Google AWARD

PaRADIIT : Pattern Redundancy Analysis for Document Image Indexation and Transcription

Laboratoire Informatique (EA 2101)
Université Francois-Rabelais de Tours - France

UNIVERSITÉ FRANÇOIS - RABELAIS TOURS

LI Laboratoire d'Informatique EA 2101

POLYTECH TOURS

Overview

- Introduction
  - Context
  - Challenges
- PaRADIIT proposition
  - Learning from the user feedback and from pattern redundancy
  - User driven systems: Agora + Clustering + Retro
- Interest of Pattern redundancy
  - Description
  - For text recognition (OCR)
  - For typography analysis
  - For word spotting
- State of the project
  - Organisation
  - WP1 : Agora
  - WP3 : Clustering
  - WP2 : Retro
- Future works : Planning

Introduction
Context of the work

- Collaboration with the CESR of Tours
  - The Humanistic Virtual Library (BVH in French)
  - A research center and library with rare books (Loire Valley)
  - From the Renaissance period (14th - 16th)
- A pluri-disciplinary collaboration
  Experts in DIA + Experts in old books + End-users
- Objectives: Deal with and manage specificity of old books
  - Fully automatic is impossible because of **variability**
  - Adaptation according to image contents (typography)
    ➔ not before but during the processing
  - Introduce more interaction into DIA systems
    ➔ user-driven method

Introduction
Challenges

- **Overcoming commercial OCR performances**
  - Segmentation in lines, words, characters is a problem

Books of the CESR	Omnipage classical segmentation	Omnipage with Ocropus segmentation
Recueil des antiquités Gauloises	89.82%	85.93%
Histoire de l'expédition chrestienne au royaume de Chine	86.48%	61.25%
Les treselegantes et copieuses annales	85.6%	73.92%
Les histoires de Diodore sicilien	90.19%	83.82%

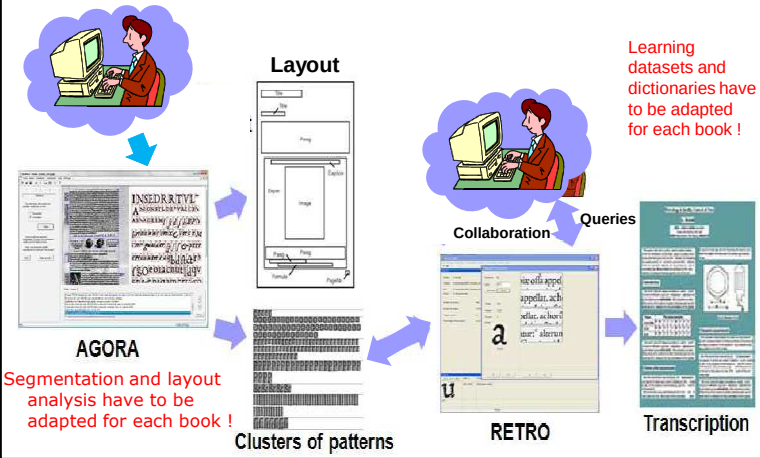
  - Learning datasets (typography) and prior knowledge (dictionaries and linguistic aspects) are very important

Font	poly-font system	Adapted poly-font system	mono-font system
Average (30 fonts)	86.59	96.02	99.55
Berkeley Old – Berkeley Oldstyle	96.62	97.2	98.98
Banco – Banco	34.08	73.46	98.77
Mistral – Mistral	46.63	92.16	94.89
Fette Kanzlei – Fette Kanzlei	68.36	95.74	99.43

Introduction
Our proposal

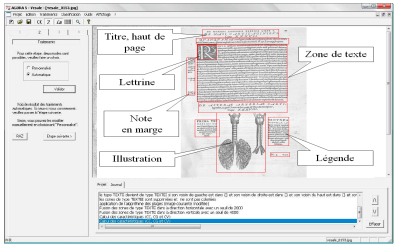
- **More interaction in DIA systems**
  - For adaptation according to each "book" specificities
  - For integration of the user needs
- **Incremental analysis of images**
  - Segmentation for recognition, recognition for segmentation
  - Solution: From the simplest to the more difficult
- **Desired Results ➔ a software suite**
  - An open-source forge for RETRO and AGORA with
  - An improved clustering method (pattern redundancy analysis)
  - An interactive and collaborative transcription system.
  - New functionalities concerning typographical studies: creation of typographical families to generate learning datasets

