Geotrek Documentation

Version 2.103.2

Makina Corpus

mars 22, 2024

About

1 Geotrek **1**
- 1.1 Qu'est ce que Geotrek ? . . . 1
- 1.2 Les quatre briques . . . 2
- 1.3 Utilisateurs . . . 2
- 1.4 A qui appartient Geotrek ? . . . 2
- 1.5 Comment rejoindre la communauté ? . . . 2
- 1.6 Composants libres . . . 3

2 Présentation générale **5**
- 2.1 Architecture . . . 5
- 2.2 Page d'authentification . . . 5
- 2.3 Les modules de Geotrek . . . 5

3 Interface, navigation et saisie **13**
- 3.1 Navigation et saisie . . . 13
- 3.2 Paramètres de configuration . . . 15

4 Modules de gestion **19**
- 4.1 Les tronçons . . . 20
- 4.2 Les sentiers . . . 20
- 4.3 Les statuts . . . 21
- 4.4 Les aménagements . . . 21
- 4.5 La signalétique . . . 21
- 4.6 Les interventions . . . 23
- 4.7 Les chantiers . . . 23

5 Modules de valorisation **25**
- 5.1 Itinéraires . . . 26
- 5.2 Points d'intérêts (POI) . . . 33
- 5.3 Services . . . 33
- 5.4 Contenus touristiques . . . 33
- 5.5 Évènements touristiques . . . 33
- 5.6 Signalements . . . 33
- 5.7 Zones sensibles . . . 33
- 5.8 Activités outdoor . . . 34

6 Edition d'un objet **37**

6.1 Les calculs géométriques sur les objets . . . 37
6.2 Attributs aditionnels sur les objets . . . 40

7 Paramétrage **47**
7.1 Accès interface de configuration . . . 47
7.2 Paramétrages des modules . . . 47
7.3 Gestion des utilisateurs . . . 50
7.4 Gestion multilingue . . . 57
7.5 Configuration des portails . . . 57
7.6 Intégration des données SIT . . . 59
7.7 Intégration des fonds de cartes . . . 60
7.8 Intégration de pictogrammes . . . 60

8 APIs **61**
8.1 API Geotrek . . . 62
8.2 APIs externes . . . 62

9 Pages statiques **65**
9.1 Créer une page statique . . . 67
9.2 Construire une page statique . . . 67
9.3 Ajouter du contenu dans un bloc . . . 70
9.4 Astuces . . . 70

10 Tutoriels **71**
10.1 Vidéos . . . 71
10.2 Visualiser les données dans QGIS . . . 74

11 Installation **79**
11.1 Ubuntu package . . . 79
11.2 Docker . . . 82

12 Upgrade **83**
12.1 From Geotrek-admin >= 2.33 . . . 83
12.2 From Geotrek-admin <= 2.32 . . . 84
12.3 From Geotrek-admin <= 2.69.0 . . . 84
12.4 Server migration . . . 85
12.5 PostgreSQL . . . 85

13 Loading data **89**
13.1 Prerequisites for your data . . . 89
13.2 Load MNT raster . . . 90

14 Configuration **91**
14.1 Basic configuration update . . . 91
14.2 Custom setting file . . . 92
14.3 NGINX configuration . . . 92
14.4 Activate SSL / HTTPS . . . 92
14.5 Mandatory settings . . . 93
14.6 Users management . . . 94
14.7 Database users . . . 94

15 Configuration avancée **95**
15.1 Application settings . . . 96
15.2 Map settings . . . 99
15.3 Modules and components . . . 106

15.4 Feedback reports settings . . . 113
15.5 Attachments . . . 118
15.6 Interface . . . 120
15.7 Edition . . . 145
15.8 Settings for Geotrek-rando . . . 151
15.9 Settings for Geotrek-mobile . . . 154

16 Maintenance **157**
16.1 Application backup . . . 157
16.2 Application restore . . . 158
16.3 PostgreSQL optimization . . . 158
16.4 Access your database securely on your local machine (QGIS) . . . 158
16.5 Manage Cache . . . 158
16.6 Major evolutions from version 2.33 . . . 159

17 Synchronization **161**
17.1 Manual synchronization . . . 161
17.2 Automatic synchronization . . . 162
17.3 Synchronization options . . . 162
17.4 Synchronization filtered by source and portal . . . 163
17.5 Synchronization filtered by touristic content categories . . . 163
17.6 Synchronization with a distant Geotrek-rando server . . . 163
17.7 Geotrek-mobile app v3 . . . 163

18 Import data **165**
18.1 Import paths . . . 166
18.2 Import data from touristic data systems (SIT) . . . 166
18.3 Import data from a remote Geotrek instance . . . 171
18.4 Import other datas from a file . . . 171
18.5 Exploitation commands . . . 181

19 Contributing **185**
19.1 Conventions . . . 185
19.2 Definition of done . . . 186
19.3 Pull requests . . . 186
19.4 Release . . . 187
19.5 Other ways to contribute . . . 187

20 Development **189**
20.1 Quickstart . . . 189
20.2 Install git hooks . . . 190
20.3 Adding or upgrade dependencies . . . 190
20.4 Model modification . . . 190
20.5 Check quality . . . 191
20.6 Run tests . . . 192
20.7 Setup to run rando synchronization locally . . . 192
20.8 Setup to use screamshotter-related features locally . . . 193
20.9 Database reset . . . 193
20.10 Restore existing Database . . . 193
20.11 Mapentity development . . . 193
20.12 UML diagrams of data model . . . 193

21 Translating **195**
21.1 Getting started . . . 195
21.2 Translation features . . . 197

21.3 Release translations (only for github repository managers) . . . 199

22 Documentation **201**
22.1 Translate documentation . . . 201

23 Design **203**
23.1 Architecture . . . 203
23.2 Main components . . . 204
23.3 Django conventions twists . . . 205
23.4 Main roles of PostgreSQL triggers . . . 205

24 Troubleshooting **207**
24.1 Frequently encountered problems . . . 207
24.2 Signature check for debian packages . . . 209
24.3 Show main packages versions . . . 209

25 Authors **211**
25.1 Makina Corpus . . . 211
25.2 Parc national des Ecrins . . . 212
25.3 Others . . . 212
25.4 Credits . . . 212

26 CHANGELOG **213**
26.1 2.103.2 (2024-03-22) . . . 213
26.2 2.103.1 (2024-03-15) . . . 213
26.3 2.103.0 (2024-03-14) . . . 213
26.4 2.102.2 (2024-03-13) . . . 214
26.5 2.102.1 (2024-02-20) . . . 214
26.6 2.102.0 (2024-02-19) . . . 214
26.7 2.101.5 (2024-01-11) . . . 215
26.8 2.101.4 (2023-11-15) . . . 215
26.9 2.101.3 (2023-10-26) . . . 215
26.10 2.101.2 (2023-10-17) . . . 216
26.11 2.101.1 (2023-10-06) . . . 216
26.12 2.101.0 (2023-10-05) . . . 216
26.13 2.100.2 (2023-09-12) . . . 216
26.14 2.100.1 (2023-09-05) . . . 217
26.15 2.100.0 (2023-09-05) . . . 217
26.16 2.99.0 (2023-07-18) . . . 217
26.17 2.98.1 (2023-05-30) . . . 218
26.18 2.98.0 (2023-03-27) . . . 218
26.19 2.97.4 (2023-03-09) . . . 219
26.20 2.97.3 (2023-02-28) . . . 219
26.21 2.97.2 (2023-02-22) . . . 219
26.22 2.97.1 (2023-02-17) . . . 219
26.23 2.97.0 (2023-02-17) . . . 219
26.24 2.96.1 (2022-02-02) . . . 220
26.25 2.96.0 (2023-02-01) . . . 220
26.26 2.95.0 (2023-01-24) . . . 220
26.27 2.94.0 (2022-12-12) . . . 221
26.28 2.93.0 (2022-12-06) . . . 221
26.29 2.92.3 (2022-12-02) . . . 222
26.30 2.92.2 (2022-12-01) . . . 222
26.31 2.92.1 (2022-12-01) . . . 222
26.32 2.92.0 (2022-11-29) . . . 222

26.33 2.91.1 (2022-11-18) . . . 223
26.34 2.91.0 (2022-11-17) . . . 223
26.35 2.90.1 (2022-11-04) . . . 223
26.36 2.90.0 (2022-11-03) . . . 223
26.37 2.89.1 (2022-10-20) . . . 223
26.38 2.89.0 (2022-10-20) . . . 223
26.39 2.88.0 (2022-10-11) . . . 224
26.40 2.87.2 (2022-09-23) . . . 224
26.41 2.87.1 (2022-09-20) . . . 225
26.42 2.87.0 (2022-09-20) . . . 225
26.43 2.86.0 (2022-09-05) . . . 225
26.44 2.85.0 (2022-07-26) . . . 226
26.45 2.84.1 (2022-06-21) . . . 226
26.46 2.84.0 (2022-06-20) . . . 226
26.47 2.83.0 (2022-06-01) . . . 227
26.48 2.82.2 (2022-04-28) . . . 228
26.49 2.82.1 (2022-04-28) . . . 228
26.50 2.82.0 (2022-04-27) . . . 228
26.51 2.81.0 (2022-04-11) . . . 229
26.52 2.80.0 (2022-04-05) . . . 229
26.53 2.79.0 (2022-03-25) . . . 229
26.54 2.78.0 (2022-03-22) . . . 230
26.55 2.77.3 (2022-03-18) . . . 230
26.56 2.77.2 (2022-03-15) . . . 230
26.57 2.77.1 (2022-03-11) . . . 230
26.58 2.77.0 (2022-03-09) . . . 231
26.59 2.76.4 (2022-03-07) . . . 231
26.60 2.76.3 (2022-02-09) . . . 231
26.61 2.76.2 (2022-02-08) . . . 231
26.62 2.76.1 (2022-02-07) . . . 232
26.63 2.76.0 (2022-02-02) . . . 232
26.64 2.75.0 (2022-01-07) . . . 233
26.65 2.74.1 (2021-12-21) . . . 233
26.66 2.74.0 (2021-12-17) . . . 233
26.67 2.73.0 (2021-12-13) . . . 234
26.68 2.72.0 (2021-11-16) . . . 234
26.69 2.71.0 (2021-11-03) . . . 235
26.70 2.70.0 (2021-11-02) . . . 235
26.71 2.69.0 (2021-10-22) . . . 235
26.72 2.68.1 (2021-10-21) . . . 235
26.73 2.68.0 (2021-10-20) . . . 235
26.74 2.67.0 (2021-10-12) . . . 236
26.75 2.66.0 (2021-09-27) . . . 236
26.76 2.65.0 (2021-09-21) . . . 236
26.77 2.64.0 (2021-09-14) . . . 237
26.78 2.63.0 (2021-09-03) . . . 237
26.79 2.62.0 (2021-07-06) . . . 237
26.80 2.61.1 (2021-06-28) . . . 238
26.81 2.61.0 (2021-06-25) . . . 238
26.82 2.60.0 (2021-06-25) . . . 238
26.83 2.59.0 (2021-06-07) . . . 238
26.84 2.58.0 (2021-05-20) . . . 239
26.85 2.57.0 (2021-04-28) . . . 239
26.86 2.56.0 (2021-04-27) . . . 239

