World Domination (the easy way!)

i18n + l10n

Sarina Canelake

PyCon 2015

sarina.github.io

A Villain's Monologue

My Motives & Plan

Ready player 1?

- i18n/l10n basics

- Presenting your project in
  a new language

- Help your community & leverage
  crowd-sourced translation work

In the year 20XX...
Some Background

What is i18n?

Internationalization:
(i + 18 letters + n = i18n)

What is i18n?

Engineering support to enable worldwide content presentation

What is i18n?

Performed once, as part of feature development

What is l10n?

Localization:
(l + 10 letters + n = l10n)

What is l10n?

Adapting i18n'd software for a specific locale

What is l10n?

Performed multiple times: once per target region

Advantages of i18n

Enable future l10n to increase your reach

Advantages of i18n

Be a good open source citizen!

World 1-2
The l10n Pipeline

Step #1

Mark strings for i18n

Wrap strings with markers to indicate they need to be translated

```python
def my_cool_function():

    welcome_msg = "Hello!"
```

```python
from django.utils.translation
                    import ugettext as _


def my_cool_function():

    welcome_msg = _("Hello!")
```

Scrape the strings

Statically scrape marked strings into a ".po" file

Django Toolkit

```
# Scrape strings
django-admin.py makemessages
```

Translate the strings

Create a .po file for each language

TIP: Knock It Out with Translation Tools

Transifex

Free crowdsource accounts for open source projects

https://www.transifex.com

```
$ cat fr/django.po

#: views/awesome_feature.py:6
msgid "Hello!"
msgstr "Bonjour!"
```

Serve the strings

Compile your "po"s to "mo"s

Django Toolkit

```
# Scrape strings
django-admin.py makemessages

# pos to mos
django-admin.py compilemessages
```

It's Time to Run

```
_("Hello!")
```

↓

```
mo_file.get("Hello!", "Hello!")
```

RECAP MINIBOSS

1. Mark strings for i18n
2. Scrape the strings
3. Translate the strings
4. Serve the strings

Watch out for sucker punches

A Note of Caution

_ has static and dynamic contexts

```
def my_cool_function():

    welcome_msg = _("Hello!")
```

Lexical Marker AND Function Call!

```python
msg = "Hello {name}!".format(
    name=user.name
)
```

```
msg = _(
    "Hello {name}!".format(name=user.name)
)
```

.po file has key "Hello {name}!"
At runtime, what's looked up in .mo file?

```
msg = _("Hello {name}!").format(
    name=user.name
)
```

.po file has key "Hello {name}!" AND
same key will be looked up in .mo file ☺

Its Dangerous to Go Alone
i18n Tools

Django toolkit

Docs:
bit.ly/django-i18n

Matt Croyden, PyCon 2010:
bit.ly/croyden-i18n-talk

Mako
+ Underscore
+ Coffeescript
+ JavaScript
+ etc...

Advanced applications

- various file types
- po segmenting
- dümmý längüägé generation

github.com/edx/i18n-tools

"Frobulate the %d %s!"

Lost in Translation

Translators...

may be non-technical

Translators…

may not have seen the UX of the rest of your application

Consider the Translator

Marking strings is worthless without context

Consider the Translator

UNTRANSLATED STRING

Could not find the {0} file in the package.

There are no instructions yet. Click here to add.

Type your translation here

Review   Save   Save all (0)

