

Developing an API for PanLex

Jonathan Pool

Computational Linguistics Group
University of Washington
30 January 2009

Topics

0. Introduction
1. Possibilities
2. Sample app
3. Under the hood
4. API benefits and costs
5. Best practices
6. Strategic choices

0. Introduction

API: application programming interface

Browse APIs by Category

Advertising (14)	Events (10)	Music (38)
Security (21)	Answers (5)	Fax (2)
News (15)	Shipping (8)	Auctions (2)
Feeds (12)	Office (13)	Shopping (43)
Blog Search (7)	File Sharing (6)	Other (80)
Sports (9)	Blogging (18)	Financial (53)
Other Search (1)	Storage (15)	Bookmarks (15)
Food (3)	Payment (7)	Tagging (8)
Calendar (4)	Games (15)	Photos (38)
Telephony (35)	Catalog (1)	Government (22)
PIM (7)	Tools (9)	Chat (11)
Internet (66)	Politics (3)	Travel (18)
Community (55)	Job Search (8)	Project Management (17)
Utility (11)	Database (8)	Mapping (76)
Real Estate (10)	Video (42)	Dating (1)
Media Management (7)	Recommendations (19)	Weather (4)
Email (27)	Media Search (1)	Reference (57)
Widgets (17)	Enterprise (31)	Medical (13)
Search (36)	Wiki (9)	Events (10)

<http://www.programmableweb.com/apis>

0. Introduction

API: application programming interface

Big Huge Thesaurus

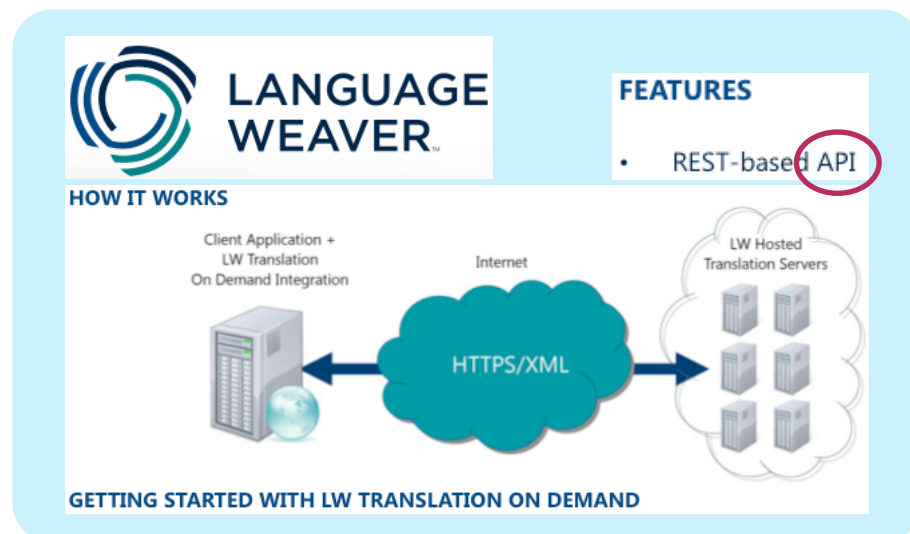
Synonyms, antonyms, and rhymes (oh my!)

API

This site sports a very simple API for retrieving the synonyms for any word.

Abbreviations.com

Our API interfaces let developers design computer programs and web applications that interact directly with the Abbreviations.com content and services.

