

# The only option is open: Why should language technology and resources be free?

Francis M. Tyers

<sup>1</sup>Departament de Llenguatges i Sistemes Informàtics,  
Universitat d'Alacant, E-03071 Alacant, Spain

11th May 2011

- ▶ Introduction to **who am I** and **why I am here**
- ▶ Description of some problems we face when developing and working with language resources and technology
- ▶ Introduction to **software and resource pools**
- ▶ Description of how these solve many of the problems described
- ▶ Discussion of some **commercial aspects** to pools
- ▶ Conclusions

# Who am I?

I...

- ▶ am a PhD student at the Universitat d'Alacant
- ▶ have been working with free software for approx. ten years
- ▶ and language technology for around five years
- ▶ spend most of my time working on machine translation,
- ▶ but also (necessarily) work on other language technology

# Why am I here?

I was asked to give a talk about the importance of

- ▶ free dissemination and access to resources
- ▶ both within a single language and multilingually

Additionally,

- ▶ I would like to describe my experiences as a developer of free/open language technology

# Why am I here?

I was asked to give a talk about the importance of

- ▶ free dissemination and access to resources
- ▶ both within a single language and multilingually

Additionally,

- ▶ I would like to describe my experiences as a developer of free/open language technology
- ▶ ... I hope it doesn't get too tiresome

# What are language resources and technology?

Often the terms are used interchangeably,

- ▶ **Language resources:** Data with which language processing applications are made.
  - ▶ Ex.: A machine-readable dictionary, a treebank or parallel corpus
- ▶ **Language technology:** Software with which language processing applications are made.
  - ▶ Ex.: A machine translation engine, parser or spellchecking engine

Sometimes in natural language processing it is difficult to separate data from software.

# What is free and open?

These are the four essential freedoms published by the Free Software Foundation (FSF).

- ▶ **Freedom 0:** The freedom to run the program, for any purpose
- ▶ **Freedom 1:** The freedom to study how the program works, and change it to make it do what you wish
- ▶ **Freedom 2:** The freedom to redistribute copies so you can help your friends and neighbours
- ▶ **Freedom 3:** The freedom to distribute copies of your modified versions to others

Open access to source code is a precondition to freedoms 1 and 3, which is why it is also called *open-source*.



























## **Problems for language resources**

(the ones not on the previous list)

# Major problems for language resources

- ▶ **Visibility:**

“When I go looking for a resource, can I find it?”

- ▶ Ex.: Swedish WordNet

- ▶ **Availability:**

“When I find the resource I’m looking for, can I use it?”

- ▶ Ex.: Beygingarlýsing íslensks nútímamáls (BÍN)

- ▶ **Sustainability:**

“Will the resource I’m using be there in next year?”

- ▶ Ex.: METIS-II project

# Problem: Visibility

So, why is **visibility** such a problem ?

- ▶ As **researchers**, we would like to be sure that our work is known
  - ▶ For example, people citing us
- ▶ As **users**, we would like to be able to find resources easily.
  - ▶ The longer it takes to find, the more we are going to assume that it doesn't exist
- ▶ As **developers**, we would like the work that we produce to be used.
  - ▶ Usage means feedback, and feedback means improvements
- ▶ As **funding bodies**, we want to effectively spend our money
  - ▶ Not unnecessarily duplicating something that already exists

# Problem: Availability

## And **availability** ?

- ▶ As **researchers**, we want to be able to reproduce the results of others
  - ▶ Many papers are difficult or impossible to reproduce – cf. “Zigglebottom tagger” (Pedersen, 2008)
- ▶ As **users**, we want to be able to use what we find with the minimum of hassle
  - ▶ Both standalone, and in combination with other software.
- ▶ As **developers**, we don't want to have to recreate something that has already been made
  - ▶ Especially if it has been made with public money
- ▶ As **funding bodies**, we don't want to fund the same thing twice
  - ▶ We also want the results of one project to be able to be used in the development of another

# Problem: Sustainability

And **sustainability** ?