26.87 2.55.1 (2021-04-15) . . . 240
26.88 2.55.0 (2021-04-09) . . . 240
26.89 2.54.0 (2021-04-09) . . . 240
26.90 2.53.1 (2021-04-07) . . . 240
26.91 2.53.0 (2021-04-01) . . . 240
26.92 2.52.0 (2021-03-25) . . . 241
26.93 2.51.2 (2021-03-16) . . . 241
26.94 2.51.1 (2021-03-05) . . . 241
26.95 2.51.0 (2021-03-02) . . . 241
26.96 2.50.0 (2021-02-19) . . . 242
26.97 2.49.0 (2021-02-09) . . . 242
26.98 2.48.1 (2021-02-05) . . . 242
26.99 2.48.0 (2021-02-03) . . . 242
26.100 2.47.2 (2021-01-28) . . . 243
26.101 2.47.1 (2021-01-27) . . . 243
26.102 2.47.0 (2021-01-26) . . . 243
26.103 2.46.0 (2021-01-25) . . . 243
26.104 2.45.0 (2021-01-10) . . . 244
26.105 2.44.0 (2020-12-18) . . . 244
26.106 2.43.1 (2020-12-10) . . . 244
26.107 2.43.0 (2020-12-10) . . . 244
26.108 2.42.0 (2020-12-04) . . . 245
26.109 2.41.2 (2020-11-27) . . . 245
26.110 2.41.1 (2020-11-25) . . . 245
26.111 2.41.0 (2020-11-25) . . . 245
26.112 2.40.1 (2020-11-23) . . . 245
26.113 2.40.0 (2020-11-18) . . . 246
26.114 2.39.1 (2020-10-28) . . . 246
26.115 2.39.0 (2020-10-27) . . . 246
26.116 2.38.6 (2020-10-20) . . . 246
26.117 2.38.5 (2020-10-20) . . . 246
26.118 2.38.4 (2020-10-16) . . . 247
26.119 2.38.3 (2020-10-05) . . . 247
26.120 2.38.2 (2020-09-24) . . . 247
26.121 2.38.1 (2020-09-22) . . . 247
26.122 2.38.0 (2020-09-21) . . . 247
26.123 2.37.0 (2020-09-16) . . . 248
26.124 2.36.1 (2020-09-04) . . . 248
26.125 2.36.0 (2020-09-01) . . . 248
26.126 2.35.1 (2020-08-24) . . . 248
26.127 2.35.0 (2020-08-21) . . . 248
26.128 2.34.0 (2020-07-10) . . . 249
26.129 2.33.13 (2020-07-01) . . . 249
26.130 2.33.12 (2020-06-23) . . . 249
26.131 2.33.11 (2020-06-05) . . . 249
26.132 2.33.10 (2020-06-02) . . . 249
26.133 2.33.9 (2020-06-02) . . . 250
26.134 2.33.8 (2020-05-22) . . . 250
26.135 2.33.7 (2020-05-18) . . . 250
26.136 2.33.6 (2020-05-14) . . . 250
26.137 2.33.5 (2020-05-13) . . . 250
26.138 2.33.4 (2020-05-04) . . . 251
26.139 2.33.3 (2020-04-28) . . . 251
26.140 2.33.2 (2020-04-28) . . . 251

26.141 2.33.1 (2020-04-28) . . . 251
26.142 2.33.0 (2020-04-28) . . . 251
26.143 2.32.11 (2020-03-17) . . . 252
26.144 2.32.10 (2020-03-11) . . . 252
26.145 2.32.9 (2020-03-06) . . . 252
26.146 2.32.8 (2020-03-05) . . . 252
26.147 2.32.7 (2020-03-02) . . . 253
26.148 2.32.6 (2020-02-28) . . . 253
26.149 2.32.5 (2020-02-18) . . . 253
26.150 2.32.4 (2020-02-12) . . . 253
26.151 2.32.3 (2020-01-27) . . . 253
26.152 2.32.2 (2020-01-09) . . . 253
26.153 2.32.1 (2019-12-20) . . . 254
26.154 2.32.0 (2019-12-13) . . . 254
26.155 2.31.0 (2019-12-06) . . . 254
26.156 2.30.0 (2019-11-26) . . . 254
26.157 2.29.15 (2019-11-12) . . . 254
26.158 2.29.14 (2019-11-04) . . . 254
26.159 2.29.13 (2019-10-30) . . . 255
26.160 2.29.12 (2019-10-23) . . . 255
26.161 2.29.11 (2019-10-17) . . . 255
26.162 2.29.10 (2019-10-08) . . . 255
26.163 2.29.9 (2019-10-02) . . . 256
26.164 2.29.8 (2019-09-26) . . . 256
26.165 2.29.7 (2019-09-25) . . . 256
26.166 2.29.6 (2019-09-19) . . . 256
26.167 2.29.5 (2019-09-13) . . . 256
26.168 2.29.4 (2019-09-09) . . . 256
26.169 2.29.3 (2019-08-28) . . . 257
26.170 2.29.2 (2019-08-28) . . . 257
26.171 2.29.1 (2019-08-26) . . . 257
26.172 2.29.0 (2019-08-20) . . . 257
26.173 2.28.0 (2019-08-09) . . . 257
26.174 2.27.12 (2019-07-22) . . . 258
26.175 2.27.11 (2019-07-17) . . . 258
26.176 2.27.10 (2019-07-10) . . . 258
26.177 2.27.9 (2019-07-01) . . . 258
26.178 2.27.8 (2019-06-28) . . . 258
26.179 2.27.7 (2019-06-26) . . . 258
26.180 2.27.6 (2019-06-04) . . . 259
26.181 2.27.5 (2019-05-29) . . . 259
26.182 2.27.4 (2019-05-27) . . . 259
26.183 2.27.3 (2019-05-23) . . . 259
26.184 2.27.2 (2019-05-14) . . . 259
26.185 2.27.1 (2019-05-06) . . . 260
26.186 2.27.0 (2019-05-02) . . . 260
26.187 2.26.5 (2019-04-19) . . . 260
26.188 2.26.4 (2019-04-18) . . . 260
26.189 2.26.3 (2019-04-12) . . . 260
26.190 2.26.2 (2019-04-10) . . . 260
26.191 2.26.1 (2019-04-03) . . . 261
26.192 2.26.0 (2019-04-01) . . . 261
26.193 2.25.3 (2019-03-26) . . . 261
26.194 2.25.2 (2019-03-26) . . . 261

26.1952.25.1 (2019-03-25) . . . 261
26.1962.25.0 (2019-03-25) . . . 261
26.1972.24.8 (2019-03-15) . . . 262
26.1982.24.7 (2019-03-13) . . . 262
26.1992.24.6 (2019-03-07) . . . 262
26.2002.24.5 (2019-03-06) . . . 262
26.2012.24.4 (2019-03-01) . . . 262
26.2022.24.3 (2019-02-28) . . . 262
26.2032.24.2 (2019-02-26) . . . 263
26.2042.24.1 (2019-02-12) . . . 263
26.2052.24.0 (2019-01-28) . . . 263
26.2062.23.0 (2019-01-24) . . . 263
26.2072.22.10 (2019-01-09) . . . 263
26.2082.22.9 (2019-01-09) . . . 264
26.2092.22.8 (2019-01-03) . . . 264
26.2102.22.7 (2019-01-03) . . . 264
26.2112.22.6 (2019-01-02) . . . 264
26.2122.22.5 (2018-12-19) . . . 264
26.2132.22.4 (2018-12-19) . . . 264
26.2142.22.3 (2018-12-14) . . . 265
26.2152.22.2 (2018-12-10) . . . 265
26.2162.22.1 (2018-11-27) . . . 265
26.2172.22.0 (2018-11-27) . . . 265
26.2182.21.1 (2018-09-18) . . . 265
26.2192.21.0 (2018-09-04) . . . 266
26.2202.20.1 (2018-07-16) . . . 266
26.2212.20.0 (2018-06-27) . . . 266
26.2222.19.1 (2018-05-28) . . . 266
26.2232.19.0 (2018-05-22) . . . 267
26.2242.18.5 (2018-05-07) . . . 267
26.2252.18.4 (2018-05-02) . . . 267
26.2262.18.3 (2018-04-27) . . . 267
26.2272.18.2 (2018-04-27) . . . 267
26.2282.18.1 (2018-03-26) . . . 268
26.2292.18.0 (2018-03-22) . . . 268
26.2302.17.3 (2018-03-23) . . . 268
26.2312.17.2 (2018-02-07) . . . 268
26.2322.17.1 (2018-02-02) . . . 268
26.2332.17.0 (2018-01-15) . . . 269
26.2342.16.1 (2018-01-10) . . . 269
26.2352.16.0 (2017-12-21) . . . 269
26.2362.15.2 (2017-09-28) . . . 269
26.2372.15.1 (2017-08-23) . . . 269
26.2382.15.0 (2017-07-13) . . . 270
26.2392.14.3 (2017-07-03) . . . 270
26.2402.14.2 (2017-06-21) . . . 270
26.2412.14.1 (2017-06-22) . . . 270
26.2422.14.0 . . . 270
26.2432.13.0 (2017-03-02) . . . 271
26.2442.12.0 (2017-02-16) . . . 271
26.2452.11.5 (2017-02-06) . . . 271
26.2462.11.4 (2017-02-01) . . . 271
26.2472.11.3 (2016-11-15) . . . 271
26.2482.11.2 (2016-09-15) . . . 272

26.249 2.11.1 (2016-08-17) . . . 272
26.250 2.11.0 (2016-08-02) . . . 272
26.251 2.10.4 (2016-05-19) . . . 272
26.252 2.10.3 (2016-05-11) . . . 273
26.253 2.10.2 (2016-04-12) . . . 273
26.254 2.10.1 (2016-03-17) . . . 273
26.255 2.10.0 (2016-03-03) . . . 273
26.256 2.9.3 (2016-02-25) . . . 274
26.257 2.9.2 (2016-02-17) . . . 274
26.258 2.9.1 (2016-02-10) . . . 274
26.259 2.9.0 (2016-02-04) . . . 274
26.260 2.8.1 (2016-01-29) . . . 274
26.261 2.8.0 (2016-01-28) . . . 274
26.262 2.7.2 (2016-01-26) . . . 275
26.263 2.7.1 (2016-01-18) . . . 275
26.264 2.7.0 (2016-01-14) . . . 275
26.265 2.6.0 (2015-12-30) . . . 275
26.266 2.5.2 (2015-12-29) . . . 275
26.267 2.5.1 (2015-12-18) . . . 276
26.268 2.5.0 (2015-12-08) . . . 276
26.269 2.4.4 (2015-12-02) . . . 276
26.270 2.4.3 (2015-11-27) . . . 276
26.271 2.4.2 (2015-11-26) . . . 276
26.272 2.4.1 (2015-11-25) . . . 276
26.273 2.4.0 (2015-11-18) . . . 277
26.274 2.3.0 (2015-11-09) . . . 277
26.275 2.2.0 (2015-10-09) . . . 277
26.276 2.1.0 (2015-09-29) . . . 277
26.277 2.0.0 (2015-07-20) . . . 278
26.278 0.35.1 (2015-07-17) . . . 278
26.279 0.35.0 (2015-07-10) . . . 278
26.280 0.34.0 (2015-05-20) . . . 279
26.281 0.33.4 (2015-04-07) . . . 279
26.282 0.33.3 (2015-04-01) . . . 279
26.283 0.33.2 (2015-04-01) . . . 279
26.284 0.33.1 (2015-03-25) . . . 279
26.285 0.33.0 (2015-03-25) . . . 280
26.286 0.32.2 (2015-03-06) . . . 280
26.287 0.32.1 (2015-03-04) . . . 280
26.288 0.32.0 (2015-03-04) . . . 280
26.289 0.31.0 (2015-03-02) . . . 281
26.290 0.30.0 (2015-02-19) . . . 281
26.291 0.29.0 (2015-02-04) . . . 281
26.292 0.28.8 (2014-12-22) . . . 282
26.293 0.28.7 (2014-12-22) . . . 282
26.294 0.28.6 (2014-12-18) . . . 282
26.295 0.28.5 (2014-12-09) . . . 282
26.296 0.28.4 (2014-11-21) . . . 283
26.297 0.28.3 (2014-11-12) . . . 283
26.298 0.28.2 (2014-11-05) . . . 283
26.299 0.28.1 (2014-11-05) . . . 284
26.300 0.28.0 (2014-11-04) . . . 284
26.301 0.27.2 (2010-10-14) . . . 285
26.302 0.27.1 (2010-10-13) . . . 285

