-10-25 17:01 History

Congratulations! You have successfully installed Plone.

If you're seeing this instead of the web site you were expecting, the owner of this web site has just installed Plone. Do not contact the Plone Team or the Plone mailing lists about this.

The first thing you should do is to set up your site by visiting the Site Setup area. Become familiar with Plone by getting one of the Plone books, and make sure you look at the available add-on products and online documentation.

Quick Start

Some useful hints if you are new to Plone:

- Access key + 4 focuses the LiveSearch field - you can start writing your search terms straight away, and have all your information at your fingertips without leaving the keyboard. For information about how to use access keys in your particular browser, see the Accessibility page.
- Plone will automatically be displayed in the language your browser asks for. If you need more control over languages in Plone, install Plone Language Tool from the Site Setup. If you need to maintain your content in multiple languages, download LinguaPlone.
- Workflow states are color coded if you are logged in, so it is easy to keep track of content security and visibility. Try the Site Map with color coding for a visual security inspection of

4 main approaches

- Fully translated sites +
- Separate country/region sites +
- Main site with child sites +
- Child sites with fallback to main/default -

Fully translated sites

- All content is available in all languages
- Dedicated translators
- Manage with
  - workflow and worklists
  - or with XLIFF exports/imports

Separate geographical sites

- “National” company sites
- Some common content, som separate
- Separate permissions for management
- No need to translate everything

Main site with secondary child sites

- The main site has all or most content
- Occasional translations or simply "brochureware" local sites
- "*More information available in Spanish…*" type sites

Child sites with fallback to main/default

- Mixes languages
- Possibly useful when all content must be available, but you have limited resources
- Makes assumptions about people knowing a ‘default’ language
- Not handled well by LinguaPlone…

Best Practices for Multilingual sites

- Don’t mix languages
- Make workflows
- Make someone responsible
- Commit resources
- Organisational processes

Don’t mix languages

Organisational challenges

- Make someone responsible
- Translation doesn’t just happen because it needs to be done.
  - Organisational processes are necessary
  - Workflow can help

Use workflows!

- Invalidate translations
- Keep content up to date

How does it work?

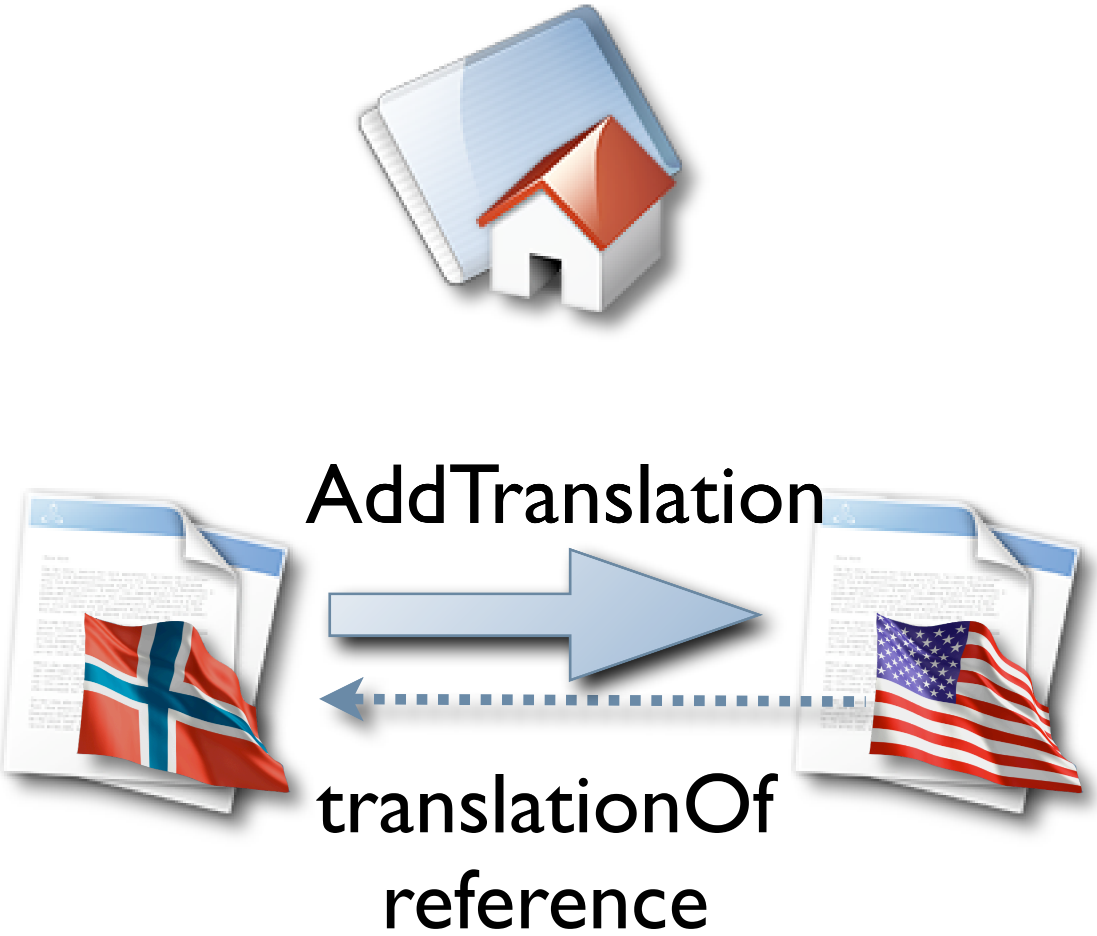
- Architecture level
- Aims to be as non-intrusive as possible
- Managing a multilingual site should **not** be fundamentally different…
  …neither for editors nor for users
- …nor for developers