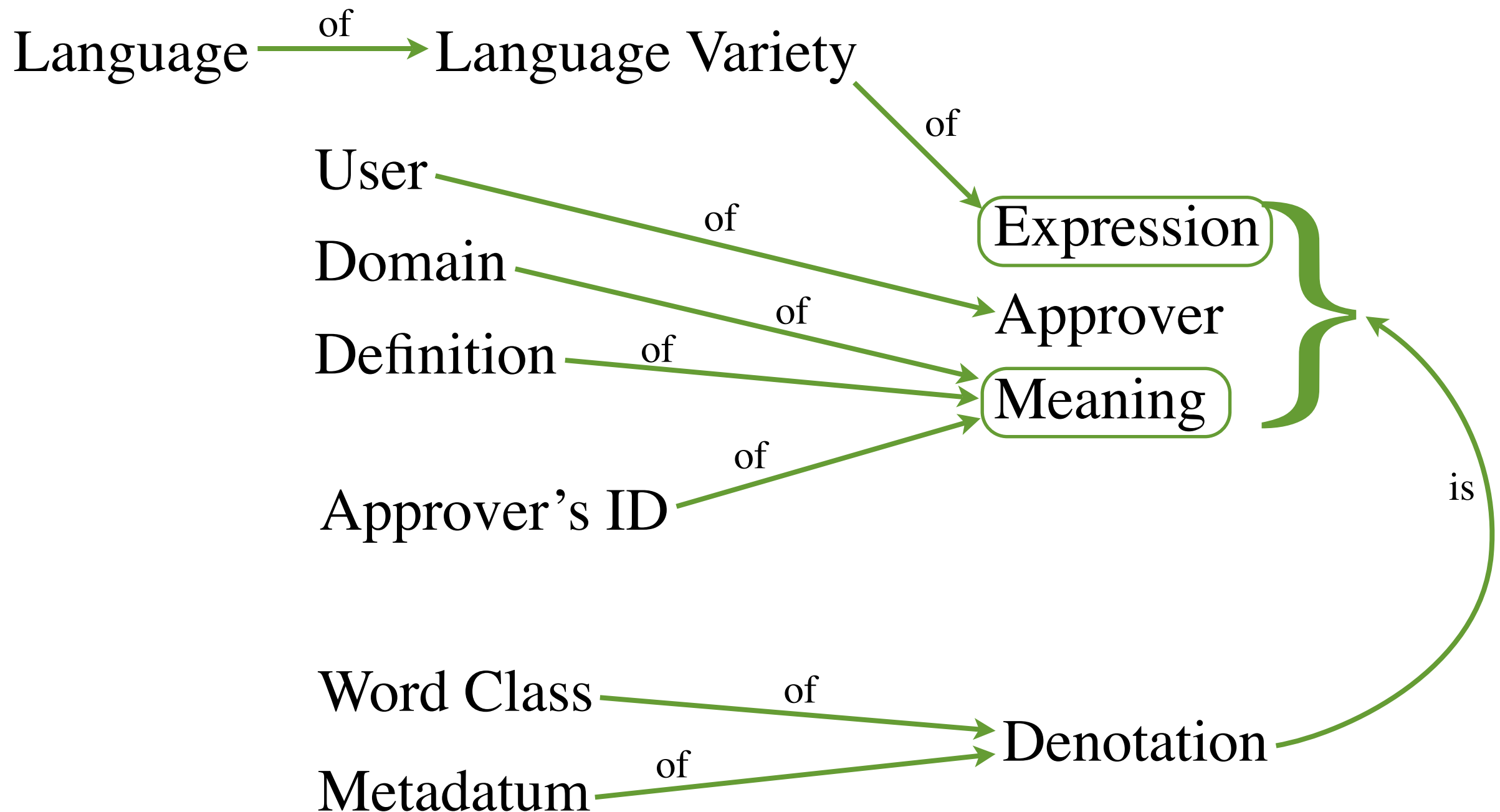


Human Translation Server © Web Service

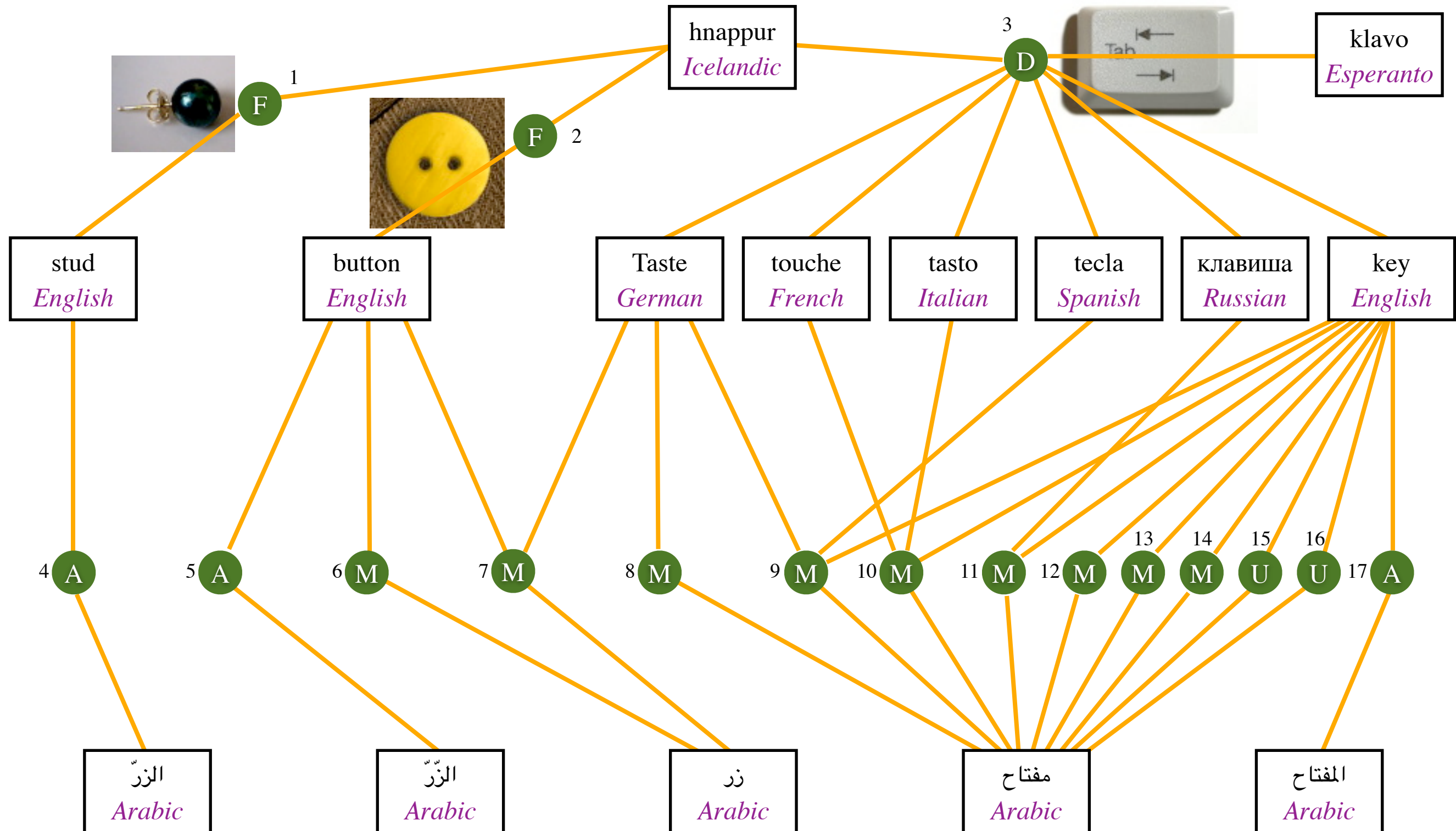
HTS™ is a set of API functions that allows you to integrate human translation into your content management workflow, your desktop environment and your server applications.

0. Introduction

PanLex: a panlingual lexical database



0. Introduction



0. Introduction

Database facts

Started 2006.

Versions:

- PostgreSQL/Linux with no inference (“PanLex”)

- SQL Server/Windows + peer-to-peer inference (“TransGraph”)

- SQL Server/Windows + centralized inference (“PanDictionary”)

Statistics by January 2009:

- 12 million lexemes

- 1,265 language varieties

- 600 lexical resources

1. Possibilities

Games:

Panlingual “Free Rice”

The screenshot displays the 'Free Rice' website interface. At the top, a green navigation bar contains the text 'FREE Rice' and a menu with links: HOME, SUBJECTS, FAQ, TOTALS, OPTIONS, PRESS, CONTACT, and ABOUT. Below the navigation bar, a green banner reads: 'For each answer you get right, we donate 10 grains of rice through the UN World Food Program to help end hunger'. The main content area is divided into two columns. The left column shows the current subject 'Italian' and a 'Change Subjects' link. It displays a correct answer: 'CORRECT! entrare = to enter'. Below this, a question asks: 'l'ufficio means:' followed by four options: 'bag', 'evening', 'office', and 'museum'. The right column features a circular image of a wooden bowl filled with white rice grains. Below the image, it states: 'You have now donated 30 grains of rice.' At the bottom of the interface, a green bar shows the user's progress: 'Level: 4 of 10', 'Best Level: -', 'Change Level', and 'Re-Start'.