PaRADIIT Architecture
Learning from the users and from the data



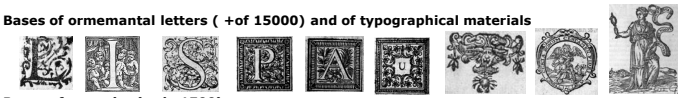
PaRADIIT Architecture
User-driven analysis with AGORA

- **User-driven analysis**
- Extraction of specific elements of contents (dropcaps, …)
- Generate XML files describing the structure (similar to Alto) Lines, words and CC positions
- Used since 2004 (CESR)



Download :
http://www.rfai.li.univ-tours.fr/pagesperso/ramel/fr/work1.html

- **Bases of ormemantal letters (+of 15000) and of typographical materials**

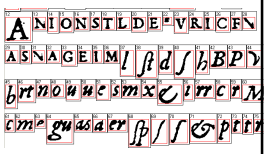


- **Bases of portraits (+ de 1500)**

Voir sur http://www.bvh.univ-tours.fr

Pattern Redundancy in text
Description

- Goal: Analyzing redundancy in images (text part for us)
  - A text, ancient or not, is made up of sequences of similar patterns
- Methods: Clustering of similar patterns to create groups (classes)
  - Comparison of patterns (matching techniques)
  - Without prior knowledge about the meaning of these patterns
- Constraints are that the techniques should:
  - Produce very homogeneous clusters ➔ Different patterns may not be blended into one cluster
  - Produce a minimal number of clusters
- What could be a pattern?
  - Connected components [Lebourgeois95]
  - Words [Kluzner&Al2009]
  - Others [Roy&Al2011]
  - Redundancy rate > 80 %



- Used first for compression in Debora project and DjVu Format

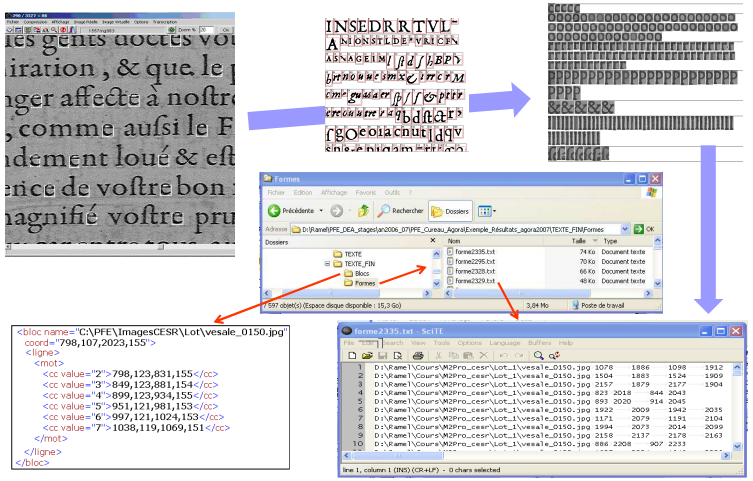
Pattern Redundancy in text
Description

- **Using connected components as patterns**
- The first and simplest way to realize such analysis
- Redundancy rate starts around 75% when using a single page
- Redundancy can reach up to 95% when processing an entire book (modern)
- This rate depends largely on the quality of printing

Number of pages	1	2	3	4	5	6	7	8
Total # of clusters of binary patterns	555	915	1,209	1,485	1,678	1,870	2,083	2,262
Total # of characters	2,327	4,245	6,681	8,681	11,159	13,589	16,141	18,028
Redundancy rate	76%	78%	81%	82%	84%	86%	87%	88%

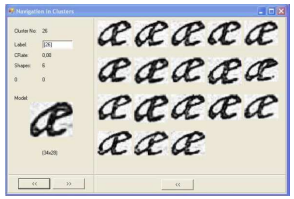
- Redundancy rates slightly upwards of 80% when documents present high typographical variabilities of character style, size, and font.

