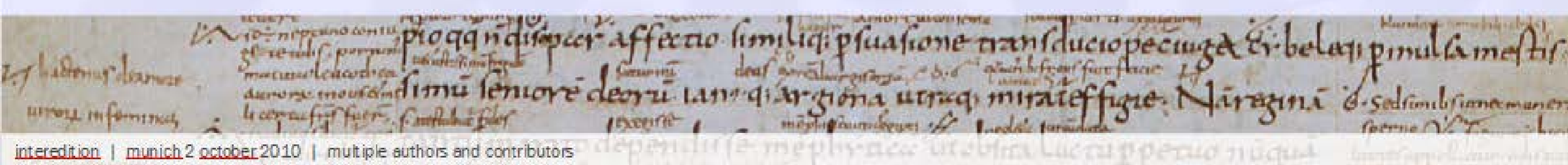


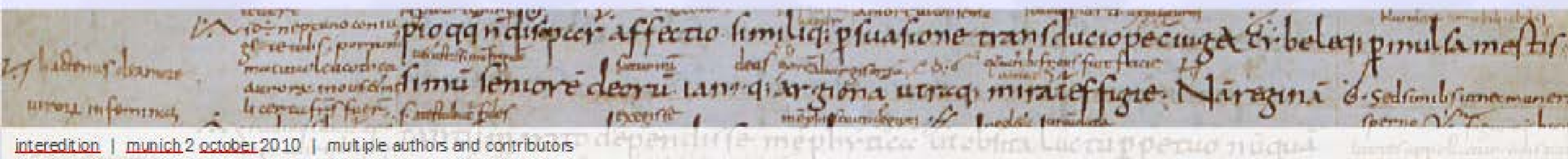
# Putting the pieces together

Interedition microservices in practice



# Why microservices?

- Every scholar wants something different
- Small tools are easier to maintain
- Small tools are useful to more people
- ...if they can be combined with other tools.



## Tokenizer

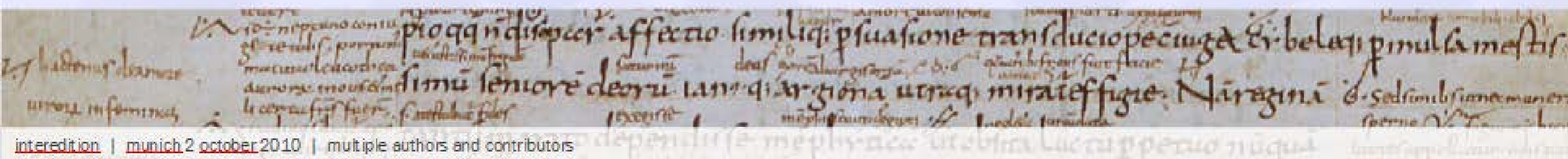
**Web service mode:**  
 Send a POST request to this service at /regularize\_fuzzy, the request body of which should contain a JSON formatted CollateX token sets adhering to <http://gregor.middell.net/collatex/api/collate>. It'll return your set containing normalized tokens based on fuzzy regularization in the vocabulary of all the text contained, otherwise your JSON will be untouched.

**Interactive mode:**  
 Not implemented yet

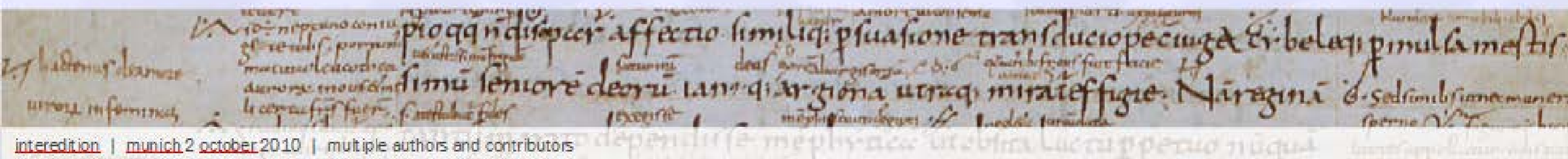
## Fuzzy matcher

## Collator

## Statistics package



# How not to do it



**Interactive Tokenizer Input**

Navigation: XML text to tokenize  
Enter some XML text here

Powered by Catalyst

Tokenizer

**Web service mode:**  
Send a POST request to this service at /regularize\_fuzzy, the request body of which should contain a JSON formatted CollateX token sets adhering to http://gregor.middell.net/collatex/api/collate . It'll return your set containing normalized tokens based on fuzzy regularization in the vocabulary of all the text contained, otherwise your JSON will be untouched.

**Interactive mode:**  
Not implemented yet

Fuzzy matcher

save the data,  
load the data

save the data,  
load the data

**REST service**

This is the REST service of CollateX. To call it, you can post witness data as specified below and get the collation result back in a number of formats.

**Input**

The service is callable via [HTTP POST requests](#). It expects witness data formatted in [JavaScript Object Notation \(JSON\)](#) as the request body; accordingly the content type of the request must be set to application/json.