1. Possibilities

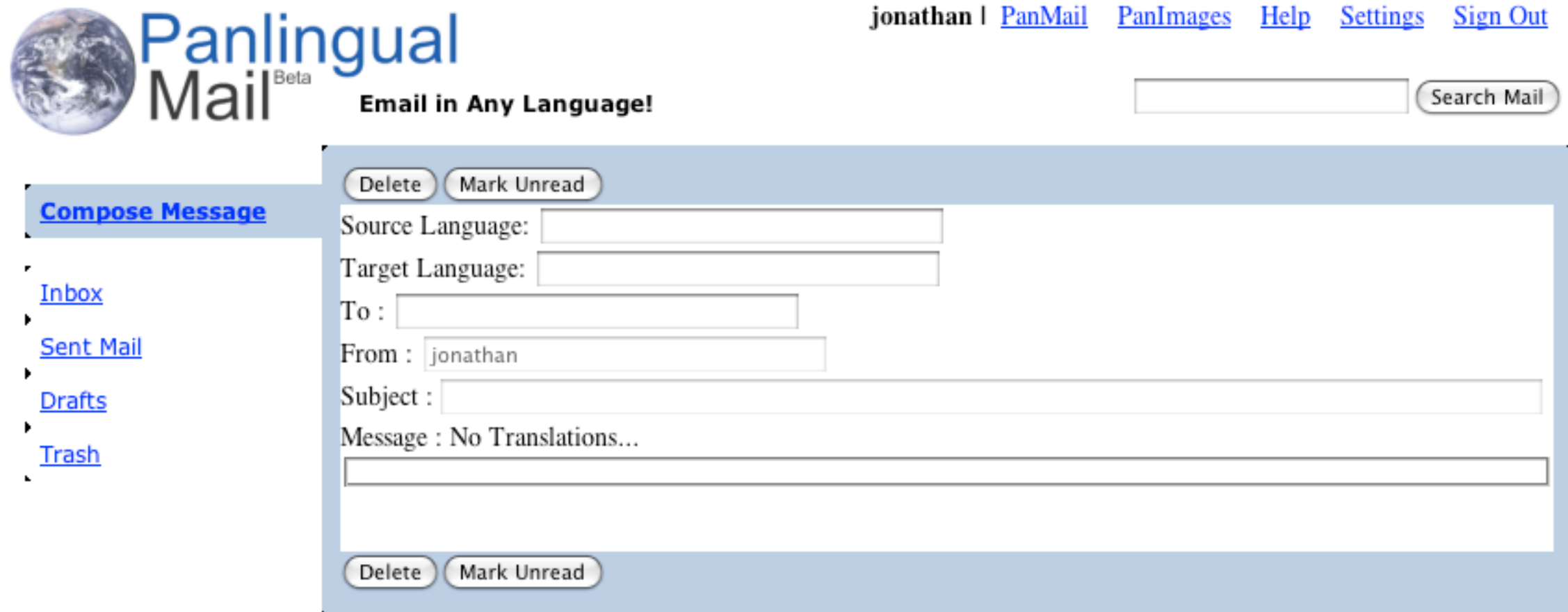
Social networking:

Panlingual personals

The screenshot shows a web interface for a panlingual dating site. At the top, there is a blue header with a globe icon and the Chinese text "让生活充满爱" (Let life be full of love). To the right of the header is the copyright notice "©YourNewLover.Com". Below the header is a navigation bar with language options: Français, Русский, English, Italiano, Deutsch, Español, and 中文. The main content area is divided into two columns. The left column features a red "+ 注册" (Register) button, followed by the text "真正的多语言系统" (True multilingual system) and "让使用不同语言的人更为简单便捷交流的语言翻译系统." (A language translation system that makes it easier and more convenient for people who use different languages to communicate). Below this is a link "详尽的..." (Detailed...). At the bottom of the left column are three blue links: "更多帮助" (More help), "补充服务" (Additional services), and "登录" (Login). The right column features a red "查找用户" (Find users) button. Below it is a search form with the following fields: "交友目的:" (Dating purpose) with a dropdown menu set to "任何的" (Any); "照片:" (Photos) with a dropdown menu set to "必须的" (Required); "语言:" (Language) with a dropdown menu set to "任何的" (Any); "年龄从:" (Age from) with a dropdown menu set to "任何的" (Any); "年龄至:" (Age to) with a dropdown menu set to "任何的" (Any); and a text input field for "... 或指出用户名:" (... or specify username:). A "搜索" (Search) button is located below the search form. Below the search form is a section titled "最近用户" (Recent users). The first user listed is "linmengling, 42岁" (linmengling, 42 years old), with a small profile picture and the text "中国" (China) and "我想结识: 一个男人, 一个不错的人即可" (I want to meet: a man, a decent person is fine).