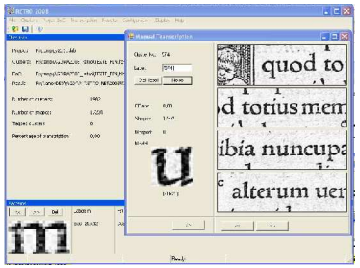
Using connected components
AGORA ouputs

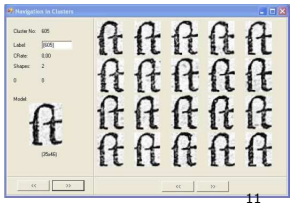


Using connected components
For Transcription

- **RETRO GUI – Computer Assisted Transcription (manual)**
  - For tagging the clusters using unicode
  - Cluster visualization
  - Characters (CCs) in context
  - Creation (selection) of new templates

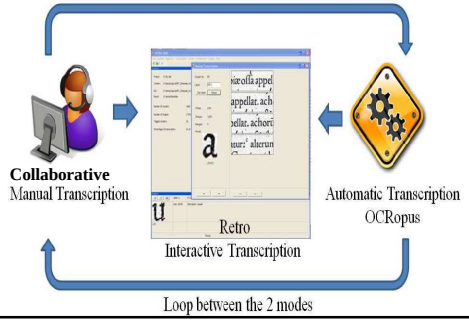






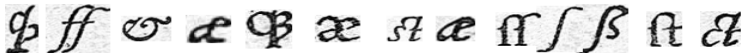
Using connected components
For Transcription

- **Inside a loop !**
  - Cooperation between manual (collaborative users), automatic (OCR) and contextual (dictionaries) contributions
  - *Adaptive-system : From simplest to the more complicated*



Using connected components
For Typographic analysis

- **Create a classification of the Early Modern fonts**
  - Relationship between words containing characters which have the same shape → typographical family and character style
  - Sorted newly created families to find the main typography class as well as minor typographies used for a precise logical meaning
  - Very small typographical families represent words which seldom occur in the text (text in graphics, titles, authors' names, etc.)
- **Study of aesthetic aspects of printing**
  - The thickness and the shape of printing types evolved greatly from the 15th to the mid-16th century
  - Extract and create new font packages from specific printing material (e.g. rare books printed with particular plug sets).

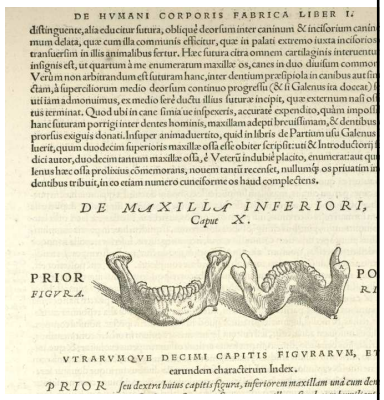


Using connected components
For Typographic analysis

- **Improving the OCR learning step (templates)**
- Dataset production
  - Based on the previous proposition ➔ typographic analysis
  - Produced fonts + model of distortion and degradation = adapted training sets
- Dynamic template selection (incremental learning)
  - Identification of specific fonts used inside the images
  - Automatic selection of specific OCR dedicated to that font (mono-font OCR)
  - Increase the potential performances of OCR engines
  - *Adaptive-systems able to learn from the data*

Using connected components
Experiments

- **Vésale – 1543**
- 150 pages in Latin
- 1.062.081 connected components
- Around 40.000 clusters
- The 200 largest classes correspond to 85% of the text
- 57% of the classes are composed of a single shape
- 90% of the classes are composed of less than 10 occurrences
- Ignoring these classes during transcription means to miss one character for 14 → more than one on each text line !!!


DE HVMANI CORPORIS FABRICA LIBER I.

DE MAXILLA INFERIORI, Caput X.

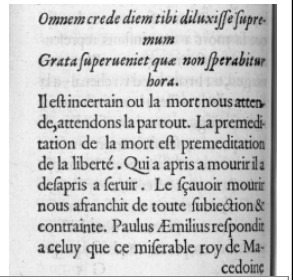
PRIOR FIGVRA.

VTRARVMQVE DECIMI CAPITIS FIGVRARVM, ET earundem characterum Index.