26.303 0.27.0 (2010-10-09) . . . 285
26.304 0.26.3 (2014-09-15) . . . 286
26.305 0.26.2 (2014-08-22) . . . 286
26.306 0.26.1 (2014-08-21) . . . 286
26.307 0.26.0 (2014-08-21) . . . 286
26.308 0.25.2 (2014-08-14) . . . 287
26.309 0.25.1 (2014-08-01) . . . 287
26.310 0.25.0 (2014-08-01) . . . 287
26.311 0.24.3 (2014-06-27) . . . 288
26.312 0.24.2 (2014-06-27) . . . 288
26.313 0.24.1 (2014-06-26) . . . 288
26.314 0.24.0 (2014-06-23) . . . 288
26.315 0.23.5 (2014-06-19) . . . 290
26.316 0.23.4 (2014-06-18) . . . 290
26.317 0.23.3 (2014-06-18) . . . 290
26.318 0.23.2 (2014-06-13) . . . 290
26.319 0.23.1 (2014-05-22) . . . 291
26.320 0.23 (2014-05-22) . . . 291
26.321 0.22.6 (2014-04-27) . . . 293
26.322 0.22.5 (2014-03-19) . . . 293
26.323 0.22.4 (2014-03-06) . . . 293
26.324 0.22.3 (2014-02-17) . . . 293
26.325 0.22.2 (2014-02-14) . . . 294
26.326 0.22.1 (2014-02-13) . . . 294
26.327 0.22 (2014-02-12) . . . 294
26.328 0.21.2 (2014-02-04) . . . 295
26.329 0.21.1 (2013-12-11) . . . 295
26.330 0.21 (2013-11-28) . . . 296
26.331 0.20.9 (2013-10-30) . . . 296
26.332 0.20.8 (2013-10-22) . . . 296
26.333 0.20.7 (2013-10-16) . . . 297
26.334 0.20.6 (2013-10-14) . . . 297
26.335 0.20.5 (2013-10-09) . . . 297
26.336 0.20.4 (2013-10-09) . . . 297
26.337 0.20.3 (2013-09-30) . . . 297
26.338 0.20.2 (2013-08-27) . . . 298
26.339 0.20.1 (2013-08-26) . . . 298
26.340 0.20 (2013-08-23) . . . 298
26.341 0.19.1 (2013-07-15) . . . 299
26.342 0.19 (2013-07-12) . . . 299
26.343 0.18 (2013-06-06) . . . 300
26.344 0.17 (2013-05-17) . . . 300

Index **301**

CHAPITRE 1

Geotrek

— *Qu'est ce que Geotrek ?*
— *Les quatre briques*
— *Utilisateurs*
— *A qui appartient Geotrek ?*
— *Comment rejoindre la communauté ?*
— *Composants libres*
  — *Geotrek-admin*
  — *Geotrek-rando*
  — *Geotrek-rando-widget*
  — *Geotrek-mobile*

1.1 Qu'est ce que Geotrek ?

Geotrek est un ensemble d'outils web dédiés à la **gestion et la valorisation** des activités de randonnées, de sports de pleine nature et des informations touristiques.

Ils sont open source et peuvent ainsi être déployés librement par les structures et territoires qui le souhaitent, mais aussi les faire évoluer, ou bénéficier des évolutions réalisées par d'autres structures sans avoir à les re-financer.

Ces outils permettent la gestion de **nombreuses pratiques sportives** liées à des linéaires (randonnée pédestre, à vélo, en VTT, à cheval, avec poussette, etc.) mais également à des **contenus outdoor** non linéaires (escalade, vol libre, sports d'eau vive, etc.).

C'est l'**application de référence** des parcs naturels, mais aussi de nombreuses structures publiques (conseils départementaux, communautés de communes, comités régionaux du tourisme, offices du tourisme, etc.).

Utiliser Geotrek vous permettra :
— d'intégrer une **communauté de gestionnaires de randonnées très active**
— de **bénéficier de l'ensemble des évolutions** réalisées par les autres utilisateurs. Le projet est vivant et les outils s'améliorent sans cesse
— de disposer d'une **offre complète** qui pourra également être utilisée par d'autres acteurs du territoire

— de facilement pouvoir **accéder à d'autres fonctionnalités** quand le besoin s'en fera ressentir : site web, impression de fiches de parcours, gestion de la signalétique, organisation des travaux…
— d'exporter facilement vos contenus valorisables (itinéraires, POIs, contenus touristiques …) vers d'autres **plateformes touristiques nationales** (IGNrando", Visorando, Outdooractive, Apidae, Cirkwi, etc.)

1.2 Les quatre briques

La suite logicielle Geotrek dispose de quatre briques à la fois distincts et complémentaires :
— *Geotrek-admin* : outil de gestion et de saisie de l'ensemble des informations, intégrant les données des Systèmes d'Informations Touristiques (SIT) et pouvant être connecté à votre SIG ou à des systèmes d'information transport
— Geotrek-rando : site web, reprenant les informations saisies dans Geotrek-admin, à destination des internautes grand public
— Geotrek-mobile : application mobile fonctionnant sous Android et iOS, reprenant des informations saisies dans Geotrek-admin et optimisées pour l'usage mobile (volume, impact sur la batterie, hors-ligne, géolocalisation…)
— Geotrek-rando-widget : nouveau composant web permettant de valoriser une offre de contenus touristiques et de randonnées auprès des usagers du territoire, en l'intégrant dans un site internet existant

Note : Cette documentation ne traite que de **Geotrek-Admin**, chaque brique ayant sa propre documentation.

1.3 Utilisateurs

L'application Geotrek, **destinée à deux types de public**, est une solution web qui apporte :
— des fonctionnalités de gestion des informations (itinéraires, sites outdoor, points d'intérêts, description, interprétation, médias…) et de gestion des infrastructures (signalétique, aménagements, travaux, réglementation…) pour les utilisateurs gérant un territoire (**Geotrek-admin**)
— des fonctionnalités simples et ludiques de recherche et de consultation d'itinéraires pour les internautes et les mobinautes (**Geotrek-rando V3**, **Geotrek-mobile** et **Geotrek-rando-widget**).
Pour retrouver plus d'informations sur la suite applicative Geotrek, rendez-vous sur geotrek.fr.

1.4 A qui appartient Geotrek ?

Geotrek est un produit libre et open source avec une importante communauté d'utilisateurs.

Retrouvez toutes les infos sur la genèse du produit, son modèle communautaire ainsi que les principales structures contributrices sur le site geotrek.fr.

1.5 Comment rejoindre la communauté ?

— Rejoignez la mailing list ! Envoyez un mail à `geotrek-fr+subscribe@googlegroups.com` et vous recevrez automatiquement une invitation.
— Ouvrir un ticket lorsqu'un bug est détecté
— Ouvrir un ticket pour proposer une suggestion ou une nouvelle fonctionnalité
— Rejoindre le canal de discussion Matrix afin d'échanger directement avec des membres de la communauté Geotrek

1.6 Composants libres

L'application Geotrek utilise les technologies open source suivantes :

1.6.1 Geotrek-admin

- **Python / Django**, l'épine dorsale de l'application qui prend en charge les principales fonctionnalités comme le module de configuration, l'exploitation de la base de données, la gestion des utilisateurs et de leurs droits ou l'intégration avec les bibliothèques cartographiques. La richesse de son écosystème permet de concevoir des applications aux possibilités infinies, en favorisant la production d'applications sécurisées, solides (tests automatiques) et robustes (Python).
- **PostgreSQL / PostGIS** pour la base de données. La totalité des données de l'application est stockée dans une instance PostgreSQL avec l'extension spatiale PostGIS :
  - attributs, comptes utilisateurs…,
  - géométries,
  - raster (Modèle Numérique Terrain).

1.6.2 Geotrek-rando

- **Next.js** (*React, Typescript*),
- **Leaflet**, utilisé comme librairie cartographique

1.6.3 Geotrek-rando-widget

- **Stencil**, framework permettant de créer des composants web personnalisables et légers.
- **Leaflet**, utilisé comme librairie cartographique

1.6.4 Geotrek-mobile

- **Angular**, framework utilisé pour l'application Geotrek-mobile.
- **Ionic**, composant UI
- **Capacitor**, boîte à outils nécessaires à la création d'applications mobiles
- **MapLibre**, utilisé comme librairie cartographique

CHAPITRE 2

Présentation générale

— *Architecture*
— *Page d'authentification*
— *Les modules de Geotrek*
  — *Modules de gestion*
  — *Modules de valorisation*

2.1 Architecture

L'application est composée :
— d'une **page d'authentification**, demandant la saisie d'un login et d'un mot de passe
— d'une **interface de consultation des objets de chaque module**
  — avec la liste de sélection déroulante des modules avec compteur des résultats
  — un onglet latéral de sélection des modules
  — un accès aux paramètres de configuration
  — un bouton déconnexion de l'application
— de **paramètres d'administration** (gestion des droits, des listes déroulantes…)

2.2 Page d'authentification

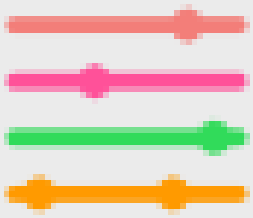
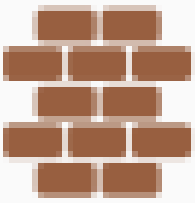
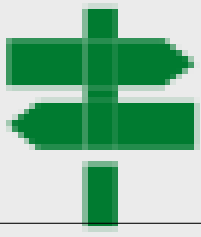
2.3 Les modules de Geotrek

Geotrek-admin est composé de deux types de modules pour la gestion et la valorisation des contenus.

FIG. 1 – Accès à Geotrek-admin via un login et un mot de passe

2.3.1 Modules de gestion

Tableau 1 – Modules de gestion

Icône	Module	Description
	Tronçon	C'est l'équivalent du réseau routier : ils constituent le support des tracés des objets linéaires (itinéraires, statuts, interventions, aménagements…). Leur modification est relativement rare (fermeture d'un chemin, éboulement…).
	Sentier	Les itinéraires, non pas de randonnée, mais de gestion, correspondent au départ et à l'arrivée d'un sentier.
	*Statut*	Gestion des organismes ayant la compétence sentiers, gestionnaires des travaux et de la signalétique. Type physique (route, piste, sente, etc.)
	*Aménagement*	Décrit et localise les ouvrages, le mobilier, les équipements.
	*Signalétique*	Reprend l'ensemble de la signalétique d'accueil, d'information et d'orientation à destination des randonneurs. Gestion des fichiers associés comme les BAT, les maquettes…

2.3.2 Modules de valorisation

Tableau 2 – Modules de valorisation

Icône	Module	Description
	Itinéraire	Randonnées présentées au grand public : l'itinéraire est défini selon la géométrie des tronçons empruntés. L'ajout, la modification sont fréquents. La dé-publication est fortement déconseillée pour le référencement et les passerelles avec d'autres outils.
	Points d'intérêts (POI)	Ponctuels à découvrir associés aux randonnées : en fonction de leur emplacement, ils sont associés automatiquement aux randonnées. Leur catégorie détermine leur pictogramme (faune, flore, patrimoine, équipements…).
	Services	Informations pratiques comme les points d'eau, passages délicats… selon la typologie souhaitée. Ils n'ont pas de description ni de nom, ni de photo et sont uniquement affichés sur la carte de l'itinéraire sous forme de pictogramme.
	Contenus touristiques	Correspond aux services touristiques pouvant être importés depuis des Systèmes d'Informations Touristiques (SIT), qui permettent d'alimenter automatiquement la base de données Geotrek. Celles-ci sont regroupées dans des catégories de type : hébergements, musées, restaurants, produits du terroir…
	Évènements touristiques	Correspond aux animations pouvant être importées depuis

Chaque module est accessible depuis le bandeau vertical situé à gauche de l'écran.

CHAPITRE 3

Interface, navigation et saisie

— *Navigation et saisie*
  — *Vue liste*
  — *Vue détail*
  — *Vue édition*
— *Paramètres de configuration*

3.1 Navigation et saisie

Les résultats sont affichés sous forme de carte et liste puis on accède aux détails des objets.

3.1.1 Vue liste

Tous les modules sont construits de la même façon :

— une liste paginée des objets du module
— la possibilité de filtrer la liste selon des attributs ou de faire une recherche libre
— la possibilité de filtrer selon l'étendue de la carte
— la sélection coordonnée (liste → carte, carte → liste)
— la possibilité d'exporter les résultats au format CSV, SHAPEFILE et en GPX
— une carte dans laquelle il est possible de naviguer (déplacer, zoomer), d'afficher en plein écran, de mesurer une longueur, d'exporter une image de la carte, de réinitialiser l'étendue, de zoomer sur une commune ou un secteur et de superposer des données des autres modules (contours communes / secteurs / physique / foncier / gestionnaires…)
— l'accès à la vue détail d'un objet au clic sur celui-ci

Note :

— Au survol d'un objet dans la liste, celui-ci est mis en surbrillance sur la carte.
— Il est possible d'ajouter de nouveaux objets depuis chaque module.
— Un clic sur un objet dans la liste ou la carte permet d'accéder à la fiche détaillée de celui-ci.

