

Pragmatic Plone Projects

PyCON –DE 2012
Leipzig

Andreas Jung

ZOPYX Ltd.

www.zopyx.com, info@zopyx.com

Speaker

- long-time Python, Zope, Plone developer and contributor
- Lead developer and principal consultant of ZOPYX Limited
 - Zope, Python, Pyramid, Plone projects
 - large portals, intranet, internet, extranet applications
 - Electronic Publishing
 - complex domain-specific applications

This talk is about

- practical hints surviving your next Plone project
- best practices
- lessons learned from our last larger Plone project

Relaunch weishaupt.de

- Weishaupt:
 - major vendor of heating systems
 - 480 M€ revenue
- Large company portal on top of Plone 4.2
- Phase 1:
 - implementation and DE content
 - 4 months
- Phase 2:
 - 20 subsidiaries
 - more than 20 language combinations

Funktionsweise der Wärmepumpe



► Heizungserneuerung



► Neubau



► Service 24/7

Gas-Brennwertgerät Weishaupt Thermo Condens WTC-GW (15 bis 60 kW)

Allgemein

Informationsmaterial

Wasssererwärmer

Referenzen

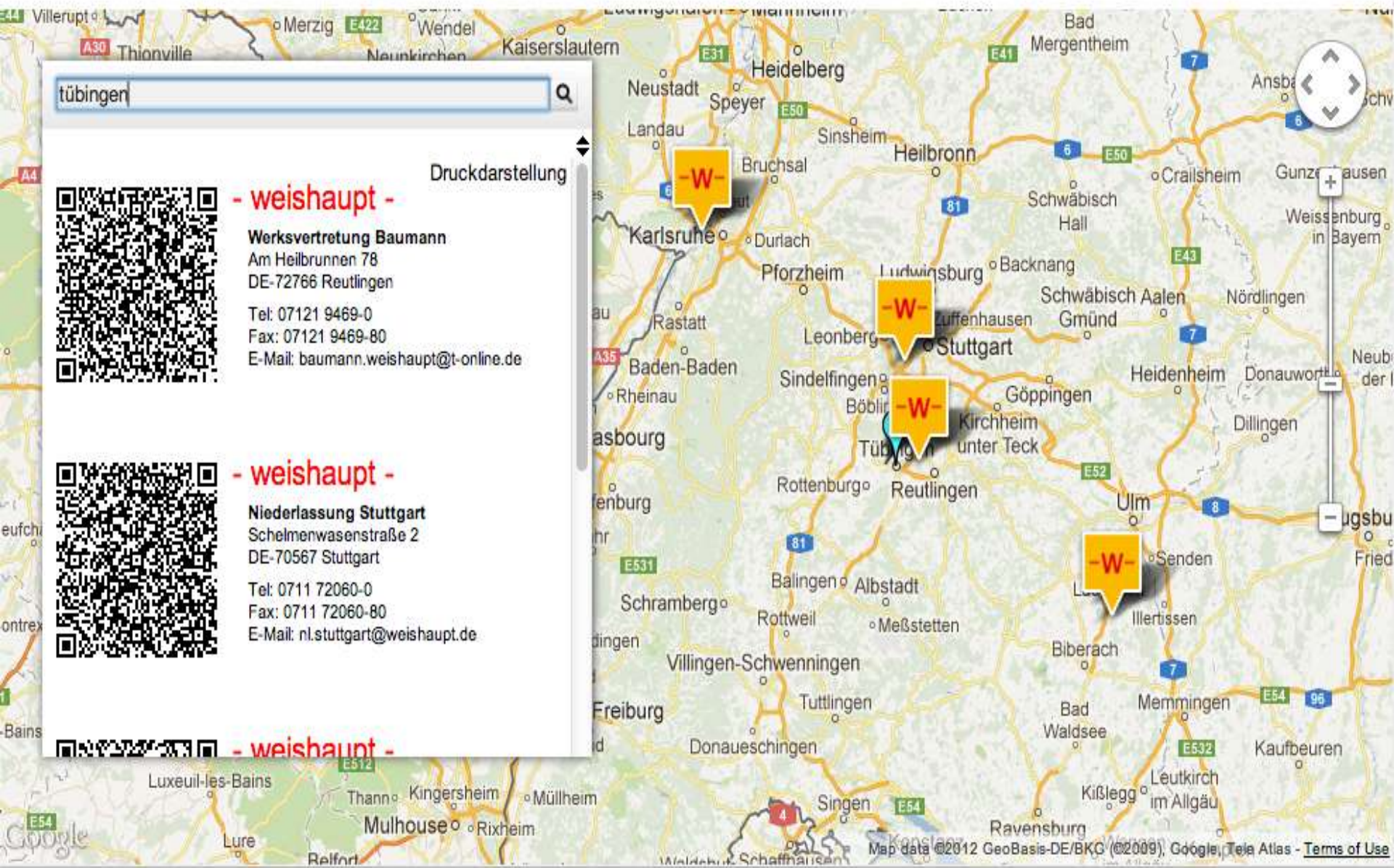
Beschreibung

Weishaupt Gas-Brennwertsysteme nutzen nahezu den vollen Energiegehalt des Brennstoffs und wandeln ihn effizient in Wärme um. Die Energie, die bei älteren Heizsystemen über die Abgase verloren geht, nutzt das Brennwertgerät und steigert damit den Norm-Nutzungsgrad im Vergleich zu Niedertemperatur-systemen von unter 100 auf rund 110 Prozent.