Flags?

- Flags are not the same as countries or languages - but... Flags work
- Easy visual recognition
- Common understanding, even if technically incorrect
- Politically sensitive

Translation references

- ‘translationOf’ reference between two objects
- A normal Archetypes reference
- Stays even if LinguaPlone is uninstalled / reinstalled
- Not related to containment or ids



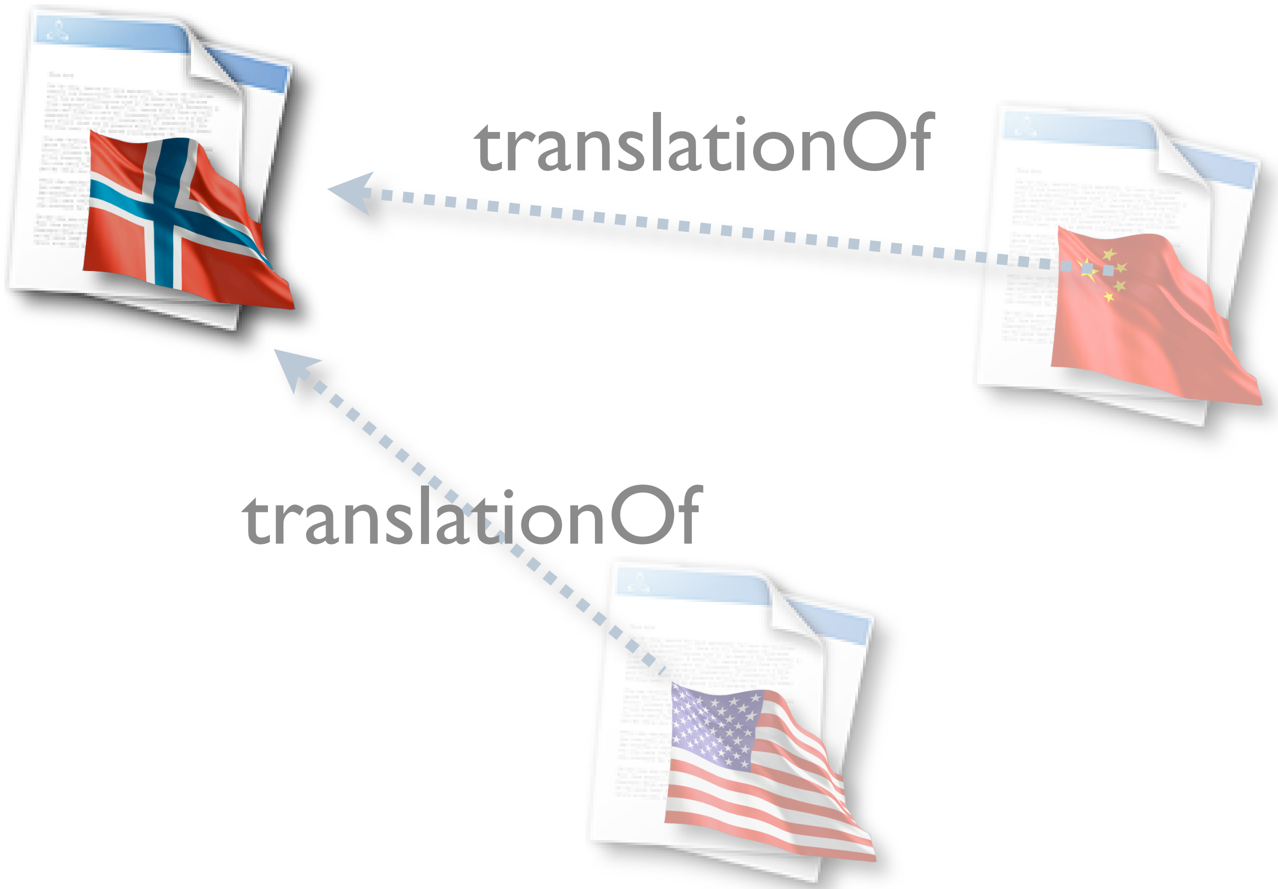
AddTranslation

translationOf
reference

Canonical

- The "master" or original piece of content
- Translations point back to the canonical

Canonical



translationOf