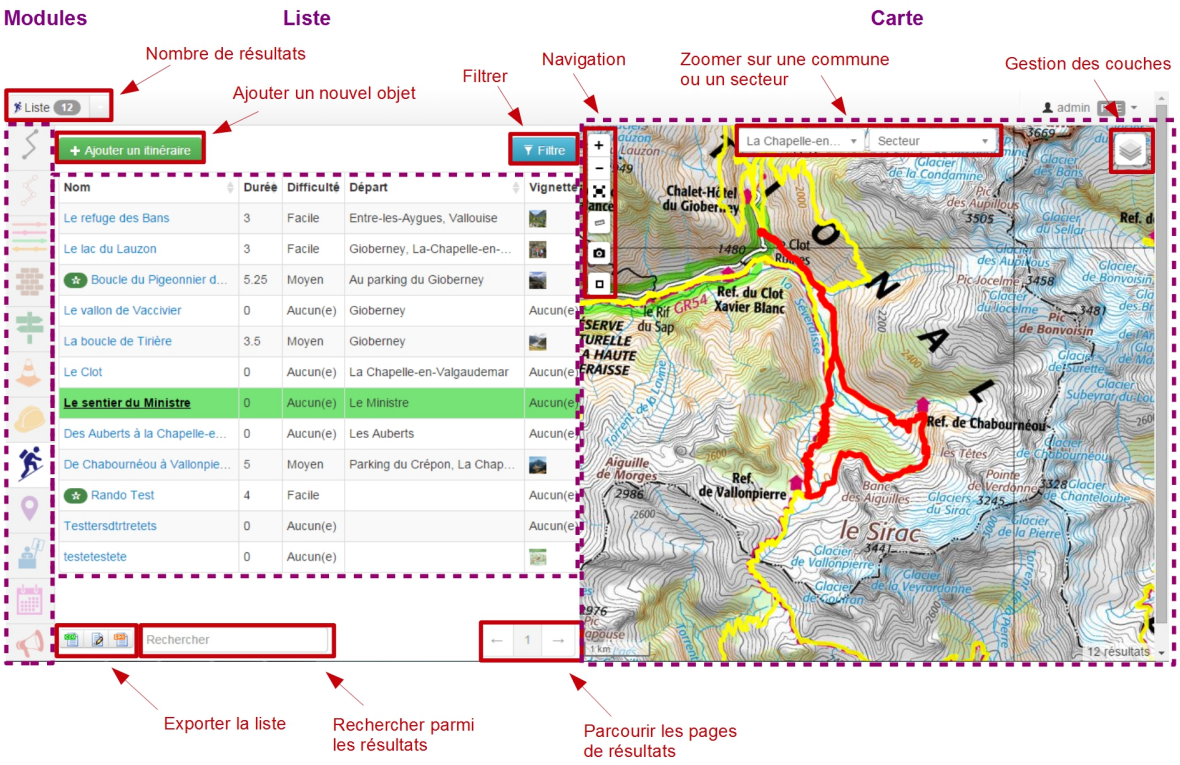


FIG. 1 – Vue liste avec la carte

Avertissement :

— La liste des résultats est filtrée en fonction de l'étendue de la carte affichée.

3.1.2 Vue détail

A partir de chaque module, il est possible d'afficher la fiche détail d'un objet en cliquant sur celui-ci dans la liste ou la carte du module. Les objets de chaque module peuvent ainsi être affichés individuellement dans une fiche détail pour en consulter tous les attributs, tous les objets des autres modules qui intersectent l'objet, les fichiers qui y sont attachés et l'historique des modifications de l'objet.

Voici les possibilités de la fiche détail :

— le récapitulatif des attributs (saisis et calculés)
— récupérer automatiquement des informations liées (communes, secteurs, POI…)
— ajouter des fichiers (redimensionnement automatique pour les photos)
— l'accès à la vue d'édition selon les droits de l'utilisateur connecté
— l'export GPX, KML, OpenDocument, Word, PDF

Note :

— Lorsque le statut de publication d'un objet est activé, celui-ci ainsi que tous ses attributs associés, sont mis en ligne.
— À tout moment et ce sur chaque module, les informations peuvent être soit mises en ligne, soit désactivées, voire supprimées.

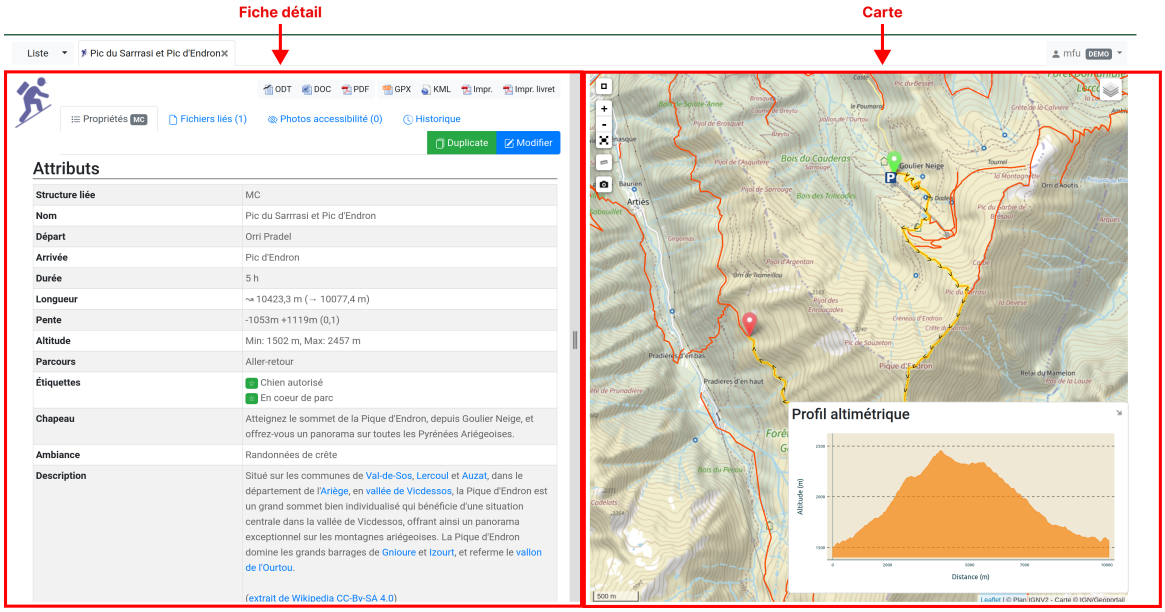


FIG. 2 – Fiche détail d'un itinéraire

— Ne sont mises en ligne que les informations choisies et disponibles. Les catégories non encore alimentées ne seront pas visibles pour le grand public. Ainsi si vous créez une nouvelle pratique « Raquettes », celle-ci ne sera pas mise ne ligne tant que vous n'aurez pas publié un premier itinéraire de raquettes.

3.1.3 Vue édition

— Saisie des champs multilingues
— Saisie des tracés
— Possibilité de forcer des points de passage (détours, boucles, aller-retours)
— Édition WYSIWYG des champs texte
— Ajout de couches locales en superposition à partir de fichiers GPX ou KML (aide à la saisie)
— Outils de mesure

3.2 Paramètres de configuration

Toutes les listes de choix (thématiques, pratiques, parcours…) sont administrables depuis le module de configuration, selon les droits dont dispose l'utilisateur connecté.

Voir la section *Paramétrage des modules*

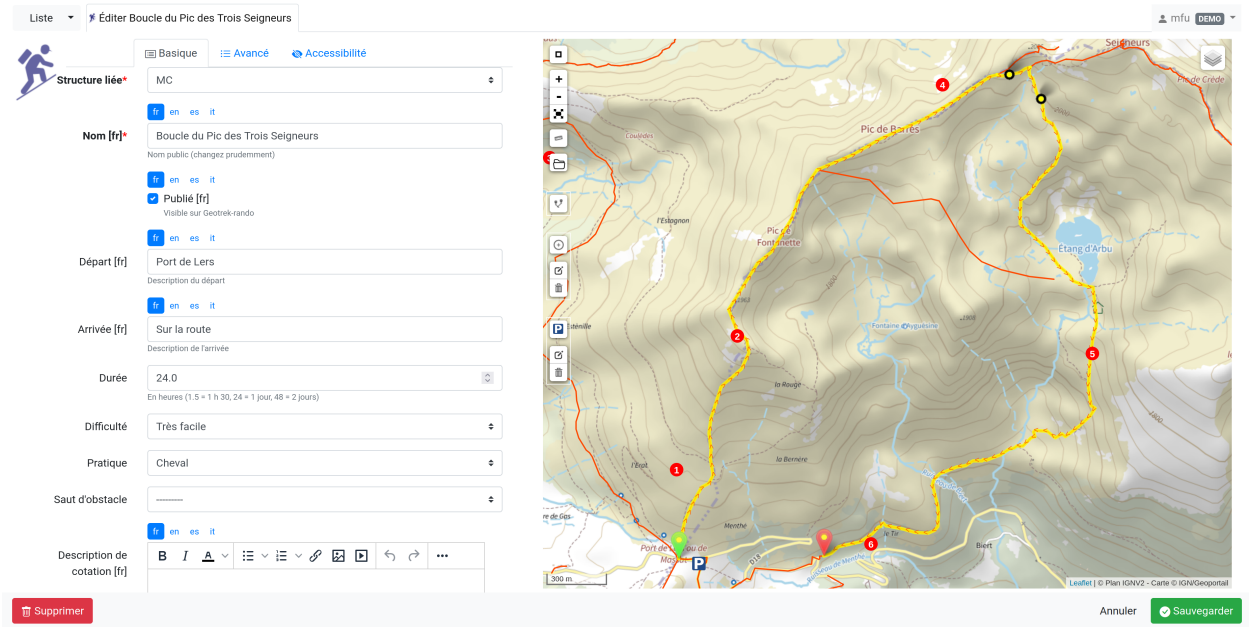


FIG. 3 – Fiche détail d'un itinéraire en édition

FIG. 4 – Interface complète du module de configuration

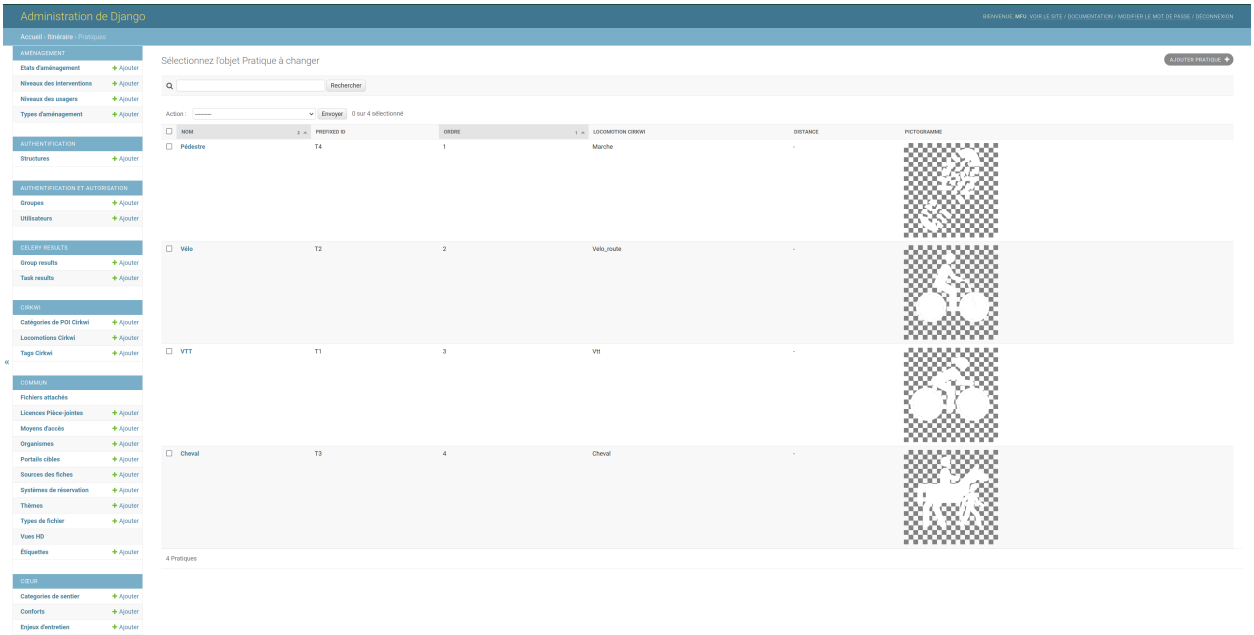


FIG. 5 – Exemple d'édition des pratiques et de leur pictogramme

CHAPITRE 4

Modules de gestion

— *Les tronçons*
— *Les sentiers*
— *Les statuts*
— *Les aménagements*
— *La signalétique*
— *Les interventions*
— *Les chantiers*

Geotrek-admin comporte un certain nombre de modules de gestion des sentiers (tronçons, sentiers, statuts, aménagements, signalétique, interventions et chantiers).