Weishaupt Thermo Condens WTC-GW

Weishaupt Niederlassungen



tübingen

Druckdarstellung



- weishaupt -

Werksvertretung Baumann

Am Heilbrunnen 78

DE-72766 Reutlingen

Tel: 07121 9469-0

Fax: 07121 9469-80

E-Mail: baumann.weishaupt@t-online.de



- weishaupt -

Niederlassung Stuttgart

Schelmenwasenstraße 2

DE-70567 Stuttgart

Tel: 0711 72060-0

Fax: 0711 72060-80

E-Mail: nl.stuttgart@weishaupt.de



- weishaupt -

Standorte International

Afrika

Asien & Naher Osten

Australien & Ozeanien

Europa

Nordamerika

Südamerika

- ▶ Belgien
- ▶ Bosnien Herzegowina
- ▶ Bulgarien
- ▶ Dänemark
- ▼ Deutschland

Weishaupt Hauptverwaltung

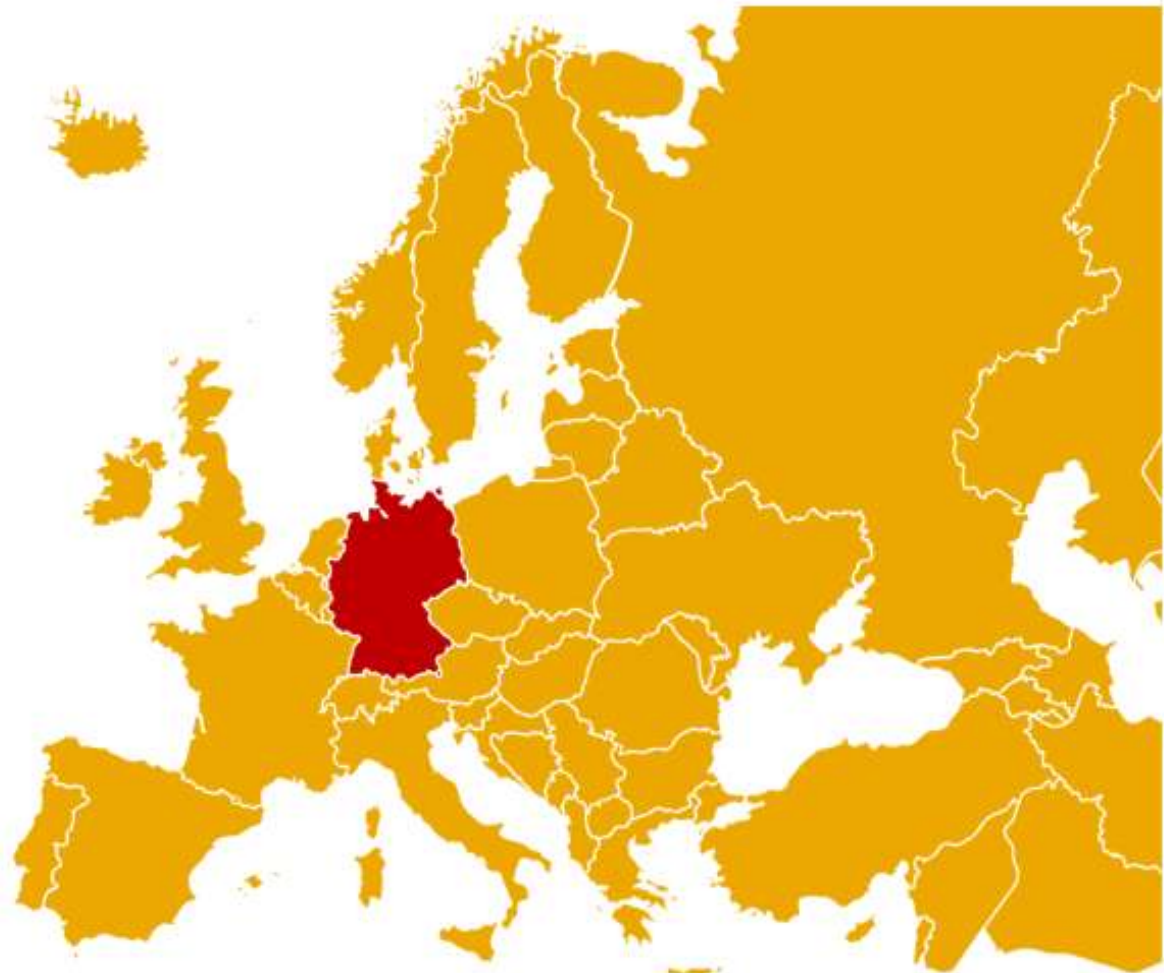
Max Weishaupt GmbH

Max-Weishaupt-Straße 14
88477 Schwendi

Tel +49 7353 83-0
Fax +49 7353 83-358

www.weishaupt.de
info@weishaupt.de

- ▶ Estland
- ▶ Finnland
- ▶ Frankreich
- ▶ Griechenland
- ▶ Großbritannien
- ▶ Irland
- ▶ Italien
- ▶ Kroatien
- ▶ Lettland
- ▶ Litauen