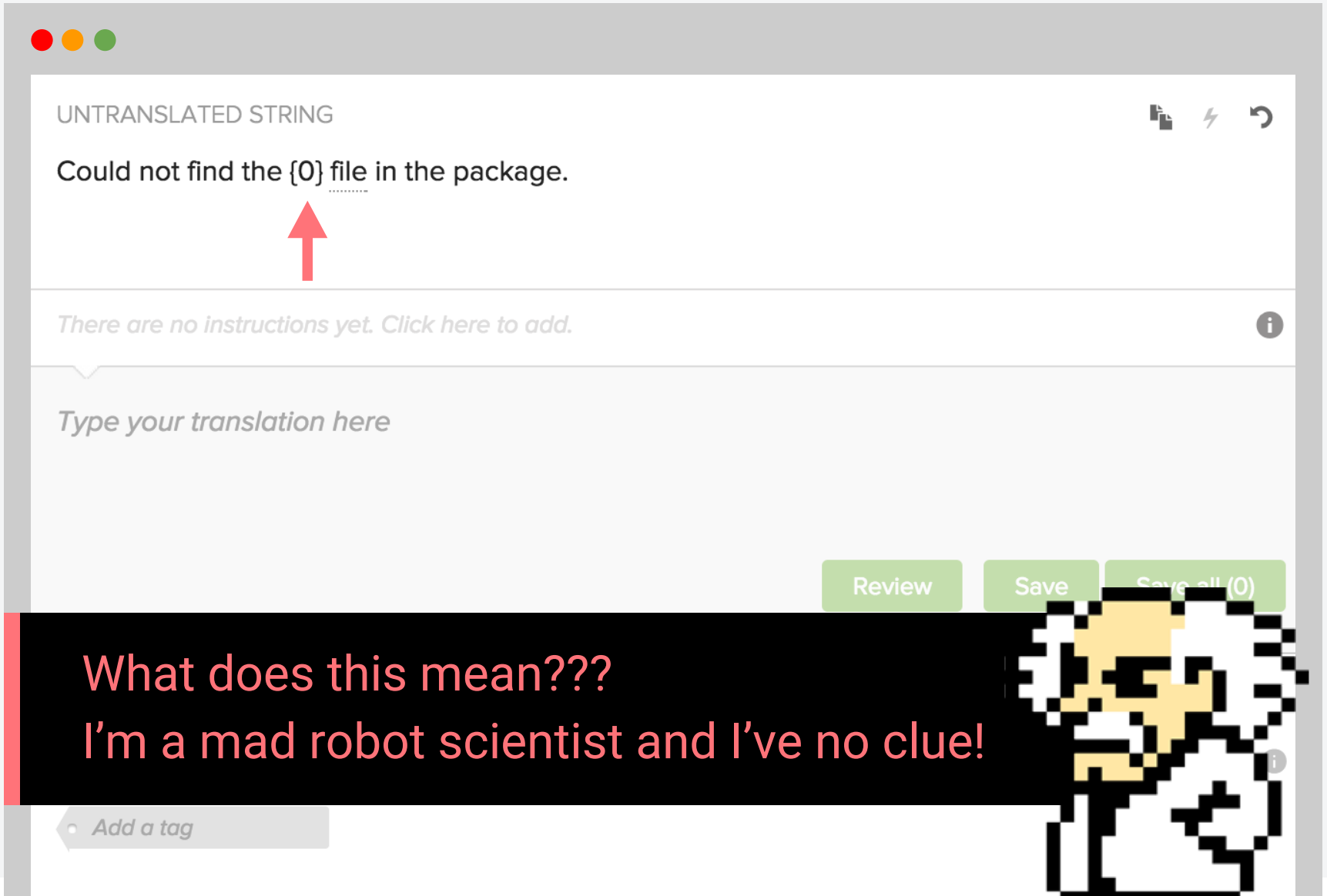**Suggestions**   **History**   **Glossary**   1   **Comments**   **Details**

Character limit: *None*

Add a tag

Consider the Translator

UNTRANSLATED STRING

Could not find the {0} file in the package.

There are no instructions yet. Click here to add.

Type your translation here

Review   Save   Save all (0)

Add a tag

What does this mean???
I'm a mad robot scientist and I've no clue!

Consider the Translator

TRANSLATED BY ⌐, 9 MONTHS AGO.

Failed in authenticating {username}↵

There are no instructions yet. Click here to add. ⓘ

{username} の認証に失敗しました

Review   Save   Save all (0)