Les tronçons sont les éléments de base sur lesquels s'appuient l'ensemble des objets des autres modules, en utilisant la segmentation dynamique (https://makina-corpus.com/blog/metier/2014/la-segmentation-dynamique).

Les modules signalétique et aménagement ont initialement été conçus dans une logique d'inventaire avec des possibilités de description basiques et génériques. Pour tout complément, il est possible d'attacher un ou plusieurs fichiers joints à chaque objet (photos, PDF, tableurs…).

Les modules interventions et chantiers ont été conçus de façon à permettre à la fois un inventaire et un suivi des travaux (prévisionnel, administratif et financier).

Pour les territoires qui le souhaitent, sur le volet gestion, les valeurs des listes déroulantes peuvent être différenciées par structure, afin que chaque structure travaillant sur une même instance Geotrek-admin puisse avoir des typologies différentes si nécéssaire (types de signalétique, d'aménagements, d'organismes…).

Néanmoins, pour les territoires qui souhaitent mutualiser les mêmes valeurs dans les listes sans avoir à les renseigner pour chaque structure, il est possible de partager des typologies entre les différentes structures en ne renseignant tout simplement pas ce champ. Un compte utilisateur appartenant à une structure X n'aura accès qu'aux typologies associées à celle-ci, ainsi qu'aux typologies partagées. De même, ce compte utilisateur ne pourra pas modifier ou supprimer des objets appartenant à une autre structure (c'est-à-dire créés par un compte utilisateur appartenant à une autre structure), sauf à avoir des permissions particulières.

Lors de la saisie d'un objet sur la carte, il est possible d'afficher une couche SIG ou un relevé GPX sur la carte lors de la création d'un objet sur la carte pour pouvoir le visualiser et le localiser sur la carte (`Charger un fichier local`

(GPX, KML, GeoJSON).

4.1 Les tronçons

C'est le socle essentiel et central de Geotrek. Un tronçon est un objet linéaire, entre deux intersections. Le mécanisme de segmentation dynamique permet de ne pas devoir le recouper pour y rattacher des informations.

Les tronçons peuvent être soit numérisés dans Geotrek-admin, soit importés directement dans l'outil via :

— la commande décrite ici, après avoir préalablement nettoyé la géométrie des lignes à l'aide du plugin GRASS dans QGIS. Cette procédure est à privilégier car elle a l'avantage de faire des vérifications topologiques sur les données.

— l'outil Qgis en suivant ce tutoriel pour charger des tronçons dans la base de données PostGIS Geotrek à partir d'un réseau de sentiers. Il faut s'assurer en amont que les lignes à insérer sont topologiquement propres.

Si ils sont numérisés directement dans Geotrek-admin, il est possible d'afficher sur la carte un fichier GPX ou GeoJSON pour faciliter leur localisation.

Quand un nouveau tronçon intersecte un tronçon existant, ce dernier est découpé automatiquement à la nouvelle intersection.

En plus de leur géométrie, quelques informations peuvent être associées à chaque tronçon (nom, départ, arrivée, confort, source, enjeu d'entretien, usage et réseaux).

Comme pour les autres objets, les informations altimétriques sont calculées automatiquement grace au MNT présent dans la base de données.

Idem pour les intersections automatiques avec les zonages (communes, secteurs, zonages réglementaires) et les objets des autres modules qui sont intersectés automatiquement à chaque ajout ou modification d'un objet.

Comme pour tous les modules, il est possible d'exporter la liste des tronçons affichés (CSV, SHP ou GPX) ou bien la fiche complète d'un tronçon (ODT, DOC ou PDF).

Comme pour tous les modules, il est aussi possible d'attacher des documents à chaque tronçon depuis sa fiche détail (images, PDF, tableurs, ZIP…).

Enfin, toujours depuis la fiche détail d'un tronçon, il est possible d'en afficher l'historique des modifications.

4.2 Les sentiers

Il s'agit d'un ensemble linéaire composés d'un ou plusieurs tronçons (entiers ou partiels) grâce à la segmentation dynamique.

Les sentiers permettent d'avoir une vision de gestionnaire sur un linéaire plus complet que les tronçons (qui sont découpés à chaque intersection) pour en connaitre les statuts, la signalétique, les aménagements, les interventions ainsi que les itinéraires et POI. Il est d'ailleurs possible d'ajouter une intervention sur un sentier complet directement depuis la fiche détail d'un sentier.

Ils permettent également de préciser une ou plusieurs certifications ainsi que leur statut.

A ne pas confondre avec le module Itinéraires qui permet de créer des randonnées publiées sur un portail Geotrek-rando.

4.3 Les statuts

Ils permettent de renseigner des informations sur le linéaire (type physique, statut foncier, organismes ayant la compétence sentiers, gestionnaires des travaux et de la signalétique) sans avoir à le faire tronçon par tronçon grâce à la segmentation dynamique qui permet de localiser le départ et l'arrivée sur un ou plusieurs tronçons.

4.4 Les aménagements

Ils permettent d'inventorier les aménagements sur les sentiers (passerelles, mains courantes, cunettes, soutènements, bancs, parkings…) en les localisant, les typant, les décrivant, renseignant leur état et leur année d'implantation.

Les types d'aménagement sont découpés en 2 catégories (Ouvrages et Equipements). Ce découpage n'est utilisé que pour filtrer les aménagements.

Il est possible de créer une intervention directement depuis la fiche détail d'un aménagement.

Comme pour les autres modules, il sont intersectés avec les autres modules pour en connaitre l'altimétrie, les zonages (communes, réglementation…), les statuts (fonciers, physique, gestionnaire), les interventions, les itinéraires…

Il est aussi possible de les exporter, de leur attacher des fichiers (images, PDF, tableurs, ZIP…) et d'en consulter l'historique des modifications.

Pour importer automatiquement des éléments d'aménagements, se référer à la section *Import Infrastructure*

4.5 La signalétique

Centralise les informations sur la signalétique d'accueil, d'information et d'orientation des randonneurs. Permet de localiser, décrire l'implantation, et gérer les fichiers concernant la signalétique (BAT, maquettes, photos, etc.).

Pour une signalétique, il est possible de définir chaque lame et chaque ligne de lame. L'image ci-dessous montre un exemple de correspondance entre une signalétique terrain et sa saisie dans Geotrek.

Lames

Code	Couleur	Direction	État	Texte	Picto	Distance	Temps
30339/087/A	Orange	Sans Direction	Bon état	Prat Peyrot - Station			
30339/087/B1	Jaune	G	Bon état	Meyrueis	GR 6	17.0 km	4h25
				Camprieu		8.2 km	2h00
				L Espérou		3.9 km	1h10
30339/087/B2	Jaune	G	Bon état	Point Sublime		3.2 km	1h10
				Cascades de l Hérault		2.9 km	0h50
30339/087/B3	Jaune	G	Bon état	Col de La Caumette		2.0 km	0h40
				Col de la Serreyrède	GR7 - GR60 - GR66	1.8 km	0h30
30339/087/C	Jaune	D	Bon état	Mont Aigoual		2.8 km	1h00
				La Draillhe		1.2 km	0h25
				Parking	GR6 - GR7 - GR60	0.3 km	0h05
30339/087/D	Jaune	G	Bon état	Draille de Calcadis		0.7 km	0h15
30339/087/E	Jaune	G	A remplacer (dégradé, manquant)	Flèche gauche			
				Picto	Equestre		
30339/087/F	Jaune	G	A remplacer (dégradé, manquant)	Balise VTT	VTT		
				3	3 bleu		

Pour importer automatiquement des éléments de signalétiques, se référer à la section *Import Signage*

4.6 Les interventions

Les interventions permettent d'inventorier et suivre les travaux réalisés sur les sentiers. Chaque intervention correspond à une action sur un tronçon, sentier, aménagement ou signalétique.

Les interventions peuvent être localisées directement sur le linéaire de tronçon en les positionnant grâce à la segmentation dynamique. Ou bien ils peuvent correspondre à un sentier, un aménagement ou une signalétique en les créant depuis leur fiche détail.

Une intervention peut être souhaitée (demandée par un agent), planifiée (validée mais à réaliser) ou réalisée.

Un enjeu peut être renseigné pour chaque intervention. Il est calculé automatiquement si un enjeu a été renseigné au niveau du tronçon auquel l'intervention se raccroche.

Chaque intervention correspond à un type. On peut aussi renseigner si celle-ci est sous-traitée, les désordres qui en sont la cause, la largeur et la hauteur. La longueur est calculée automatiquement si il s'agit d'une intervention linéaire mais est saisie si il s'agit d'une intervention ponctuelle.

Plusieurs interventions peuvent être rattachées à un même chantier pour avoir une vision globale de plusieurs interventions correspondant à une opération commune.

L'onglet *Avancé* du formulaire permet de renseigner des informations financières sur chaque intervention (coût direct et indirect lié au nombre de jours/agents dissocié par fonction).

4.7 Les chantiers

Les chantiers permettent de grouper plusieurs interventions pour en avoir une vision globale et d'y renseigner globalement des informations administratives (contraintes, financeurs, prestataires, cout global, maitrise d'ouvrage…) et éventuellement d'y attacher des documents (cahier des charges, recette, plans…).

Leur géométrie est la somme des géométries des interventions qui les composent.

CHAPITRE 5

Modules de valorisation

- *Itinéraires*
  - *Fiche détaillée*
  - *Itinérance*
  - *Points de référence*
- *Points d'intérêts (POI)*
- *Services*
- *Contenus touristiques*
  - *Zone tampon pour contenus à proximité*
- *Évènements touristiques*
- *Signalements*
- *Zones sensibles*
- *Activités outdoor*
  - *Sites et parcours*
  - *Nomenclatures*
  - *Filières*

5.1 Itinéraires

Les itinéraires sont des randonnées présentés au grand public : l'itinéraire est défini selon la géométrie des tronçons empruntés. L'ajout ou la modification sont fréquents.

5.1.1 Fiche détaillée

Basique

Structure liée ~ requis
- Description : nom de la structure d'appartenance de l'itinéraire
- Type : liste déroulante
- Choix : unique
- URl de configuration : /admin/authent/structure/
- Visibilité : interne
- Exemple : CD09

Nom [fr] ~ requis
- Description : nom de l'itinéraire
- Type : champ libre
- Multilingue : oui
- Visibilité : publique
- Exemple : GR09 Boucle du Pic des Trois Gentilhommes

En attente de publication
- Description : itinéraire en attente d'être publié
- Type : case à cocher
- Valeur par défaut (décoché) : itinéraire publiable
- Visibilité : publique

Publié [fr]
- Description : Itinéraire publié ou en brouillon
- Type : case à cocher
- Valeur par défaut (décoché) : brouillon
- Visibilité : interne

Départ [fr] ~ Recommandé
- Description : description du lieu de départ
- Type : champ libre
- Visibilité : publique
- Exemple : Refuge de les Caussis

Arrivée [fr] ~ Recommandé
- Description : description du lieu de l'arrivée
- Type : champ libre
- Visibilité : publique
- Exemple : Refuge de les Caussis

Durée ~ Recommandé
- Description : durée de l'itinéraire (en heures (1.5 = 1 h 30, 24 = 1 jour, 48 = 2 jours))
- Type : numérique
- Visibilité : publique
- Exemple : 1.5

Difficulté ~ Recommandé
- Description : niveau de difficulté de l'itinéraire
- Type : liste déroulante
- Choix : unique
- Valeurs de champ paramétrables dans l'outil d'administration : oui

- Chemin d'accès dans l'outil d'administration : /admin/trekking/difficultylevel/
- Visibilité : publique
- Exemple : Intermédiaire

Pratique ~ Recommandé
- Description : type de pratique de l'itinéraire
- Type : liste déroulante
- Choix : unique
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/trekking/practice/
- Visibilité : publique
- Exemple : Pédestre

Échelle de cotation
- Description : définition d'une cotation de l'itinéraire spécifique à la pratique
- Type : liste déroulante
- Choix : unique
- Conditionnel : selon la pratique choisie
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/trekking/ratingscale/
- Visibilité : publique
- Exemple : Technicité : 3 - Moyen

Description de cotation [fr]
- Description : précision sur la valeur de cotation de l'itinéraire spécifique à la pratique
- Type : champ libre
- Visibilité : publique
- Exemple : La technicité de cet itinéraire est moyenne car elle ne comprend pas de passages délicats.