Weishaupt Luft/Wasser-Wärmepumpen (WWP L I) zur Innenaufstellung

Allgemein

Mediathek

Informationsmaterial

Wassererwärmer

Willi und die Wärmepumpe



Willi und die Wärmepumpe

Wie man Wärme aus der
Erde holt

07:04 min



Wärmepumpen für jeden Bedarf

04:57 min

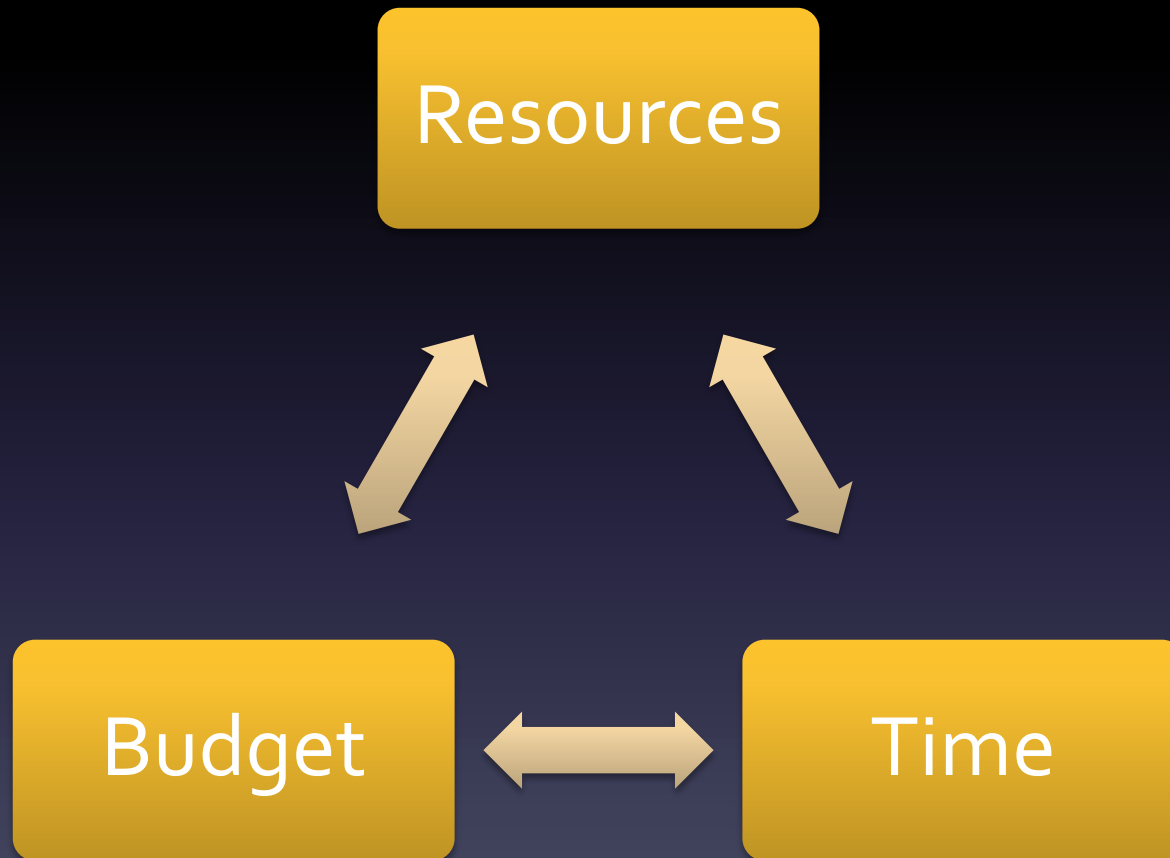


Project constraints (1/3)

There is no real-world project with

- unlimited budget
- unlimited human resources
- unlimited time

Project constraints (2/3)



Project constraints (3/3)

Conclusions

- usually impossible finding the „perfect“ solution
- the „perfect“ solution is „mission impossible“

Getting things done

- within a reasonable time-frame
- on budget
- with the available human resources

Common customizations

- Bugs in Plone and 3rd-party packages are all around you
- Particular behavior of Plone or 3rd-party packages is a common project requirement
- What must be usually customized:
 - Python code (browser views, skin scripts)
 - Resources (browser view templates, JS, CSS)

Where to **fix** things?

- Package maintainer (Plone and 3rd-party) love
 - bug reports
 - bug fixes
 - contributions
- Stuff on Github: Fork & Pull request
- SVN repositories: svn branch & svn merge
 - check if the package is maintained
 - ask the maintainer for merging your changes or merge yourself
 - move packages to Github if appropriate

Where and how to **customize** things?

- Is your required change of interest for the public?
 - fork on Github
 - branch in SVN
 - speak with the package maintainers
- Your change satisfies an absurd customer request?
 - keep it for yourself.

(Monkey) Patching Python code

- **Monkey patching:**

„dynamic modifications of a class or module at runtime“

(Source: Wikipedia)

- MP in general should be considered evil and bad-style
- MP may have side-effects
- Use MP carefully (and only when you know what you are doing)

Monkey patching in Python

Original (foo.py)	Patched version (my_foo.py)