1. Possibilities

Messaging:

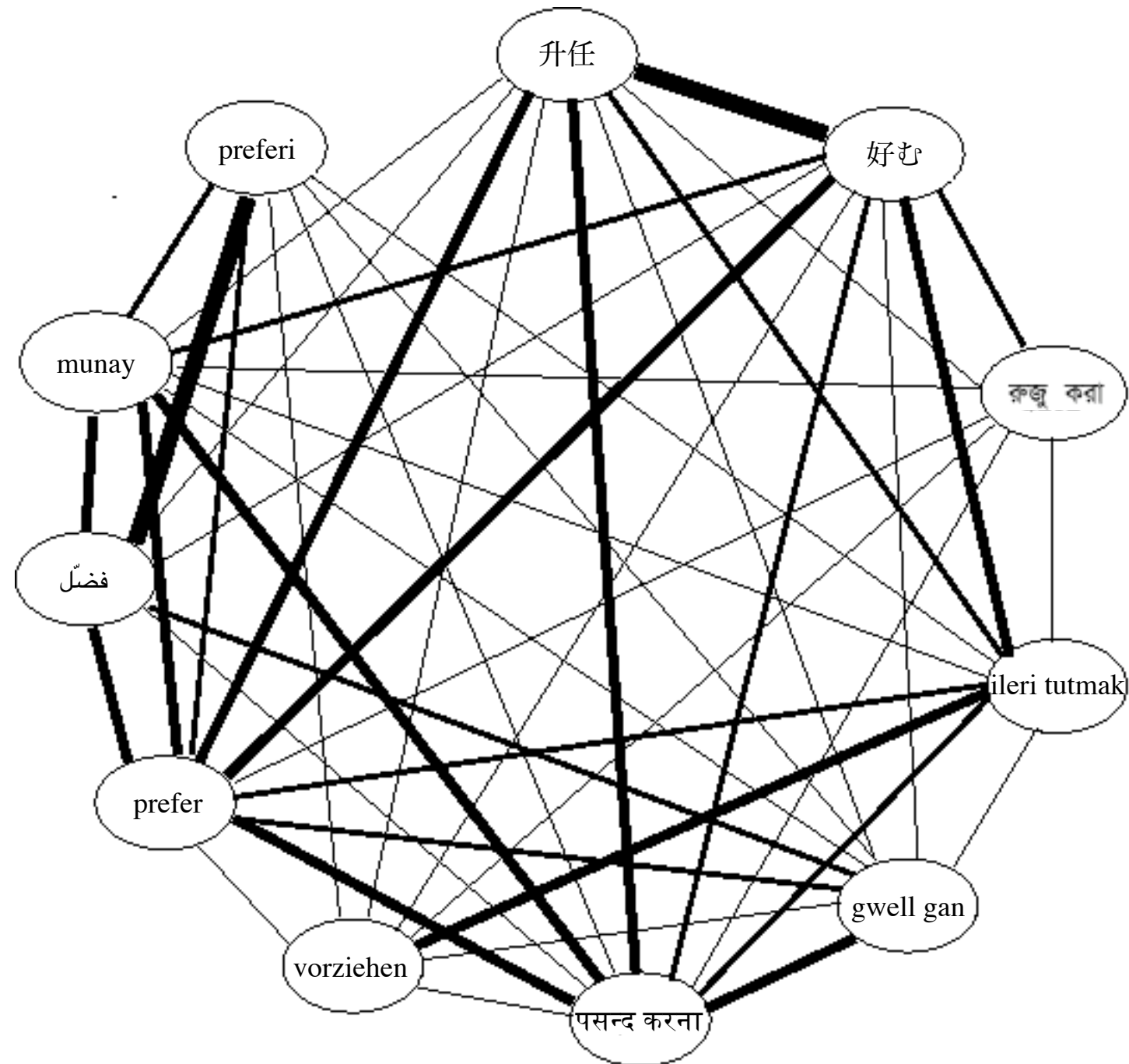


The screenshot displays the Panlingual Mail web interface. At the top left is the logo for Panlingual Mail Beta, featuring a globe and the text "Panlingual Mail^{Beta} Email in Any Language!". To the right of the logo, the user's name "jonathan" is displayed, followed by navigation links: [PanMail](#), [PanImages](#), [Help](#), [Settings](#), and [Sign Out](#). A search bar with the text "Search Mail" is located in the top right corner. On the left side, a vertical navigation menu includes links for [Compose Message](#), [Inbox](#), [Sent Mail](#), [Drafts](#), and [Trash](#). The main content area shows a message composition form with the following fields: "Source Language:" and "Target Language:" (both with empty text boxes), "To:" (with an empty text box), "From:" (with the value "jonathan"), and "Subject:" (with an empty text box). Below these fields, the message content is displayed as "Message : No Translations...". At the top and bottom of the form area are buttons for "Delete" and "Mark Unread".