Parcours
- Description : type de parcours
- Type : liste déroulante
- Choix : unique
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/trekking/route/
- Visibilité : publique
- Exemple : Boucle

Accès routier [fr]
- Description : accès routier jusqu'au point de départ
- Type : champ libre
- Visibilité : publique
- Exemple : Depuis Savines-Le-Lac (17km), prendre la D41 jusqu'à Réallon. Suivre ensuite la D241 jusqu'au hameau des Gourniers au fond de la vallée.

Chapeau [fr] ~ Recommandé
- Description : bref résumé de l'itinéraire avec accroche
- Type : champ libre
- Visibilité : publique
- Exemple : Une agréable randonnée familiale en boucle avec un beau point de vue sur la vallée de Réallon.

Ambiance [fr]
- Description : attractions principales et intérêts
- Type : champ libre
- Visibilité : publique
- Exemple : La montée commence dans la fraîcheur d'un bois de hêtre puis d'une belle forêt de mélèzes avant d'arriver à d'anciens près de fauche, témoignage des activités passées. Les ruines d'anciens chalets d'alpage rappellent ce qu'était la vie en montagne. Quand le sentier passe en balcon le paysage s'ouvre en un large point de vue sur la vallée de Réallon.

Description [fr]
- Description : description technique pas à pas de l'itinéraire (liste numérotée conseillée)

- Type : champ libre
- Visibilité : publique
- Exemple : Du parking, traverser le pont, au carrefour du hameau prendre la direction de Chargès, remonter la rue jusqu'à la dernière maison.

1. Prendre le sentier à droite direction l'Oussella
2. Après la marmite de Géant et le pont, continuer à gauche direction l'Oussella.

Avancé

Parking conseillé [fr]
- Description : nom du lieu recommandé pour se garer en voiture
- Type : champ libre
- Visibilité : publique
- Exemple : Parking du refuge de les Caussis.

Transport en commun [fr]
- Description : indications du ou des transports en commun pour se rendre au départ
- Type : champ libre
- Visibilité : publique
- Exemple : Ce GR est accessible en train, il démarre de la gare SNCF de Boussenac (ligne Seix - Boussenac).

Recommandations [fr]
- Description : recommandations sur les risques, danger ou meilleure période pour pratiquer l'itinéraire
- Type : champ libre
- Visibilité : publique
- Exemple : Attention en cas d'orage. Fortement déconseillé par mauvais temps !

Matériel [fr]
- Description : matériel nécessaire ou conseillé
- Type : champ libre
- Visibilité : publique
- Exemple : Chaussures de randonnées

Thèmes
- Description : thématiques principales de l'itinéraire
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/common/theme/
- Visibilité : publique
- Exemple : Lacs et glaciers, Géologie, Point de vue

Étiquettes
- Description : éléments de recommandation ou utiles à connaître
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/common/label/
- Visibilité : publique
- Exemple : Chien autorisé

Réseaux
- Description : nom du réseau de balisage de l'itinéraire
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/core/network/
- Visibilité : publique

- Exemple : GR

Liens web
- Description : liens web apportant des compléments d'informations utiles
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/trekking/weblink/
- Visibilité : publique
- Exemple : Consulter la météo locale de Boussenac

Lieux de renseignement
- Description : lieux de renseignements utiles
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/tourism/informationdesk/
- Visibilité : publique
- Exemple : Office de tourisme de Seix, Office du tourisme de Boussenac

Source
- Description : nom de l'organisme auteur de l'itinéraire
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/common/recordsource/
- Visibilité : publique
- Exemple : Conseil départemental de l'Ariège

Portail
- Description : site web grand public sur lequel sera publié l'itinéraire
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/common/targetportal/
- Visibilité : publique
- Exemple : CD09

Enfants
- Description : ensemble des itinéraires étapes constituant l'itinérance
- Type : liste déroulante
- Choix : multiple
- Visibilité : publique
- Exemple : Etape GR09 Refuge les Caussis-Étang Rond, Etape GR09 Étang Rond-Refuge les Caussis.

ID externe
- Description : identifiant de l'itinéraire dans sa base de données source (dans le cas d'un import)
- Type : numérique
- Visibilité : interne
- Exemple : 15715

Deuxième id externe
- Description : identifiant secondaire de l'itinéraire dans sa base de données source (dans le cas d'un import)
- Type : numérique
- Visibilité : interne
- Exemple : 15716

Système de réservation
- Description : nom du système de réservation
- Type : liste déroulante
- Choix : unique
- Valeurs de champ paramétrables dans l'outil d'administration : oui

- Chemin d'accès dans l'outil d'administration : /admin/common/reservationsystem/
- Visibilité : publique
- Exemple : Open system

ID de réservation
- Description : identifiant de l'itinéraire dans son système de réservation
- Type : numérique
- Visibilité : interne
- Exemple : 157187456

POI exclus
- Description : liste des points d'intérêt associés à l'itinéraire à ne pas faire remonter sur celui-ci
- Type : liste déroulante
- Choix : multiple
- Visibilité : interne
- Exemple : les Estagnous

Accessibilité

Type d'accessibilité
- Description : type d'accessibilité
- Type : liste déroulante
- Choix : multiple
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/trekking/accessibility/
- Visibilité : publique
- Exemple : Fauteuil roulant, poussette

Niveau d'accessibilité
- Description : niveau d'accessibilité
- Type : liste déroulante
- Choix : unique
- Valeurs de champ paramétrables dans l'outil d'administration : oui
- Chemin d'accès dans l'outil d'administration : /admin/trekking/accessibilitylevel/
- Visibilité : publique
- Exemple : Débutant

Aménagements d'accessibilité [fr]
- Description : infrastructure d'accessibilité spécifique à disposition
- Type : champ libre
- Visibilité : publique
- Exemple : Rampes d'accès amovibles.

Pente accessibilité [fr]
- Description : description de la pente : supérieure à 10 % (Nécessite une assistance quand la pente est supérieure à 8%)
- Type : champ libre
- Visibilité : publique
- Exemple : Pente supérieure à 12%.

Revêtement accessibilité [fr]
- Description : description des revêtements rencontrés sur la totalité d'un itinéraire
- Type : liste déroulante
- Visibilité : publique
- Exemple : Piste ensablée à partir des Estagnous.

Exposition accessibilité [fr]
- Description : description des expositions et des zones ombragées
- Type : champ libre
- Visibilité : publique

— Exemple : Piste ombragée.

Largeur accessibilité [fr]

— Description : description des rétrécissements des sentiers et la largueur minimum
— Type : champ libre
— Visibilité : publique
— Exemple : Sentier étroit demandant une forte technique de conduite, avec un passage d'un mètre de large. Passage resserré sur le pont traversant la rivière.

Conseil d'accessibilité [fr]

— Description : éléments particuliers permettant d'apprécier le contexte de l'itinéraire pour les PMR (conseils, passages délicats, etc.)
— Type : liste déroulante
— Visibilité : publique
— Exemple : La montée du Saut du Laire, particulièrement les 150 derniers mètres, sont déconseillés aux joëlettes, notamment dans la perspective du retour en descente sur pavé pouvant être glissant.

Signalétique accessiiblité [fr]

— Description : description de taille, forme et couleurs des signalétiques d'accessibilité
— Type : liste déroulante
— Visibilité : publique
— Exemple : Panneau de signalisation PMR rampe d'accès amovible.

5.1.2 Itinérance

Il est possible de créer des randonnées itinérantes (sur plusieurs jours) et d'y associer des étapes comme sur cet exemple : https://www.grand-tour-ecrins.fr/trek/937571-GR%C2%AE54—Tour-de-l-Oisans-et-des-Ecrins.

Pour cela il faut créer un itinéraire parent (séjour itinérant complet) puis y associer des itinéraires enfants (étapes) de manière ordonnée, dans le champs *Enfants* présent dans l'onglet *Avancé* du formulaire itinéraire du séjour complet.

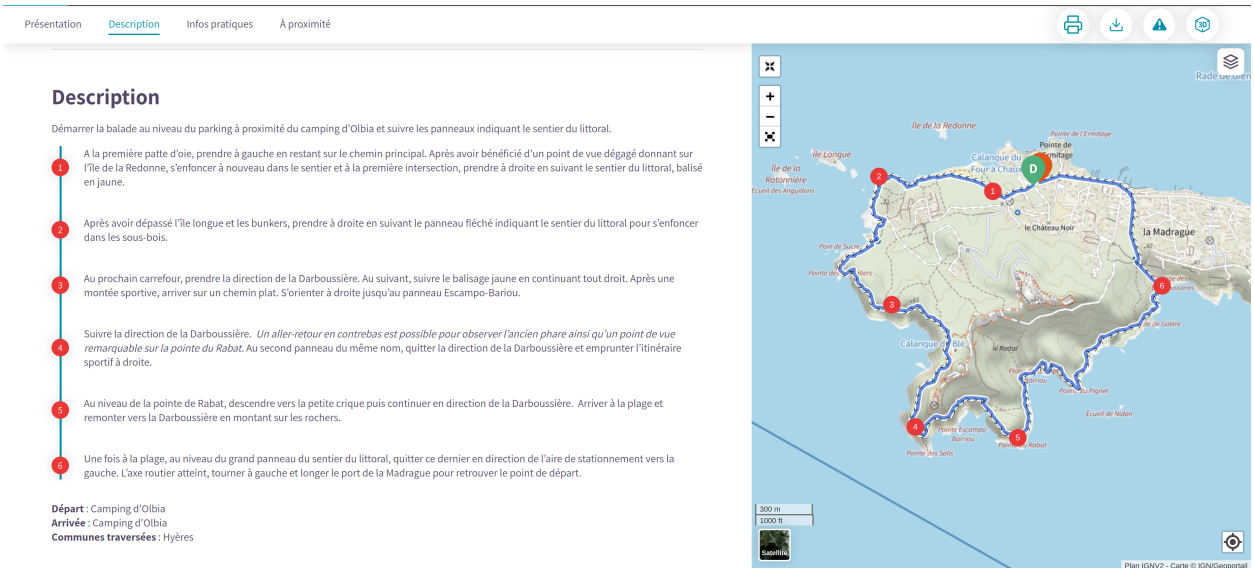
Le séjour complet ainsi que chaque étape sont donc chacunes des randonnées comme les autres. La seule différence est que les étapes (itinéraires enfants) sont rattachées à l'itinéraire parent.

Si vous ne souhaitez pas que les étapes soient affichées dans la page de Recherche de Geotrek-rando, il ne faut pas les publier. Il suffit alors de publier l'itinéraire parent, pour que toutes les étapes qui y sont rattachées apparaissent uniquement dans sa fiche détail de Geotrek-rando.