<pre>class Foo: def bar(self): return 42</pre>	<pre>def my_bar(self): return 43 from foo import Foo Foo.bar = my_bar # or (needed in some situations) Foo.bar.func_code = my_bar.func_code</pre>

Monkey patching Plone

collective.monkeypatcher (patches.zcml)

ZCML-level configuration of patching information:

- module-level patches
- class-level patches

```
<configure...>
  <include package="collective.monkeypatcher" />
  <monkey:patch
    description="ISE-42: OFS.Image.tag()"
    class="Products.CMFCore.FSImage.FSImage"
    original="tag"
    replacement=".patches.tag"
  />
</configure>
```

collective.monkeypatcherpanel (patches.zcml)

Monkey Patches

Note that we show only monkey patches installed through **collective.monkeypatcher**. Other monkey patches may have been installed through other means.

ISE-42: OFS.Image.tag()

Original Products.CMFCore.FSImage.FSImage.tag

Replaced by weishaupt.policy.patches.tag

ZCML File "/home/ajung/sandboxes/weishaupt/src/weishaupt.policy/weishaupt/policy/patches.zcml", line 8.4-13.10

wh_skin cookie support

Original Products.CMFCore.Skinnable.SkinnableObjectManager.changeSkin

Replaced by weishaupt.policy.patches.changeSkin

ZCML File "/home/ajung/sandboxes/weishaupt/src/weishaupt.policy/weishaupt/policy/patches.zcml", line 49.4-54.10

Patching resources (1/2) – overrides.zcml

- Standard mechanism for overriding configuration settings introduced through a different package
- *overrides.zcml* is an optional ZCML configuration file

Products.PloneGlossary (configure.zcml)

```
<configure..>  
  <browser:page  
    name="glossary_main_page"  
    for="Products.PloneGlossary.IPloneGlossary"  
    class=".pages.GlossaryMainPage"  
    permission="zope2.View" />  
</configure>
```

my.package (overrides.zcml)

```
<configure..>  
  <browser:page  
    name="glossary_main_page"  
    for="Products.PloneGlossary.IPloneGlossary"  
    class=".glossary.GlossaryView"  
    permission="zope2.View" />  
</configure>
```

Patching resources (2/2) – z3c.jbot

- z3c.jbot allows you to override resources of other packages inside your own policy package
- Limited to .pt, .cpt, .js?!

