BeETLe project:
ETL geo-spatial tool

Juan Arévalo, César Martinez, Walter Simonazzi

Barcelona,  September 9th, 2010
1. Project context
   a. Introduction to ETC-LUSI
   b. Work environment
   c. Processing needs
2. Use case: Current methodology for LEAC project. Problems
3. Solution: BeETLe project
4. Project goals
   a. Unify technologies
   b. Ability to process big data
   c. Standardization and document data work-flows
   d. Parallel execution
5. Roadmap
6. (Possible) future work directions
ETC-LUSI

- European Topic Centre on Land Use and Spatial Information (Universidad Autónoma de Barcelona): http://etc-lusi.eionet.europa.eu/

- European Consortium to support the European Environmental Agency (EEA)

- Main work field: Monitoring of land use and land use changes, and their environmental consequences

- Other thematics related with spatial information: coasts, ecosystem accounting...
ETC-LUSI

• Manages a lot of information at European scale
  → Data has big size
  → Data Types: vector, raster and non-geo

• Data is updated periodically
  → Repetitive work-flows

• Several projects at European scale: FP-7, Espon,
• Other projects at national and regional scale
Use case: LEAC project
Current methodology

- Several tools and programming languages
- Mainly interactive processes
Use case: LEAC project

Current methodology problems

• Several tools:
  → Experienced users
  → License costs
• Format conversions → Processing time
• Interactive processes → User time
• Work-flows hard to to standardise
  → human error
• Work-flows hard to document

Limitations or errors in software: “in the next version or next service pack”
Solution: BeETLe project

- ETL geo-espatial tool
- Based on (Geo-)Kettle and Sextante (+Grass?)
Solution: BeETLe project

• Other solutions were analysed: Talend
• Decision was taken based on:
  • Maturity of the project
  • Community
  • Leader organization supporting the project (Pentaho, Spatialytics, University of Laval).
• Future plans
ETL (Extract, Transform, Load)

- Tools to define work-flows to automate tasks:
  - The model documents the work-flow in a formal way
  - Parallel process execution
Kettle (Pentaho Data Integration):
• ETL open source tool (LGPL)
• Part of the BI suite designed by Pentaho

GeoKettle
ETL for Geospatial Data:
- Kettle extension with spatial support
- Limited support to vector operations (there is no raster support)
- Developed by the GeoSOA research group at University of Laval, Canada.
Kettle

- Easy and intuitive interface
- Parallel and distributed execution
- High number of data sources and transformations available
What does BeETLe bring to GeoKettle?

<table>
<thead>
<tr>
<th>Feature</th>
<th>GeoKettle</th>
<th>BeETLe</th>
</tr>
</thead>
<tbody>
<tr>
<td>License Type</td>
<td>LGPL</td>
<td>LGPL</td>
</tr>
<tr>
<td>Number of GIS formats supported</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Programming Language and libraries</td>
<td>Java</td>
<td>Java</td>
</tr>
<tr>
<td></td>
<td>JTS</td>
<td>JTS</td>
</tr>
<tr>
<td></td>
<td>GeoTools</td>
<td>GeoTools, Sextante</td>
</tr>
<tr>
<td>Raster Support</td>
<td>NO</td>
<td>SI</td>
</tr>
<tr>
<td>Support vector</td>
<td>SI</td>
<td>SI</td>
</tr>
<tr>
<td>Vector Analysis Operations</td>
<td>&gt; 25</td>
<td>&gt; 100</td>
</tr>
<tr>
<td>Raster analysis operations</td>
<td>No</td>
<td>&gt; 100</td>
</tr>
<tr>
<td>Parallel and distributed processing</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Visor Cartográfico integrado Integrated Mapping Viewer</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
BeETLe: goals

- Unified technology:
  - Easy to use
  - Software licenses
  - Less format conversions - higher throughput
BeETLe: goals

- Standardization and documentation of work-flows:
  - Reduce human error
  - Processes can be reproduced and audited
  - Non-interactive processes: processing and user time
BeETLe: goals

• Parallel execution
  - Using the ETL technology
  - GIS specific issues
BeETLe: goals

- Ability to process big data
  - Free software: can be improved and adapted
  - Benefits from parallel processing (ETL tools)
BeETLe: features

- Supports raster, vector and table data
- All the Sextante algorithms available in a single ETL tool
- Plus all the features provided by Kettle
Kettle Transformations and Jobs

• Jobs:
  – Sequential execution
  – Component-level parallelism

• Transformations:
  – Concurrent execution
  – Data parallelism and parallel segmentation
Technical challenges

- Sextante vs Kettle architectures: Data pull vs Data push
- Sextante is not designed for parallel computing: API and implementation must be adapted
Technical challenges (II)

- Big data processing: limitations on base libraries (GeoTools, Sextante)
- Data and task distribution; result consolidation
• 1\textsuperscript{st} milestone: Sextante as Kettle Jobs
  – no changes are required in Sextante
  – limited parallel execution
  – full range of Sextante algorithms available in Kettle
  – vector and raster support
• 2\textsuperscript{nd} milestone: Sextante as Kettle Transformations
  – bigger effort (requires changes in Sextante)
  – more powerful parallel execution
  – a sub-set of algorithms available as Transformations
Algorithm categories

- If the algorithm can be applied independently to different subsets of the data to get a valid result: *Directly parallelizable algorithms*. Examples:
  - raster sum, product, division, etc: can be calculated on overlapping tiles
  - vectorial buffer: can be calculated on each geometry
• The algorithm can be applied to different subsets of the data, but a global post-process (and/or pre-process) is necessary to get a valid result: Indirectly parallelizable algorithms. Examples:
  – Tabulate area algorithm: the result of tabulating tiles does not match the global result, but these partial result can be easily merged
• Sequential algorithms: when no parallelism is possible
Thinking out loud: OGC Services

- Remote services (WMS, WFS, etc) as data sources
  - Use WFS as vector data input
  - Use WMS or WCS as raster data input
- WPS services as BeETLe transformations
  - Similar to Sextante algorithms, but remotely processed using 3\textsuperscript{rd} party resources
Thinking out loud: WPS designer

• BeETLe as WPS flow modeller:
  – Design a complex data-flow in BeETLe
  – Be able to publish this data-flow as WPS service
Sextante is developing a Grass module that allows to execute Grass algorithms from Sextante
So we could use the Sextante connector to make Grass algorithms available in BeETLe
• Official blog: http://beetle-project.blogspot.com/

• OSOR Project (SVN, tickets, development documentation): http://forge.osor.eu/projects/etclusi/

• ETC-LUSI: http://etc-lusi.eionet.europa.eu/