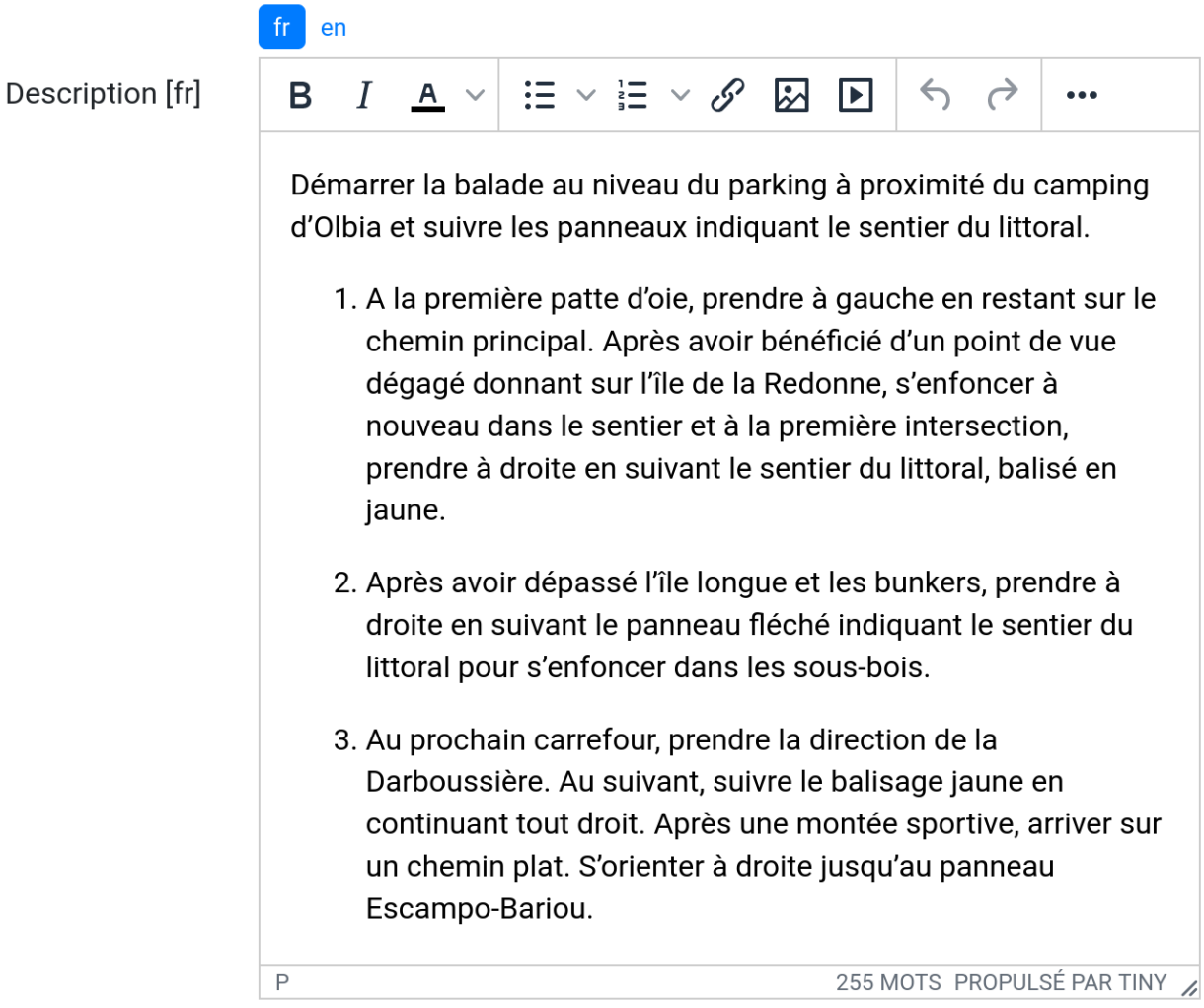
5.1.3 Points de référence

Lorsqu'on localise un itinéraire, il est aussi possible de localiser le parking de la randonnée et de placer des points de référence numérotés sous forme de puces rouges sur la carte.

Ces derniers servent à y faire référence dans le champs Description de l'itinéraire (Pas à pas) :



Pour que des puces numérotées sous forme de pastilles rouges soient affichées dans la description, il suffit de les saisir en tant que Liste numérotées dans le champs Description :



5.2 Points d'intérêts (POI)

Les POIs ne sont pas associés aux itinéraires par zone tampon, mais par segmentation dynamique, en fonction des tronçons communs de l'itinéraire et des POIs.

Pour importer automatiquement des éléments de POIs, se référer à la section *Import POIs*

Note : Lorsque Geotrek est installé en mode « sans segmentation dynamique », les POI sont rattachés aux itinéraires par zone tampon (buffer). Dans ce cas le paramètre pris en compte est le paramètre *TREK_POI_INTERSECTION_MARGIN* qui est paramétrable dans le custom.py.

5.3 Services

5.4 Contenus touristiques

5.4.1 Zone tampon pour contenus à proximité

La taille de la zone tampon est fixée à 500m autour de l'itinéraire pour remonter les informations des contenus/évènements touristiques et services.

Pour modifier la distance de la zone tampon (buffer), se référer à la section *Distances*

5.5 Évènements touristiques

5.6 Signalements

Pour configurer Suricate, se référer à cette section *Suricate support*

5.7 Zones sensibles

Ce module permet d'inventorier, de localiser des zonages réglementaires (de type sites Natura 2000, Arrêtés Préfectoraux de Protection de Biotope (APPB), sites classés, sites d'intérêt écologique) ou des surfaces liées à la présence d'une ou plusieurs espèces protégées ou patrimoniales.

La saisie peut se faire sous forme de polygone ou de cercle pour les zones de sensibilité. Le diamètre du cercle est généré automatiquement selon la zone ou l'espèce, avec une valeur définie préalablement.

Si une randonnée ou un site outdoor intersecte une ou plusieurs zones zones de sensibilité, alors celles-ci sont affichées à la suite des recommandations dans Geotrek-Rando.

Il est également possible d'importer directement les zonages depuis un format Shapefile (SHP).

Pour activer le module Zones sensibles, se référer à cette section *Sensitive areas*

5.8 Activités outdoor

Par défaut, ce module est désactivé dans Geotrek. Pour activer le module Outdoor, se référer à cette section *Outdoor*

5.8.1 Sites et parcours

Geotrek-admin dédie deux modules aux activités outdoor : les sites et les parcours. Un site correspond à une zone ou à un réseau hydrographique à gérer et/ou à valoriser d'un seul tenant : site d'escalade, cours d'eau, zone de vol libre…

Les sites peuvent être subdivisés en sous-sites (dits enfants), qui peuvent eux-mêmes être subdivisés en sous-sites : secteurs d'escalade (groupe de blocs ou falaise), aires de décollage ou d'atterrissage…

Chaque site (ou sous-site) peut contenir différents parcours : voie d'escalade, parcours d'eau vive…

Il est possible de regrouper un enchaînement de parcours sous forme d'un parcours particulier appelé itinérance : grande voie d'escalade, enchainement entre différentes pratiques…

Les deux modules outdoor permettent de :

- **Lister, filtrer, créer, modifier et exporter des sites et des parcours outdoor** de manière générique pour gérer tout type de pratiques (escalade, alpinisme, via ferrata, canyoning, kayak, vol libre, plongée…)
- **Ajouter les pratiques** que l'on souhaite de manière générique, et de définir leurs propres types et leurs propres niveaux et valeurs de cotation
- **Lier des sites entre eux** de manière hiérarchisée, pour avoir des sites, des sous-sites, des sous-sous-sites… Par exemple pour un site global avec différentes pratiques, pour un site d'escalade avec des secteurs, des sous-secteurs…
- **Agréger des informations au niveau d'un site**, en fonction des sous-sites qui lui sont rattachés. Par exemple les pratiques d'un grand site outdoor sont l'agrégation des pratiques des sous-sites qui le composent
- **Lier des parcours à des sites** et leur appliquer une cotation en fonction de la pratique du site auquel ils sont rattachés
- **Lier des parcours entre eux pour faire de l'itinérance** avec une fiche mère et des fiches enfants (étapes), comme c'est déjà la cas pour les itinéraires
- **Associer des aménagements aux sites** (parking, toilettes, banc…) automatiquement par intersection géographique
- **Associer des interventions à des sites ou parcours**, automatiquement par intersection géographique, ou explicitement en passant par le site ou le parcours sur lequel l'intervention a été réalisée
- **Associer des POI à un site ou parcours**, automatiquement par intersection géographique
- **Associer des accès au site**, automatiquement par intersection géographique des itinéraires à proximité

Pour activer le module Outdoor, se référer à la section *Outdoor*

Arborescence

Les fiches détail des sites et des parcours présentent les liens entre eux sous forme d'une arborescence. Pour ne pas surcharger, tous les éléments ne sont pas repris, mais uniquement :

- le site/parcours courant,
- le site auquel il appartient (parent), ainsi que le grand-parent, etc. jusqu'à remonter au plus haut niveau,
- Les sites et parcours « frères », ayant le même site parent.
- les différents sous-sites et/ou parcours enfants.

Des liens permettent d'ajouter des sites ou parcours en les positionnant directement dans l'arborescence.

5.8.2 Nomenclatures

Via l'interface de configuration de Geotrek-Admin, il est possible de modifier les nomenclatures.

- **Filières** : elles servent à regrouper les pratiques pour pouvoir filtrer rapidement les sites ou parcours. Par exemple la filière « eau vive » peut regrouper « kayak » et « canyoning ».
- **Pratiques** : les pratiques sportives. Vous pouvez préciser à quelle filière elle appartient.
- **Types de sites** : ces catégories permettent d'étiqueter et de filtrer les sites. Elles sont spécifiques à chaque pratique. Par exemple « Site école » pour l'escalade.
- **Échelle de cotation** : permet de regrouper les cotations faisant partie de la même échelle. Elles sont spécifiques à chaque pratique.

5.8.3 Filières

Escalade

La notion de site est naturelle. Elle peut être définie géographiquement par un polygone. Il est possible (mais pas obligatoire) de créer des sous-sites pour représenter des secteurs. Ou pour des falaises, elles-mêmes divisées en sous-sous sites pour les différents secteurs.

Chaque voie correspond à un parcours. La voie étant verticale et la carte horizontale, le plus pertinent est de définir géographiquement la voie par un simple point. Une grande voie peut être décrite simplement par un parcours mais, pour plus de détails, il est également possible de créer autant de parcours que de longueurs dans la grande voie. Lors de la saisie de la grande voie, il faudra préciser les différentes longueurs dans le champ « Enfants », dans le bon ordre. Le nom de chaque longueur pourra reprendre le nom de la voie suffixé par « longueur 1 », « longueur 2 », etc.

Vol libre

La zone de vol n'est pas définie géographiquement de manière précise mais fait quand même l'objet d'un site avec un nom (ex : « massif de … ») et un polygone approximatif ou bien un point (de préférence celui de départ). Cela n'a pas une importance déterminante. Ce qui compte c'est 1) de rendre cela lisible sur une carte et 2) d'être cohérent entre les différents sites.

Pour chaque zone de vol, les différentes aires de décollage et d'atterrissage sont définies à l'aide de sous-sites. Afin de les identifier, il faut créer les types de site « Aire de décollage » et « Aire d'atterrissage » pour la catégorie « Vol libre » dans la nomenclature et associer ces types aux aires. Comme le vol est libre, il n'est pas nécessaire de définir des parcours. Cependant, il est possible d'en définir pour donner des exemples de trajectoires.

Eau vive

Le site est généralement constitué par une rivière ou une portion de rivière. Il est possible d'y adjoindre des affluents. La géométrie du site est donc un linéraire correspondant à un réseau hydrographique.

Les aires d'embarquement/débarquement sont définies par des sous-sites. Leur géométrie peut être définie sous forme d'un point ou d'un polygone. Les parcours sont automatiquement attachés à une aire d'embarquement et une aire de débarquement qui sont les aires les plus proches respectivement du début et de la fin du parcours.

CHAPITRE 6

Edition d'un objet

— *Les calculs géométriques sur les objets*
  — *Informations 3D*
  — *Calcul du dénivelé*
  — *Lien entre les POIs et les itinéraires*
  — *Segmentation dynamique*
  — *Snapping - Aimantage - Accrochage*
  — *Informations altimétriques*
— *Attributs aditionnels sur les objets*
  — *Fichiers liés*
  — *Photos accessibilité*
  — *Historique*

6.1 Les calculs géométriques sur les objets

6.1.1 Informations 3D

Toutes les géométries des tronçons sont *drapées* sur un modèle numérique de terrain, lors de leur création ou mise à jour.

Tous les objets linéaires reposant sur les tronçons récupèrent leurs informations 3D à partir des tronçons associés, au lieu d'utiliser le modèle numérique de terrain.

6.1.2 Calcul du dénivelé

Nous calculons le dénivelé positif cumulé.