Configuration: `<browser:jbot directory="overrides" />`

Existing template: `plone.app.search/plone/app/search/search.pt`

Customization: `your.package/your/package/overrides/plone.app.search.search.pt`

Links

- <http://blog.keul.it/2011/06/z3cjbot-magical-with-your-skins.html>
- <http://pypi.python.org/pypi/z3c.jbot>

Unconfigure resource configurations

- Sometimes you just don't want or need ZCML configurations introduced by other packages
- `z3c.unconfigure` is your friend

some.package(configuration.zcml)	my.package (configuration.zcml)
<pre><configure..> <browser:page name="glossary_main_page" for="Products.PloneGlossary.IPloneGlossary" class=".pages.GlossaryMainPage" permission="zope2.View" /> </configure></pre>	<pre><configure..> <include package="z3c.unconfigure" file="meta.zcml" /> <unconfigure> <browser:page name="glossary_main_page" for="Products.PloneGlossary.IPloneGlossary" class=".glossary.GlossaryView" permission="zope2.View" /> </unconfigure> </configure></pre>

Extending schemas

Plone comes with two content-type systems

- Archetypes (old-style)
 - use `archetypes.schemaextender`
 - modify any AT-based content-type (modifying fields, adding fields)
 - *SchemaExtender*, *SchemaModifier*
- Dexterity (new-style)
 - Dexterity introduces „behaviors“
 - „A behavior is a re-usable aspect of an object that can be enabled or disabled without changing the component registry“
(Source: Dexterity developer manual)

Keep your designer happy

Implementing CSS styles
requires representative content



<http://localhost:8080/@@new-site?content=1>

Auto-generated content

Wärmepumpen

Netus purus vitae morbi sit odio.

Class fames nostra pellentesque sodales curabitur enim. Proin justo integer conubia risus. Etiam hac cras arcu vel leo at pharetra. Class nulla vehicula. Lorem netus magnis accumsan. Lacus ipsum congue dictum quisque euismod eget lacus integer. Lacus proin. Purus justo lacus vivamus montes eu. Proin augue donec lobortis nunc. Justo lorem etiam pede justo sollicitudin malesuada tellus morbi vestibulum suspendisse luctus. Proin vitae massa quis sociis fames vitae proin pulvinar netus. Etiam aenean egestas massa ipsum.

Fachwortlexikon

Aenean fames integer blandit nostra velit augue platea at.

Messekalender

Messen Europa

Datum	Titel der Messe	Ort/Land	
02.01.2013 - 04.01.2013	Risus neque maecenas.	Schwendi	► Details zur Messe
14.07.2013 - 16.07.2013	Etiam morbi orci venenatis eu dignissim.	Schwendi	► Details zur Messe
06.09.2013 - 08.09.2013	Lacus felis nibh.	Schwendi	► Details zur Messe
20.10.2013 - 23.10.2013	Nulla class libero justo adipiscing lobortis.	Schwendi	► Details zur Messe
18.10.2014 - 19.10.2014	Justo lacus primis mollis urna sociis in enim litora.	Schwendi	► Details zur Messe
25.04.2015 - 26.04.2015	Vitae class tellus maecenas venenatis neque sed fusce massa.	Schwendi	► Details zur Messe
24.03.2016 - 25.03.2016	Velit lacus nulla dis metus.	Schwendi	► Details zur Messe
04.04.2016 - 07.04.2016	Netus velit scelerisque suscipit phasellus fusce.	Schwendi	► Details zur Messe

Predefined sample content

- Write a browser view
 - creating a Plone site with policy package + add-ons
 - installing the basic site-structure
 - creating example content for each content-type, content-listing etc
- use <http://loempixel.com/600/400> ...
- look at *loremipsum*, *collective.loremipsum* or *zopyx.ipsumplone*

Predefined sample content

- Throw your sandbox/Plone working site away as often as possible
- sometimes I created 30-40 new Plone sites per day
- Pragmatic side-effect
 - the content fixture code can be used as unit test where all your content-types and site-infrastructure is created and tested in one run
 - not the best solution but it works reasonably well

Some good hints

- **Never ever** perform customizations in-place in existing 3rd-party packages. **NEVER!!!**
- Customizations **always** belong into your own policy package.
- Local customizations of 3rd-party package will be lost with the next version of customized package.

Questions?