[Panlingual Mail is provided by the Turing Center](#)

1. Possibilities

Science:



1. Possibilities

Translation:

TümSöz
Çevirici

Burada Türkçe sözleri 1.260 dile çevirebilirsiniz!

Bir söz (ya da söz gibi kısa deyim) buraya yazın:

Çevirelim!



Sample app →

2. Sample app

<http://panlex.org/demo/trtur.html>

Translate Turkish words
into 1260 languages!

Enter a word or phrase:



The screenshot shows the 'TümSöz Çevirici' web application. At the top, the title 'TümSöz Çevirici' is displayed in orange. Below it, a green banner reads 'Burada Türkçe sözleri 1.260 dile çevirebilirsiniz!'. A green-bordered box contains the instruction 'Bir söz (ya da söz gibi kısa deyim) buraya yazın:' followed by a white text input field. Below the input field is a button labeled 'Çevirelim!'. At the bottom, there is a red square with a white crescent and star, two blue globes, and a 'W3C XHTML 1.1' logo with a blue checkmark.

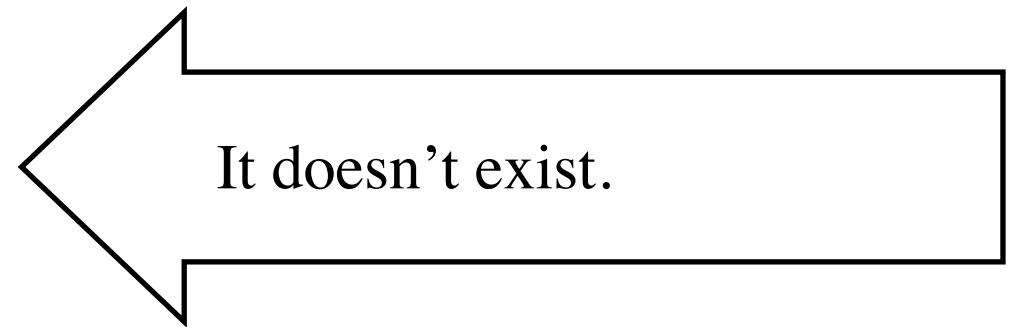
Under the hood →

3. Under the hood

Case 1

Client

Server



3. Under the hood

Case 2

Client

What is the ID of “çünkü” in Turkish?

What language varieties can you translate 1355719 into?

What translations from 1355719 into Ukrainian exist?

Server

It is 1355719.

Afrikaans; ...; עברית; ...;
日本語; ...; isiZulu

адже; бо; внаслідок;
оскільки; так як; тому; ...

3. Under the hood

Protocol: HTTP.

Server port: TCP 80.

Method: POST.

Content-type: XML.

Encoding: UTF-8.

Client → Server

```
POST http://panlex.org/p
```

```
Content-Type: application/xml; charset=utf-8
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<data><vr>00</vr><op>ex-0</op>
```

```
<lv>738</lv><et>çünkü</et></data>
```


3. Under the hood

Protocol: HTTP.

Server port: TCP 80.

Method: POST.

Content-type: XML.

Encoding: UTF-8.

Server → Client

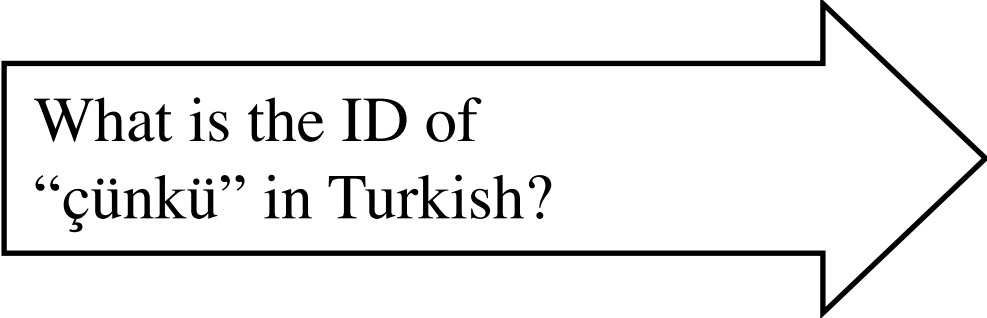
```
HTTP/1.1 200 OK
```

```
Content-Type: application/xml; charset=utf-8
```

```
<?xml version="1.0" encoding="utf-8"?>  
<data><vr>00</vr><op>ex-0</op>  
<lv>738</lv><et>çünkü</et><ex>1355719</ex></data>
```