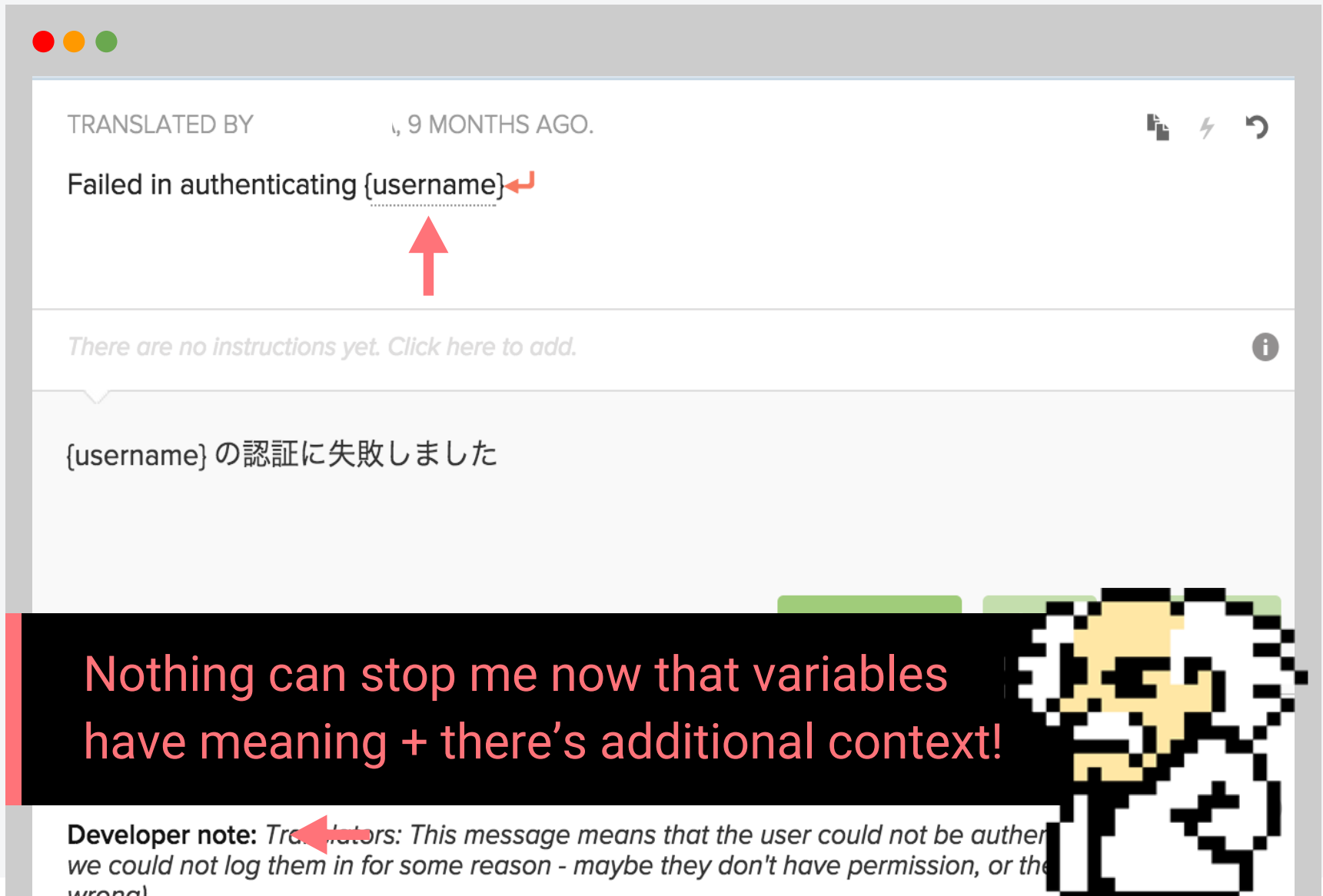Suggestions **5**   History **1**   Glossary **1**   Comments   **Details**

ⓘ

Developer note: *Translators: This message means that the user could not be authenticated (that is, we could not log them in for some reason - maybe they don't have permission, or their password was wrong).*

Consider the Translator

TRANSLATED BY   ı, 9 MONTHS AGO.

Failed in authenticating {username}↵

There are no instructions yet. Click here to add.

{username} の認証に失敗しました

Nothing can stop me now that variables have meaning + there's additional context!