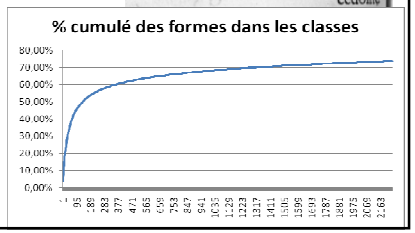
PRIOR seu dextra huius capitis figura, inferiorem maxillam unà cum den

Using connected components
Experiments

- ***Montaigne - 1557***
- 119 pages – 3260 text blocks
- 125 744 connected components (pseudo characters)
- 29 943 clusters
- 25 classes = 25% of the text
- 136 classes = 50%
- 4 000 classes = 75%
- 20 000 classes = 90%
- 79% of the classes are composed of a single shape
- The biggest class = 3%
- 1,2% of the shapes are put in the wrong cluster

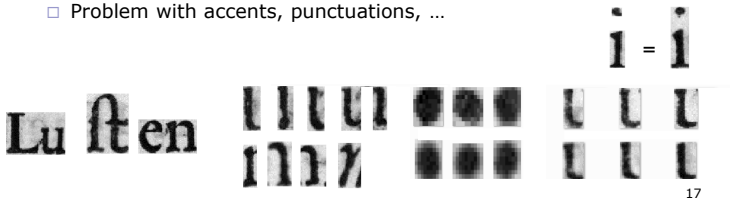

Omnem crede diem tibi diluxisse supremum
Grata superueniet quæ non sperabitur hora.

Il est incertain ou la mort nous attende, attendons la par tout. La premeditation de la mort est premeditation de la liberté. Qui a apris a mourir il a desapris a seruir. Le sçauoir mourir nous afranchit de toute subiection & contrainte. Paulus Æmilius respondit a celuy que ce miserable roy de Macedoine



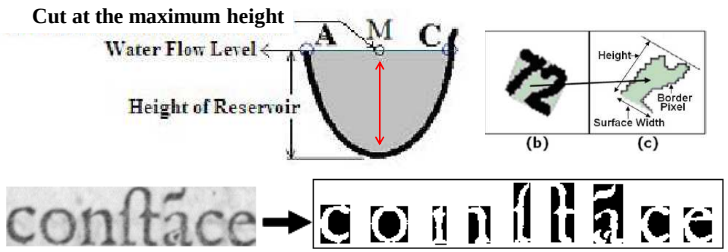
Using connected components (CC)
Discussions

- **Good points with CC redundancy**
  - CCs should correspond to characters
  - System can learn from the images and adapts itself to the used typographic materials
  - Cluster transcription or recognition instead of individual pattern recognition (collaborative, manual, contextual, automatic, …)
- **The segmentation problem is still remaining…**
  - Still require a segmentation step for characters, words and lines
  - Problem with touching and broken characters (CCs)
  - Problem with accents, punctuations, …



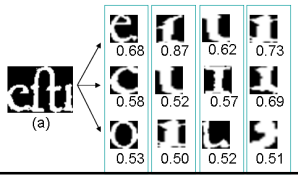
Using glyphs [Roy&Al2011]
For Word spotting

- **Overcoming of the segmentation problem**
  - Glyphs (parts of the connected components) instead of CCs
  - Don't need a segmentation in words and in characters
  - Water reservoir method to split a CC into glyphs



Using glyphs
For word spotting

- **Clustering of glyphs**
  - Similarity measure between glyphs
    - For comparison of 2 glyph images
    - Clustering of glyphs ➔ Codebook of glyphs
  - Text blocks are encoded
    - Each text line is indexed by a string of codebook indexes
    - Glyph clasisication with the same similarity measure
- **Word retrieval**
  - Glyph extraction in the query image and classification
  - Finds all substrings of the *Query* in encoded text blocks