3. Under the hood

Request syntax



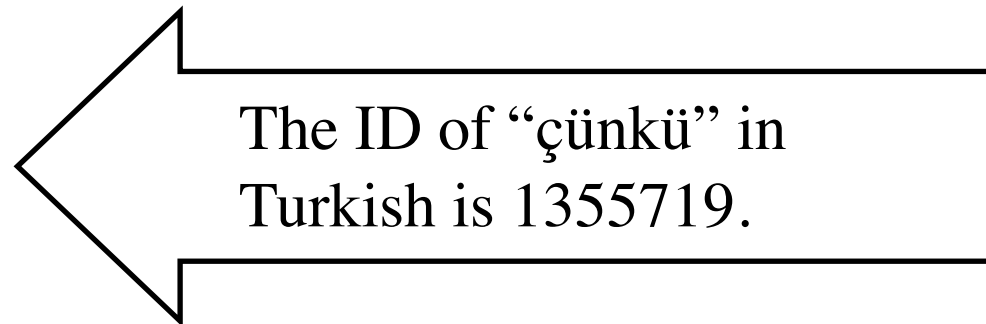
What is the ID of
“çünkü” in Turkish?

```
<data>  
  <vr>00</vr>  
  <op>ex-0</op>  
  <lv>738</lv>  
  <et>çünkü</et>  
</data>
```

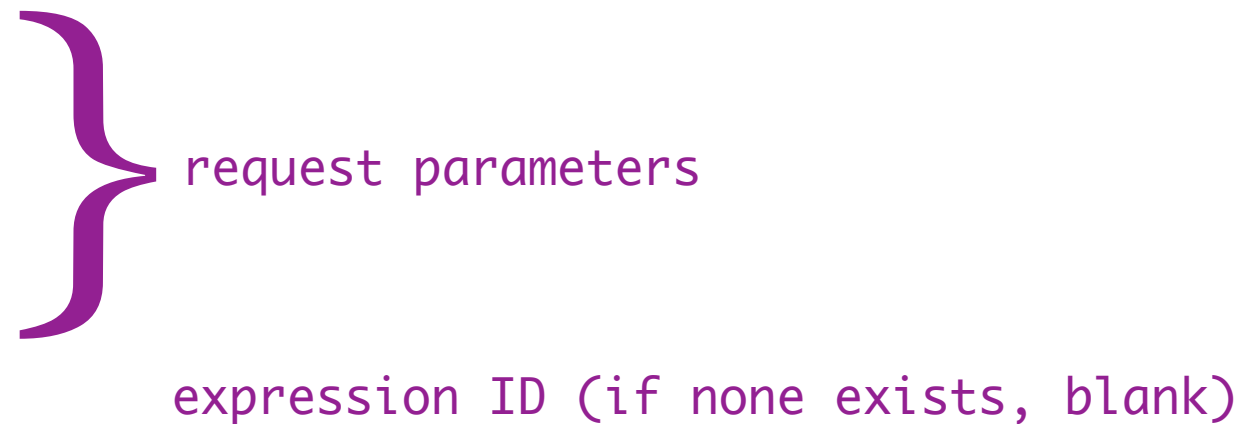
API version
operation
language variety
expression text

3. Under the hood

Response syntax



```
<data>  
  <vr>00</vr>  
  <op>ex-0</op>  
  <lv>738</lv>  
  <et>çünkü</et>  
  <ex>1355719</ex>  
</data>
```



3. Under the hood

Error messages

<data>

<err>In 2: Request content not UTF-8: 0fffd:000ef</err>

</data>

<data>

<err>ex-0-DoArgs 3: No arg lv</err>

</data>

4. API benefits and costs

Benefits

Total utilization of the data.

Diversity of applications using the data.

Freedom from the obligation to do everything.

Recruitment of partners.

Revenue based on the total value of the resource (e.g., grants).

Visibility of the project.

Costs

Control of look and feel.

Control of uses of the data.

Risk of claims of infringement on intellectual-property rights.

Revenue from human visitors to one's own service (e.g., advertising).

Freedom from the obligation to continue uninterrupted service provision.