- ▶ As **researchers**, we want to be able to reproduce our work (and that of others), even after fifteen years
  - ▶ Our results are highly dependent on specific versions of data and code
- ▶ As **users**, we want to be able to depend on software
  - ▶ If we build a service or software around it, we don't want it to disappear overnight
- ▶ As **developers**, we want our code to keep working
  - ▶ We might not have time to keep up with the changes in library versions, but someone else will
- ▶ As **funding bodies**, we want the results of our investments to be relevant for as long as possible
  - ▶ That's cost effective

## Particular issues for M-languages

In my opinion, of the three problems outlined, for marginalised and minority languages, the biggest one is **availability**.

- ▶ Less “noise”, when there is only one of something it sticks out more.
- ▶ Projects tend to be “labours of love” for the participants, meaning that they are more likely to have staying power

Availability is a problem why ?

- ▶ M-languages often barely have the base to make their own resources, let alone duplicate resources of a major language as well.
- ▶ While major languages can often afford to rewrite resources several times, this is rarely the case for M-languages.

# Why do these problems occur ?

These problems occur as a result of how language resources and technology are developed and published. Here are some common experiences:

- ▶ **Commercial:** A company makes a resource, and sells licences for it in the usual fashion.
- ▶ **Big research:** A big consortium or group develops a resource, and charges for its use. Sometimes with public money, sometimes without.
- ▶ **Small research:** A small group develops something, and publishes it on their university web. They mark it as *research only/non commercial*
- ▶ **Single person:** A lecturer or student develops something, and as above.

## **Free/open-source pools**

What is a pool (Scannell, 2006) ?

**“ A pool is a multilingual collection of resources of the same form and function under a free/open licence ”**

Features of a pool:

- ▶ **Multilingual:** To be a pool, the collection must be multilingual
- ▶ **Community maintained:** The collection must have an active community of unpaid users, developers and maintainers
- ▶ **Open to contribution:** The collection must allow external contribution
- ▶ **Uniform:** The collection should be homogenous as far as possible, dictionaries with dictionaries, taggers with taggers
- ▶ **Free / open:** Everything in the collection should be released under free/open licences, ideally the same one<sup>1</sup>

---

<sup>1</sup>More on this later

# Existing pools

Here are some examples of existing pools:

▶ **Corpora:**

- ▶ OPUS: Open Parallel Corpus
- ▶ EuroParl: Corpus of European Parliament Proceedings

▶ **Grammars:**

- ▶ DELPH-IN: Collection of HPSG grammars
- ▶ MOLTO: Grammars based on Grammatical Framework (GF)

▶ **Morphological analysers:**

- ▶ Giellatekno: Morphological analysers for the Sámi languages, and others
- ▶ Apertium: Morphological analysers for a range of languages

▶ **Spellcheckers:**

- ▶ {A,I,Hun,My}spell: Ubiquitous free spellcheckers

▶ **Machine translation systems:**

- ▶ Apertium: Simple RBMT systems for many languages
- ▶ MOLTO: Interlingua-based MT systems

# What is not a pool?

The definition of *pool* is quite wide, but does not include the following:

- ▶ ELRA/LDC: Sparse databases of some existing language resources
- ▶ CORPORA-LIST: Mailing list for corpora and language resources in general
- ▶ ACLWIKI: A collaboratively-maintained Wiki of pointers to language resources

**The solution**

When you add your resource to an existing collection you gain visibility.

- ▶ The resource is with similar resources, so is more likely to be referred to in passing.
  - ▶ Higher search engine rankings
- ▶ The more languages a collection has, the more visible it is, so it is in the interest of the maintainers to have more languages.

All of the resources in the pool have compatible licences

- ▶ Resources and software can be shared between projects
  - ▶ No need to call in the lawyers
- ▶ No need to wait years for someone to finally make a decision
- ▶ Results can be reproduced and new results published without issue

And they are all hosted on the same project-independent (and mirrored) infrastructure

The pool is project-independent, and community maintained

- ▶ It can use freely available infrastructure... for example GNU Savannah, SourceForge, Google Code, La Farga, ...
  - ▶ No problem of the servers being turned off when the project ends
- ▶ A lot of maintenance is done by people who aren't tied to project funding – although they may be being paid
- ▶ Data conversion happens automatically
- ▶ Feedback about one language can be used to improve all languages

## Particular issues for M-languages

M-languages can gain a lot by being pooled – either with major languages – or with other M-languages, or both.