translationOf

Language-independent fields

- Field shared across translations
- Dates, numbers, locations, names etc.
- Updating canoncial updates all
- languageIndependent = True (in Schema)

Catalog changes

- portal_catalog patch
- Only returns results in the current language

Catalog used for…

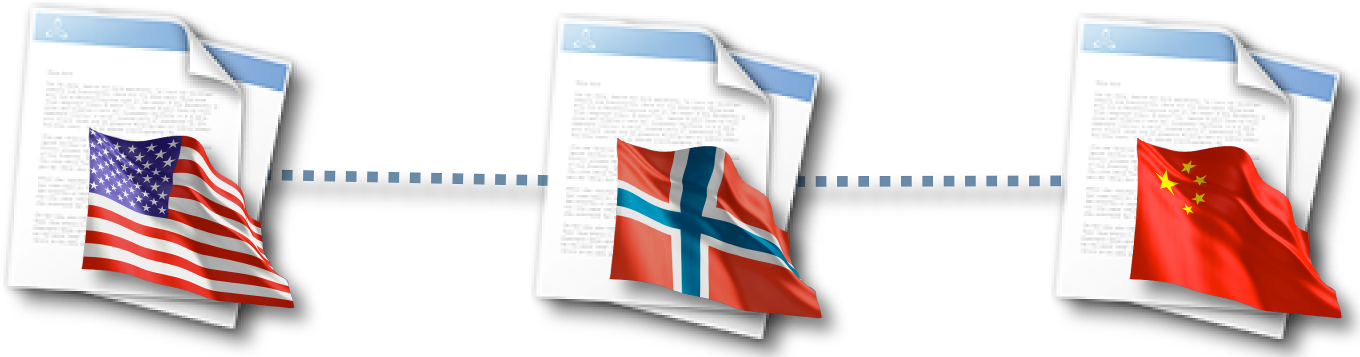
In Plone > 2.1

- Searching
- Navtree
- Folder listings
- Other API calls

Overriding Catalog filter

- `Language='all'`
- `Language=['en', 'de', 'no']`
- Objects that have no value for Language
- Automatic override for `Language, UID, getId, id`