5. Best practices

API pundits* say:

- Design APIs for simple, sequential request/response **turns**.
- Make rules **easy** and provide **templates**: Most developers are amateur or lazy.
- Offer **multiple** styles/formats/languages: Let each developer use familiar tools.
- Start **basic**: Let developers tell you what else they want.
- Don't change a **version**: Never make developers change working apps.
- Write **multiple demos**: Force versatility on your API.
- **Manage** use: Human limits on traffic and abuse don't apply.
- Follow general **programming** best practices: Document, modularize, hide, etc.

*Examples of API punditry:

Joshua Bloch (<http://lcsd05.cs.tamu.edu/slides/keynote.pdf>)

Nemetril (<http://nemetril.net/2008/06/10/the-pursuit-of-apiness-part-1/>)

Mashery (<http://blog.mashery.com>)

Alex Barnett (http://alexbarrett.net/blog/archive/2006/10/28/Web-API-Design-_2D00_-Keep-Some-of-it-Simple_2C00_-Stupid.aspx)

6. Strategic choices

- Protocol/style: DICT vs. REST vs. SOAP vs. XML-RPC vs. SQL.
- Server port: TCP 80 (HTTP) vs. TCP 2628 (DICT).
- Message medium: URI (GET) vs. standard input (POST).
- Encoding: UTF-8 re-encoding: no (multipart) vs. XML vs. URL.
- Content language: XML vs. JSON vs. token-sequence (DICT).
- Statefulness: yes (DICT) vs. no vs. partly (expiring details).
- Customization: stored settings.
- Utilization management: client keys, throttling, premium services.
- Limitations on resource-intensivity.
- Client hosting.
- Service elaboration.
- Aggregation of common service sequences.
- SQL as a service.
- Performance enhancement: caching, derivative tables, server clustering, EC2.
- Organization for durability.

6. Strategic choices

Popular protocols/styles

Dictionaries: DICT (RFC 2229*). Example:

```
C: DEFINE * shortcake

S: 150 2 definitions found: list follows
S: 151 "shortcake" wn "WordNet 1.5" : text follows
S: shortcake
S:   1. n: very short biscuit spread with sweetened fruit and usu.
S:     whipped cream
S: .
S: 151 "Shortcake" web1913 "Webster's Dictionary (1913)" : text follows
S: Shortcake
S:   \Short"cake`\, n.
S:   An unsweetened breakfast cake shortened with butter or lard,
S:   rolled thin, and baked.
S: .
S: 250 Command complete
```

*<http://www.faqs.org/rfcs/rfc2229.html>

6. Strategic choices

Popular protocols/styles

Everything else:

ProgrammableWeb* classifies 1,107 APIs by:

Protocol or style: 64% REST, 22% SOAP, 7% JavaScript.

Data format: 68% XML, 19% JSON, 7% RSS.

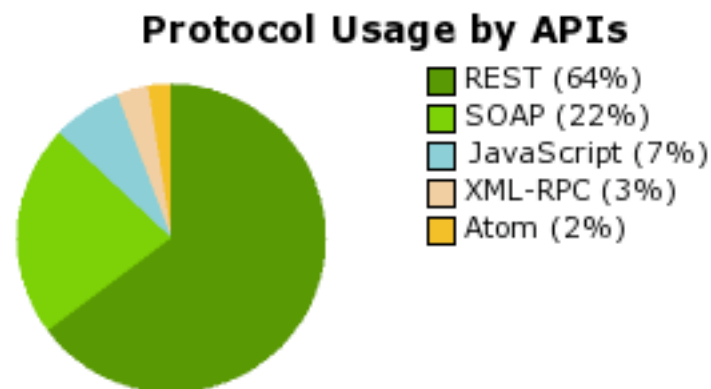
Of these, 57 are “reference” APIs. They are:

Protocol or style: 54% REST, 46% SOAP.

Data format: 81% XML, 18% JSON.

*<http://www.programmableweb.com/apis/directory>

API Protocols



6. Strategic choices

Popular message media

Standard input (less popular):

```
<data>  
  <vr>00</vr>  
  <op>ex-0</op>  
  <lv>764</lv>  
  <et>bưƠm bưỚm</et>  
</data>
```

```
<data>  
  <vr>00</vr>  
  <op>ex-0</op>  
  <lv>764</lv>  
  <et>bÆ°Æj m bÆ°á»>m</et>  
</data>
```

UTF-8

URI (more popular):

```
vr=00&op=ex-0&lv=764&et=bưƠm bưỚm
```

```
vr=00&op=ex-0&lv=764&et=b%c6%b0%c6&a1m b%c6%b0%e1%bb%9bm
```

URL-encoded UTF-8