- ▶ No need to build their own infrastructure – spend more time on linguistic matters
- ▶ Sharing of infrastructure and expertise
- ▶ To make major language ↔ M-language applications, both free M-language *and* free major language resources are necessary.

## **Commercial aspects**

How does the pool (i.e. being free/open) effect commercial use and sales ?

- ▶ **Commercial use:** Explicitly allowed by the first freedom.
- ▶ **Commercial sales:** Allowed with permission of all authors.

Many companies dual-license their software or resources, with

- ▶ A **free licence** for use with other free/open software, and
- ▶ A **commercial licence** for use with proprietary/commercial software

Brass tacks: You can still sell licences (if you own all the rights).

There are a number of things to consider when choosing a licence

- ▶ **Type:** Is the thing to be licensed code, or data ?
- ▶ **Copyleft:** Should all changes be released under the same licence ?
- ▶ **Compatibility:** Is the licence compatible with other software ?

And there is one thing that shouldn't be considered:

- ▶ **Non-commercial:** Limiting the commercial use of your resource, or limiting it to *research only*

Why ?

## What about *non-commercial*?

What other people think of when they think of *commercial use*:

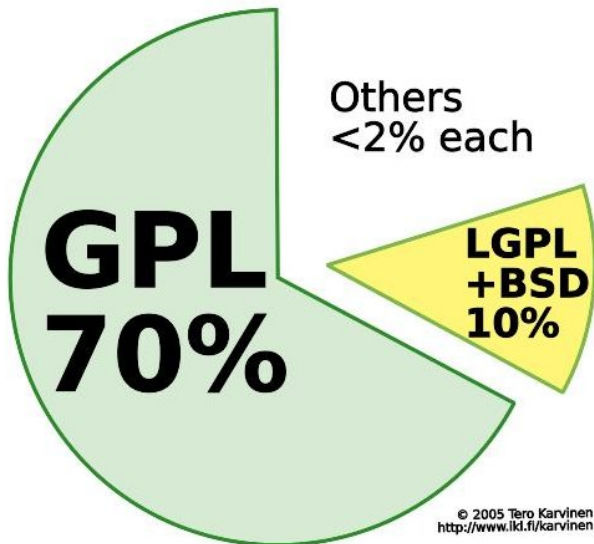
- ▶ Microsoft™ is going to come along and take my Finnish-North Sámi wordlist, make a dictionary and sell it for a million euros. I will die poor and penniless.

What I think of:

- ▶ A company offering bundled versions of OpenOffice and a spellchecker to schools on easy-to-install DVDs. The DVD price covers costs. The company makes its main profit offering services.
- ▶ A minority language newspaper wants to produce bilingual editions of some of their articles. They download an MT system and use it in their commercial operation.

Outside academia, almost everything can be classified as commercial.

# Which licence should I choose?



- ▶ Do not hesitate to dual license
- ▶ **The language comes first!**
  - ▶ If the only way to get your software onto the desktops of M-language users is to pact with the devil, do it.

I see two main avenues for the development of language technology and resources

▶ **Open / free**

- ▶ Sharing
- ▶ Easy collaboration
- ▶ Linguistically rich applications
- ▶ Inclusion of all languages

▶ **Closed / proprietary**

- ▶ Duplicated work
- ▶ Reduced collaboration
- ▶ Dependence on linguistically-poor techniques
  - ▶ Spellcheckers working on simple wordlists
  - ▶ “Basic” phrase-based statistical MT
- ▶ Inclusion of *profitable* languages

- ▶ Pedersen, T. (2008) 'Empiricism Is Not a Matter of Faith'. *Computational Linguistics* 34(3), 465–470.
- ▶ Scannell, K., Streiter, O. and Stuflessner, M. (2006) 'Implementing NLP Projects for Non-Central Languages: Instructions for Funding Bodies, Strategies for Developers' *Machine Translation* 20(4), 267–289

**Pieldes ! · Tack ! · Tak ! · Takk fyrir ! ·  
Takk fyri ! · Kiitti ! · Gæhjtøe ! · Ačiū ! ·  
Giitu !**