Managing life-cycle information: from IGN France’s BDUni to the OME2 project

Noémie Grémeaux (IGN-France)
Contents

1. IGN France’s BDUni

2. The BDUni management system

3. Use and re-use of the BDUni management system
1. IGN France’s BDUni
IGN France’s BDUni

- IGN France’s central production database
- Created in 2006
- 1:10k multi-theme vector data
  - Buildings
  - Administrative
  - Hydrography
  - Land cover
  - Named places
  - Transport
  - …
- 200 tables and 1800 fields
- ~180 million live objects and 450 million historical objects
- 690 Go
- PostgreSQL/PostGIS
Updating the BDUni

1. IGN France’s BDUni

IGNF field agent
IGNF Central management team
External partner

PostgreSQL
geoconcept
Collaborative tools

BDUni
The BDUni management system

Workflow

- Many users are working on the BDUni at the same time so conflicts can occur.
- It is essential to track changes and know who has been doing what.

Main purposes of the BDUni management system:

- Keep track of all changes performed on the data through the different tools.
- Record every version of every object in the database, in order to visualize evolutions and provide incremental updates.
- Manage updates from the different tools and handle conflicts.
2. The BDUni management system
Implementation

7 technical fields on all tables

- `trcon_de_route`
- Columns (129)
  - `cleabs`
  - `gcms_detrut`
  - `gcms_date_creation`
  - `gcms_date_modification`
  - `gcms_date_destruction`
  - `gcms_numrec`
  - `gcms_fingerprint`

A history table for each table

To store all the successive versions of every object

- `transport_par_cable`
- `transport_par_cable.h`
- `trcon_de_randonnee_hivernale`
- `trcon_de_randonnee_hivernale.h`
- `trcon_de_route`
- `trcon_de_route.h`
- `trcon_de_voie_ferree`
- `trcon_de_voie_ferree.h`

PostgreSQL triggers

To fill the required information in case of modifications:

- UID updating rules (cleabs)
- Technical fields (dates, destroyed...)
- History tables.
### History table example

#### Main table

<table>
<thead>
<tr>
<th>clebs</th>
<th>gcms_dentruit</th>
<th>gcms_date_creation</th>
<th>gcms_date_modification</th>
<th>gcms_date_destruction</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRONROUT000000013612462</td>
<td>false</td>
<td>2006-08-18 12:13:50.587418</td>
<td>2023-08-26 08:48:30.650131</td>
<td>[null]</td>
</tr>
</tbody>
</table>

#### History table

<table>
<thead>
<tr>
<th>clebs</th>
<th>gcms_dentruit</th>
<th>gcms_date_creation</th>
<th>gcms_date_modification</th>
<th>gcms_date_destruction</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRONROUT000000013612462</td>
<td>false</td>
<td>2006-08-18 12:13:50.587418</td>
<td>[null]</td>
<td>[null]</td>
</tr>
</tbody>
</table>

- **Current version of the object**
- **Former versions of the object**
### Management rules

<table>
<thead>
<tr>
<th></th>
<th>Main table</th>
<th>History table</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Object created</strong></td>
<td>➢ New entry</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>➢ Fill creation date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>➢ Generate fingerprint</td>
<td></td>
</tr>
<tr>
<td><strong>Object modified</strong></td>
<td>➢ Fill modification date</td>
<td>➢ Record former version</td>
</tr>
<tr>
<td></td>
<td>➢ Update fingerprint</td>
<td></td>
</tr>
<tr>
<td><strong>Object deleted</strong></td>
<td>➢ Set « destroyed » field to true</td>
<td>➢ Record former version</td>
</tr>
<tr>
<td></td>
<td>➢ Fille deletion date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>➢ Update fingerprint</td>
<td></td>
</tr>
</tbody>
</table>
Handling conflicts

A user extracts some BDUni data and modifies an object → the object is modified only locally for now

The modified object is sent to the BDUni server with all its attributes, including its original gcms_fingerprint

Does gcms_fingerprint have the same value on the modified object as on the server version of the object?

Yes

The object is recorded on the server and the technical fields/history tables are filled according to the management rules

No

The object is rejected. Help is provided to the users to solve the conflict through a dedicated interface.

It means that another user has modified the object between the time when the data was extracted and the time when the modified object was sent.

Once he has solved the conflict, the user can try to send his modification again
3. Use and re-use of the BDUni management system
Accessing historical information

On the « Collaborative space » website* (https://espacecollaboratif.ign.fr)

Ex: https://espacecollaboratif.ign.fr/gcms/database/bdtopo_metropole/feature-type/troncon_de_route/feature/TRONROUT0000000013612462/versions

* For registered users only

3. Use and re-use of the BDUni management system
Delivering incremental updates

➢ A change-only updates edition of the BDUni is available every trimester.

➢ On the « Collaborative space » website, registered users can download change-only updates on request.
Setting up databases with life-cycle management for our partners

- Two simple parameters on the Collaborative space
- Automatic creation of technical fields and triggers

3. Use and re-use of the BDUni management system
Open Maps for Europe 2

- Consortium led by EuroGeographics
  - IGN France
  - Kadaster Netherlands
  - NGI Belgium
  - Hellenic Cadastre
  - Spanish Cadastre

- Co-funded by the European Commission

- To create a production process for:
  - An open large-scale database
  - Containing key themes: Administrative units, Transport network, Hydrography
  - With authoritative, harmonised and edge-matched data
  - Including life-cycle management (INSPIRE compliant).
Capitalizing on the BDUni experience

- Similar concepts: unique identifier and life-cycle attributes
- The same principles are being implemented in the OME2 database:
  - History tables
  - Re-use of the BDUni triggers to fill the life-cycle information.

→ A functional life-cycle management system with very little additional implementation.
Thank you for your attention

Contact: noemie.gremeaux@ign.fr