Separate objects



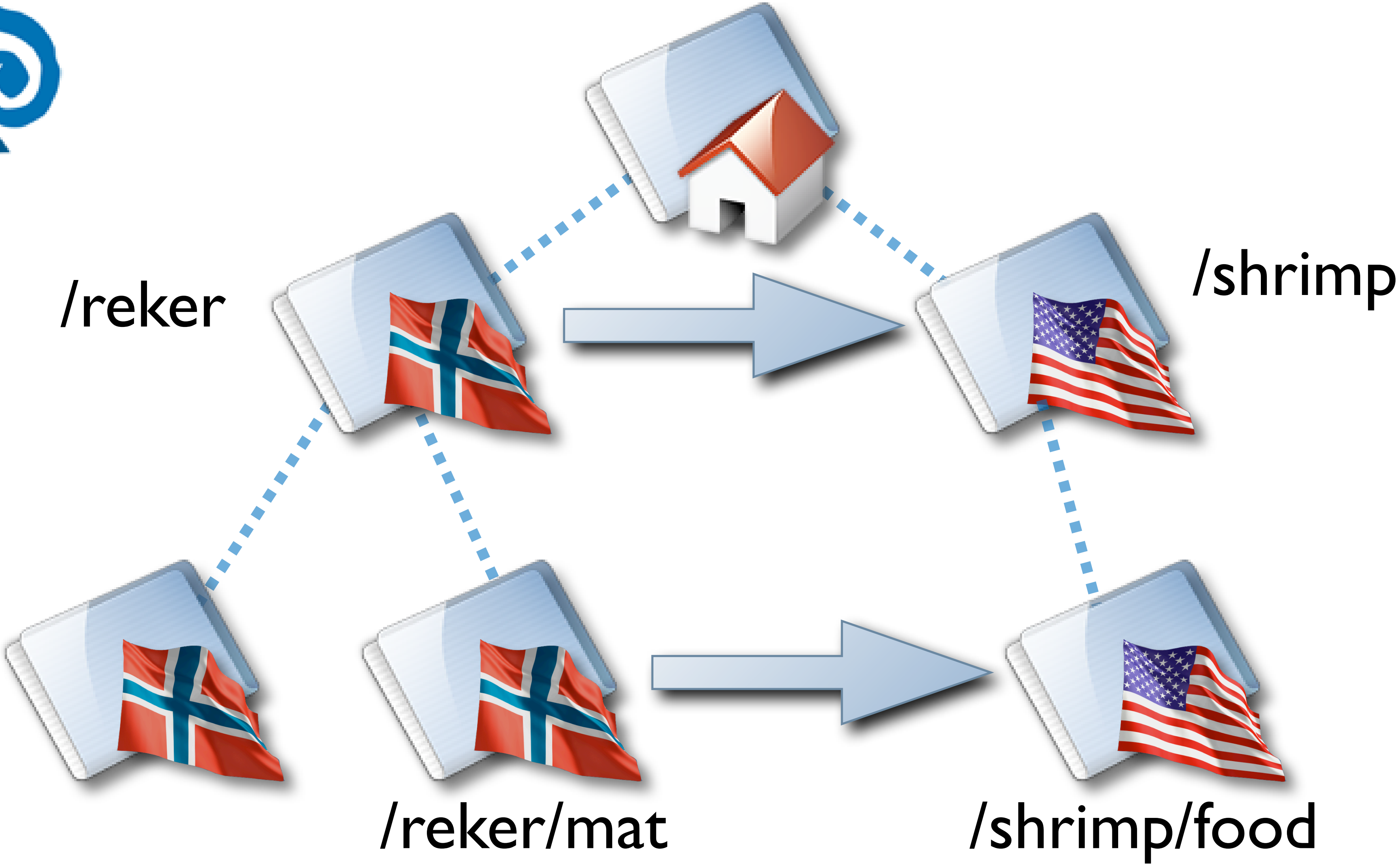
- Each translation is a separate object
- Security works as normal
- Workflow works as normal

Webdav / FTP / Marshall

- Separate objects, no magic
- Subtree for each language
- Should work reasonably transparently with
  - Webdav
  - FTP
  - Marshall

Subtrees

- Translating folderish objects creates new subtrees



/reker

/shrimp

/reker/mat

/shrimp/food

Multilingual URLs

- Result of automatic subtrees
- Plone 2.1 has automatic id-from-title-generation
- defaults to pretty, multilingual urls
- no more index_html-jp

Multilingual URLs

/en/	/no/
/en/shrimp/	/no/reker/
/en/shrimp/food	/no/reker/mat

Language negotiation order

1. Language codes in URL
2. http cookie for override.
3. browser language.
4. Default fallback

Workflow

- Hooks for when the canonical object is updated
- Allows for building translation workflows
- invalidate translations when editing canonical

ITranslatable

- The interface that describes multilingual content in Plone
- In Plone itself, not dependant on LinguaPlone
- All multilingual content-types should support this
- You can provide your own implementations of interface methods to have special behaviour

```
isTranslation()

addTranslation(language, **kwargs)

removeTranslation(language)

getTranslation(language='language')

getTranslationLanguages()
```

```
getTranslations()
 {lang : [object, wf_state]}

isCanonical()

getCanonicalLanguage()

getCanonical()

setLanguage(language)

Language()
```

Towards 1.0 and further

- Kill bugs
- Minor improvements
- Zope 3 technologies
- Bundled workflow

?

PLONE
SOLUTIONS

www.plonesolutions.com