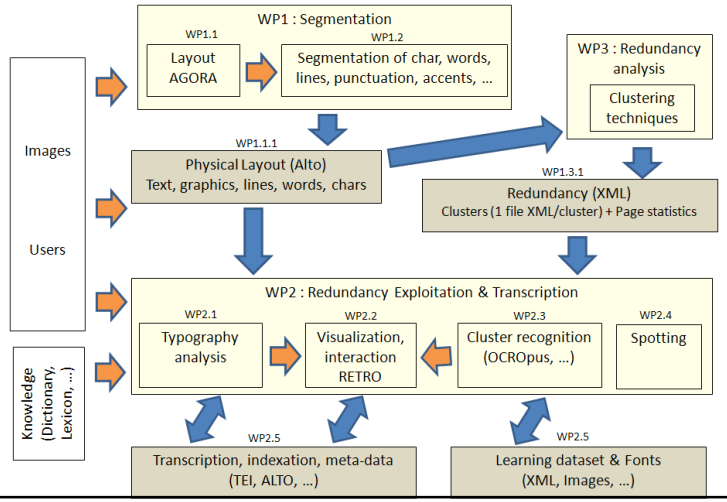


State of the Project
PaRADIIT Project Organisation

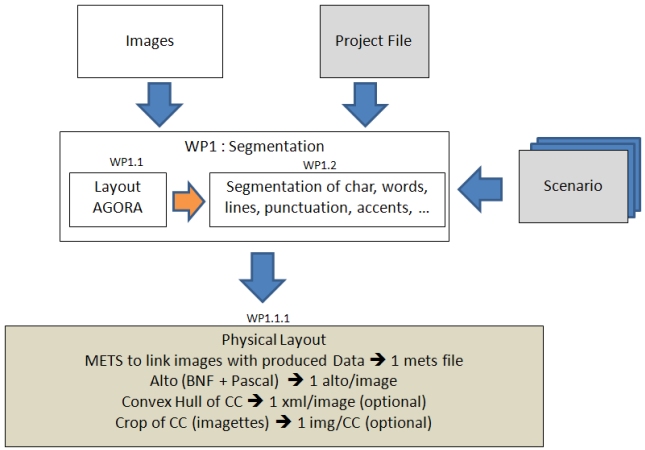
A Completer FRED

- **The PaRADIIT team**
  - Frederic, Pascal, …
- **The PaRADIIT environment**
  - PaRADIIT Web site
    - 3 libraries : Management of ALTO 2.0 / METS 1.0 / TEI Renaissance
  - SVN, Google code…
    - A validation and output generation tool
- **The PaRADIIT events**
  - PaRADIIT in the press
  - Frederic arrival
  - Impact Rouen
  - DH 2011 Stanford

PaRADIIT Project Workplan

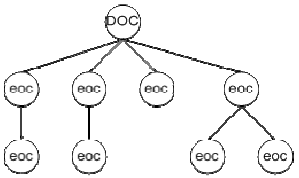


State of the Project
WP1 : Segmentation - Agora



State of the Project
WP1 : Segmentation - Agora

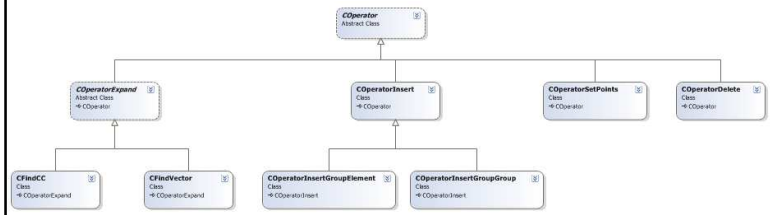
- **Works done since January 2011**
  - Analysis of existing tools, libraries and source code
    - Agora 2008 (C++)
    - Selection of the used Aforge.NET, C#
  - Specification of AGORA2011
    - Input specification
    - Output specification
    - Internal Structure



  - Developments of the libraries
    - 3 libraries : Management of ALTO 2.0 / METS 1.0 / TEI Renaissance
    - A validation and output generation tool

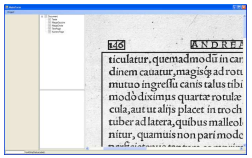
State of the Project
WP1 : Segmentation - Agora

- **Works done since January 2011**
  - Development of AGORA2011 Engine
    - Around 30 classes in C# (≈ 6000 lines)
    - Graph management
    - Operators
    - Unit Tests of these classes

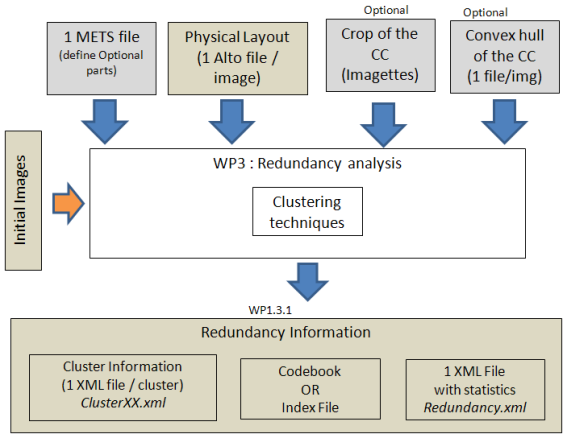


State of the Project
WP1 : Segmentation - Agora

- **Works to do**
  - Development of the GUI
  -
  - Development of document image processing operators
    - incremental analysis and classification of the images contents
    - Considered to Expert level operators
  - Reflexion about high level scenario integration
    - EoC extraction
    - Considered to user level operators
  - Software test and validation