Developer note: *Translators: This message means that the user could not be auther we could not log them in for some reason - maybe they don't have permission, or the wrong).*

Secrets of World Dominators
Pro Tips for i18n

PRO TIP: Basic Rules

Use `.format` over `%s` & `%d` interpolation

Use named, not positional, variables
(`{user}` vs `{}` or `{0}`)

Add `# Translators:` comments

```python
# Translators: This message means that the
# user could not be logged in to the site
msg = _(
    "Failed in authenticating {username}\n"
).format(
    username=eamap.uname
)
```

PRO TIP: Named Variables

```
_("Today is {} {}")
```

PRO TIP: Named Variables

```
_("Today is {} {}")
            ↑   ↑
```

Doesn't allow reordering!
Use "{month} {day}"

PRO TIP: Keep Phrases Together

```
_("That is a ") + car.color +
    _(" car")
```

PRO TIP: Keep Phrases Together

```
_("That is a ") + car.color +
    _(" car")
```

↓

En: "That is a blue car"
Fr: "C'est une bleue voiture"

Sacré Bleu!!!

PRO TIP: Keep Phrases Together

```python
_(
    "That is a {color} car"
).format(color=car.color)
```

↓

En: "That is a blue car"
Fr: "C'est une voiture bleue"

PRO TIP: Plurals with ngettext

```python
from django.utils.translation
                    import ungettext


msg = ungettext(
    "There is {num} file",
    "There are {num} files",
    num_files
).format(num=num_files)
```

PRO TIP: Plurals with ngettext

one | other

REVIEWED. TRANSLATION BY   , 2 MONTHS AGO

You only have {days} day left to verify for this course.

There are no instructions yet. Click here to add.

RTL

zero | one | two | few | many | other

تبقى لديك {days} أيام فقط لتثبيت هذا المساق.

Unreview | Save | Save all (0)

Fill in all **plurals** to continue

PRO TIP: Plurals with ngettext

```python
if num_tags >= 2:
    msg = _(
        "There are {num} retweets!"
    ).format(num=num_tags)
```

PRO TIP: Plurals with ngettext

```python
if num_tags >= 2:
    msg = _(
        "There are {num} retweets!"
    ).format(num=num_tags)
```

Darny darn darn! This breaks on languages with differing plural forms.

PRO TIP: Plurals with ngettext

```python
if num_tags >= 2:
    msg = ungettext(
        "There is {num} retweet!",
        "There are {num} retweets!",
        num_tags
    ).format(num=num_tags)
```

PRO TIP: Remove HTML from Strings

`"<em>Welcome!</em>"`

↓

`u"<em>" + _("Welcome!") + u"</em>"`

PRO TIP: Remove HTML from Strings

```
"Confirm <em>{action}</em>?"
```

↓

```
_("Confirm {action}?").format(
  action=u"<em>" + action + u"</em>"
)
```

PRO TIP: Remove HTML from Strings

```
"<p class='blah'>Lots of words!</p>"
```

↓

```
u"<p class='blah'>{text}</p>".format(
        text=_("Lots of words!")
)
```

PRO TIP: Don't Leave Order to Chance

```
# Translators: Leave asterisk at end!
_("Password *")
```

↓

```
_("Password") + u" *"
```

PRO TIP: Try to Keep Strings Constant

All changes to strings create new entries in .po files

"login", "Login", "Log In"

PRO TIP: Take a Hit for the Team

Complex code
is better than
inscrutable translation
strings

Welcome to the Warp Zone

i18n Bonus Levels

Fight for the User (Experience)

Non-Latin characters may be taller, wider

给你点颜色看看!!!

German, Dutch can have extremely long words like...

Die deutsche Bevölkerungspyramide

Verflixt und zugenäht!!!

Be multi-directional with Right to Left (RTL) support

عربي، תירבע، ودرا، يسراف

يا ساتر!!!

Fight for the User (Experience)

Responsive Design & Layout/Grid Management

Fight for the User (Experience)

bi-app Sass:
anasnakawa.github.io/bi-app-sass

Neat Grid Support:
neat.bourbon.io

Keep it Local with Localization

Don't forget
dates & numbers!

Larger & Commercial Projects

Hire a Localization Manager

Pay for translation services

Lost Levels (and Resources)

- ## Unicode sandwich
  bit.ly/unipain

- ## edX's i18n guidelines
  bit.ly/edx-i18n-rules

- sarina.github.io

Thanks for playing!

Now for Discussion

sarina@edx.org
github.com/sarina
@robotsari