A typical request's content might look as follows:

```
Content-Type: application/json; charset=UTF-8
{
  "witnesses": [
    { "id": "A", "content": "A black cat in a black basket" },
    { "id": "B", "content": "A black cat in a black basket" },
    { "id": "C", "content": "A striped cat in a black basket" },
    { "id": "D", "content": "A striped cat in a white basket" }
  ]
}
```

The root object in the body must contain a property `witnesses`, which contains a list of witness objects in turn. Each witness object has to contain a unique identifier in the property `id`. Besides the object can either contain the textual content of the witness as a string property named `content` (as shown above). The other option is a pre-tokenized witness, that is comprised of list of tokens notated as follows:

```
{
  "witnesses": [
    { "id": "A", "tokens": [
      { "t": "A" },
      { "t": "black" },
      { "t": "black" }
    ]
    }
  ]
}
```

Collator

**Mobyle@pasteur**

Welcome Programs Data Bookmarks Jobs Tutorials

Phylip pars

**Discrete character parsimony**

Input File (PhylipDiscreteCharMatrix)

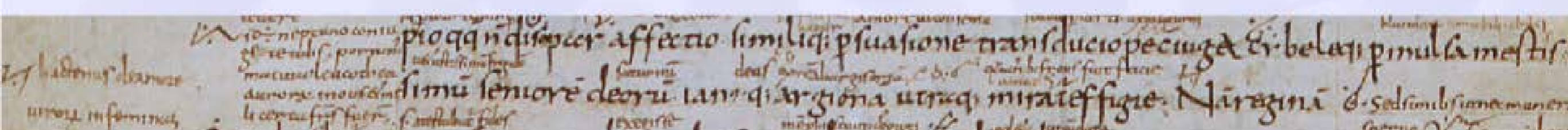
Paste File clear data

Input File Format

Sequential Interleaved

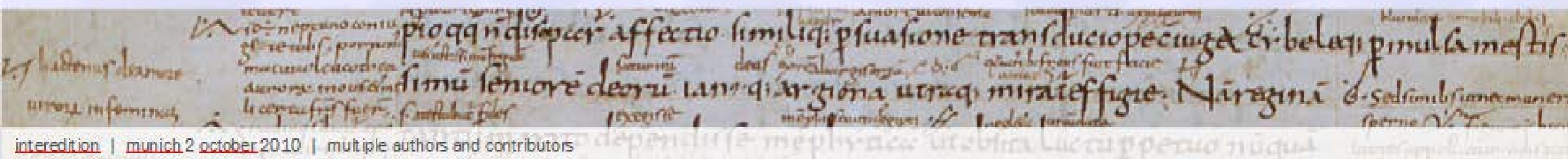
Statistics package

save the data,  
load the data

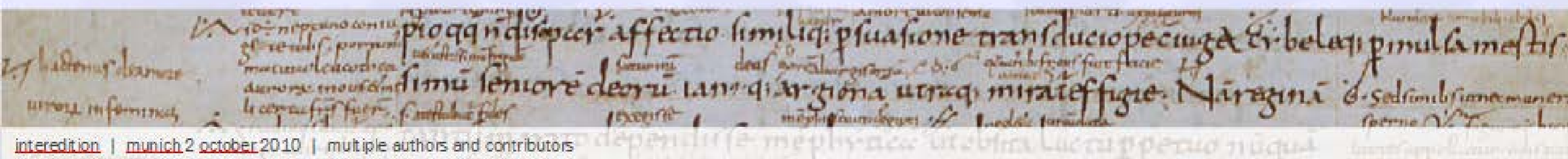




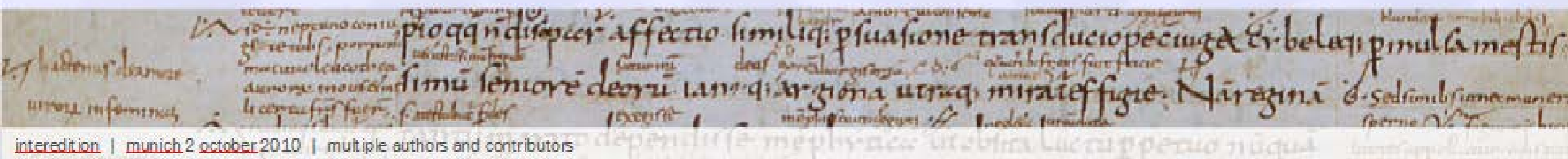
...lost in intermediate  
data files yet?



- Good microservices can be called by other applications.
- Make your own application
- However you want
- Without duplicating the work of others



# An application for text collation





This a CollateX based text collation client to show the strength of web services based pipeline workflow as an interoperability mechanism between digital and computational textual scholarship tools. This client is using the CollateX collation engine (running on a different server than this client). CollateX is a powerful fully automatic baseless text collation engine for multiple witnesses.

## 1 Files

### Select files

Select the files on your local drive, or provide one or more URLs, that contain the texts that you would like to collate.

 [Remove](#)

[Add Another](#)

Or type the URL to the online text:

 X

[Add Another](#)

## 2 Configure

Fuzzy Matching (e.g. for medieval texts)

### Collator

- ▶ Collatex
- ▶ Other [not available]

### Results as:

- ▶ TEI
- ▶ JSON
- ▶ HTML

## 3 Result

### Collated Result