State of the Project
WP3 : Redundancy Analysis - Clustering



State of the Project
WP3 : Redundancy Analysis - Clustering

- **Works done since January 2011**
  - Study of existing techniques and tools
    - Extraction of the patterns
    - Features selection and computation
    - Pattern comparison (distance selection)
    - Clustering
  - Specification of the pattern extraction algorithm
    - From Bounding box to Convex hull

ALTO

```
<String ID="00004.167.1.0.2"
        HEIGHT="33"
        WIDTH="21"
        HPOS="702"
        VPOS="850" />
```

(850, 702)

CCDESC

```
<CC ID="00004.167.1.0.2">
  <POINT RANK=" " X=" " Y=" "/>
  ...
  <POINT RANK=" " X=" " Y=" "/>
</CC>
```

1st: (X, Y)

last: (X, Y)

These pixels must not be taken into account when computing features

State of the Project
WP3 : Redundancy Analysis - Clustering

- **Works done since January 2011**
  - Specification of the features extraction algorithm
    - Combination of Hue's and Zernike's moments
    - Computation time analysis
  - Specification of the Pattern comparison algorithm
    - Distance selection: L-norm, Cosinus, Jacquard index, ...
    - Experiments to select the best one
    - Computation time analysis ➔ we need to merge the patterns in sets before feature comparison
  - Specification of the clustering algorithm
    - Computation of the prototype for a set of patterns
    - Method to cluster prototypes (and then contained patterns)
    - Computation time analysis

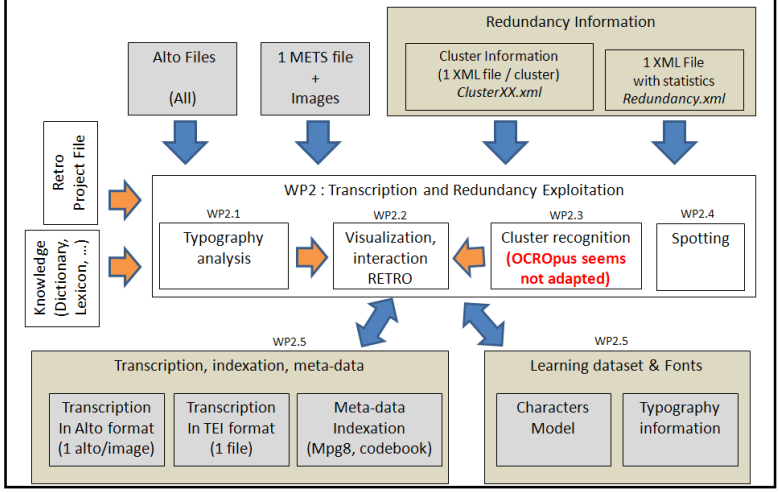
State of the Project

WP3 : Redundancy Analysis - Clustering

- **Works to do**
  - Development of the features extraction algorithm
    - Parallelization of the computing (with parallel techniques such MapReduce)
  - Development of the Pattern comparison algorithm
    - Clever definition of a neighborhood to create sets of patterns
    - Parallelization of the algorithm
  - Development of the algorithms for clustering
    - For the prototype (of a set of patterns) computation
    - To cluster the prototypes (and the patterns)

State of the Project

WP2 : Exploitation - Retro



State of the Project

WP3 : Exploitation- Retro

A Completer FRED

- **Works done and to do**
  - Development of user level operators for document image processing (EoC classification and extraction)
  - Reflexion about high level scenarion integration
    - Operators
    - Unit Tests of these classes

State of the Project

Future Works Planning

- **PaRADIIT project has really started in April 2011**
- **For the end of 2011**
  - September ➔ AGORA2011 Beta
  - October ➔ RETRO2011 Beta
  - December ➔ RETRO2011 Beta
- **For April 2012**
  - AGORA2011 final
  - RETRO2011 final
  - Clustering2011 final
- **We hope the main goals will be reached in April 2012**

Questions ?