Nous échantillonnons le DEM tous les 25m par défaut (voir paramètre `ALTIMETRIC_PROFILE_PRECISION`), et nous ajoutons un peu de lissage pour éviter le bruit.

Pour plus de détails :
- The smoothing explanation
- The piece of code to compute positive and negative accumulations
- Draping lines with PostGIS (or french version)

6.1.3 Lien entre les POIs et les itinéraires

Les POIs sont considérés comme des informations *éditoriales* et sont créés le long des randonnées.

Lorsqu'un POI est créé, il est rattaché à l'itinéraire le plus proche.

Un itinéraire repose sur un ou plusieurs tronçons, et certains POIs leur sont associés.

Note : Il n'existe actuellement aucun moyen de contrôler manuellement l'association entre randonnées et les POIs.

Si le paramètre `TREKKING_TOPOLOGY_ENABLED` est défini sur `False`, aucun tronçon n'est impliqué. Les POIs sont ainsi liés à des treks utilisant une intersection spatiale de base, avec une distance définie dans le paramètre `TREK_POI_INTERSECTION_MARGIN` (par défaut à 500 mètres).

6.1.4 Segmentation dynamique

Certains objets sont saisis et stockés relativement aux tronçons, en utilisant la segmentation dynamique. Il s'agit des objets suivants : sentiers, statuts, aménagements, interventions, itinéraires et POI. Tous les autres objets sont indépendants et ont leur propre géométrie.

La **segmentation dynamique** permet de saisir et stocker des informations relatives aux tronçons. Elle s'accompagne d'un ensemble de déclencheurs calculant automatiquement les géométries résultantes à la création et lors de modifications sur le réseau de tronçons.

Cela permet de garantir une **cohérence géométrique et topologique entre les tronçons et les objets liés** (travaux, aménagements, statuts, itinéraires…), mais aussi d'associer tous les objets entre eux en vérifiant leur superposition sur le réseau.

C'est pourquoi, modifier un tronçon peut entrainer des modifications des objets qui lui sont rattachés (interventions, itinéraires, POIs…). Supprimer un tronçon, supprime les objets qui lui sont rattachés par segmentation dynamique.

Les éléments ponctuels et linéaires des différents modules sont stockés sous forme d'évènements (PKdebut, PKfin et décalage dans la table `geotrek.core_topology`) liés à un ou plusieurs tronçons (`geotrek.core_pathaggregation`).

Un objet peut ainsi être associé à un ou plusieurs tronçons, partiellement ou entièrement.

Les objets ponctuels ne sont associés qu'à un seul tronçon, sauf dans le cas où ils sont positionnés à une intersection de tronçons.

Chaque évènement dispose néanmoins d'une géométrie calculée à partir de leur segmentation dynamique pour faciliter leur affichage dans Geotrek ou dans QGIS (voir la section *Visualiser les données dans Qgis*). Il ne faut néanmoins pas modifier directement ces géométries, elles sont calculées automatiquement quand on modifie l'évènement d'un objet.

6.1.5 Snapping - Aimantage - Accrochage

Quand vous créez un objet, il est possible de le snapper (aimanter) aux objets existants. C'est notamment utile pour bien raccorder les tronçons entre eux. Quand vous raccrochez un tronçon à un tronçon existant, ce dernier est coupé automatiquement à la nouvelle intersection.

Les fonctions d'aimantage ne sont pas disponibles lors de la création d'un nouvel objet (linéraire ou ponctuel). Il faut commencer par le créer puis le modifier pour disposer des fonctionnalités d'aimantage, activées automatiquement lorsque l'on se rapproche d'un objet existant. Par défaut la distance d'imantage est de 30 pixels mais elle est modifiable en configuration avancée.

6.1.6 Informations altimétriques

L'édition des objets s'effectue en 2D, et leur géométrie est "drapée" automatiquement en 3D sur un Modèle Numérique de Terrain (MNT) stocké dans la base de données au format raster.

Cette information est visible à travers le profil altimétrique pour les tronçons, les itinéraires et les autres objets linéaires.

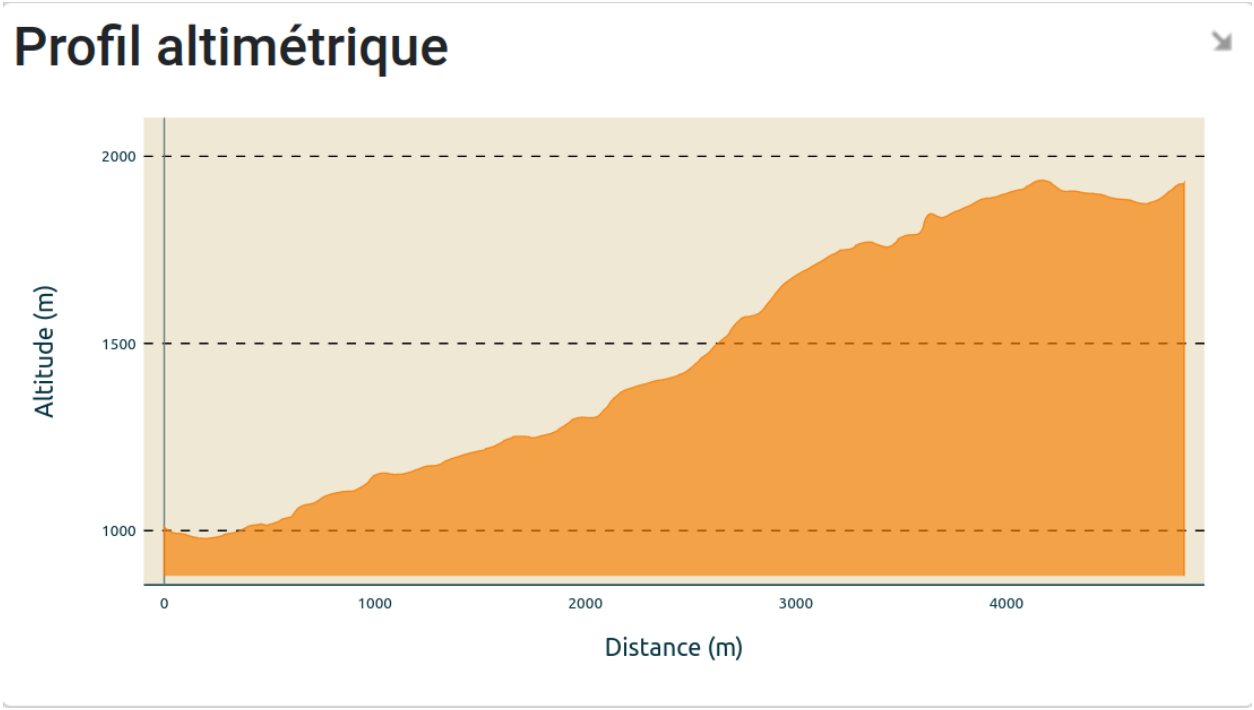


Fig. 1 – Profil altimétrique d'un itinéraire

Pour importer automatiquement un MNT, se référer à la section *Import DEM (altimetry)*

6.2 Attributs aditionnels sur les objets

6.2.1 Fichiers liés

Fichiers liés

L'ajout, la modification ou la suppression des documents, illustrations et photos s'effectuent depuis l'onglet « Fichiers liés » de la fiche détail.

Pour chaque fichier lié, l'auteur, le titre, la légende et sa catégorie sont saisis. Les fichiers liés peuvent être de tout type (photo, vidéo, dessin, PDF, tableur, fichier audio…). Pour les images, un aperçu est présenté.

Les vignettes et versions redimensionnées des photos sont créées automatiquement lors de l'ajout. Les contenus saisis sont publiés automatiquement.

Il est possible de limiter la gestion des fichiers liés à un groupe restreint d'utilisateurs.

Représentation verticale : les vues HD

Pour aller au-delà de la localisation sur une carte dans la représentation des sites d'activité Outdoor, notamment celles verticales (escalade, via-ferrata, alpinisme…), nous avons la possibilité d'ajouter des photos très haute définition (gigapixel) sur les itinéraires, POI et sites outdoor, et d'annoter celles-ci pour les enrichir.

Le bloc « Vues HD » dans l'onglet « Fichier liés » permet d'associer une photo très haute définition (de plusieurs dizaines ou centaines de Mo) aux itinéraires, POI et sites Outdoor. Ces images sont tuilées automatiquement pour disposer de fichiers plus légers à charger dans un navigateur (en fonctionnant comme les fonds de carte tuilés).

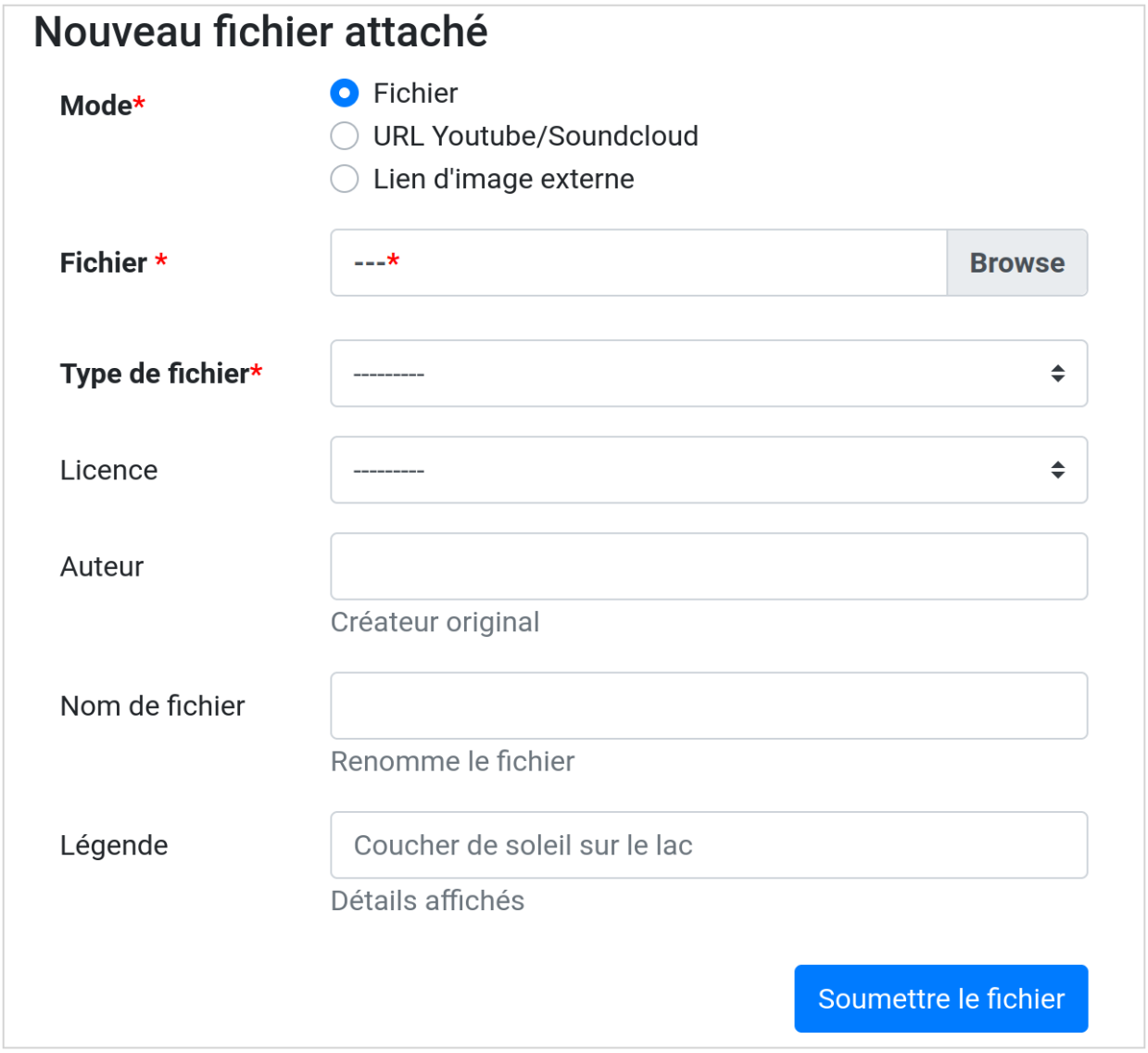
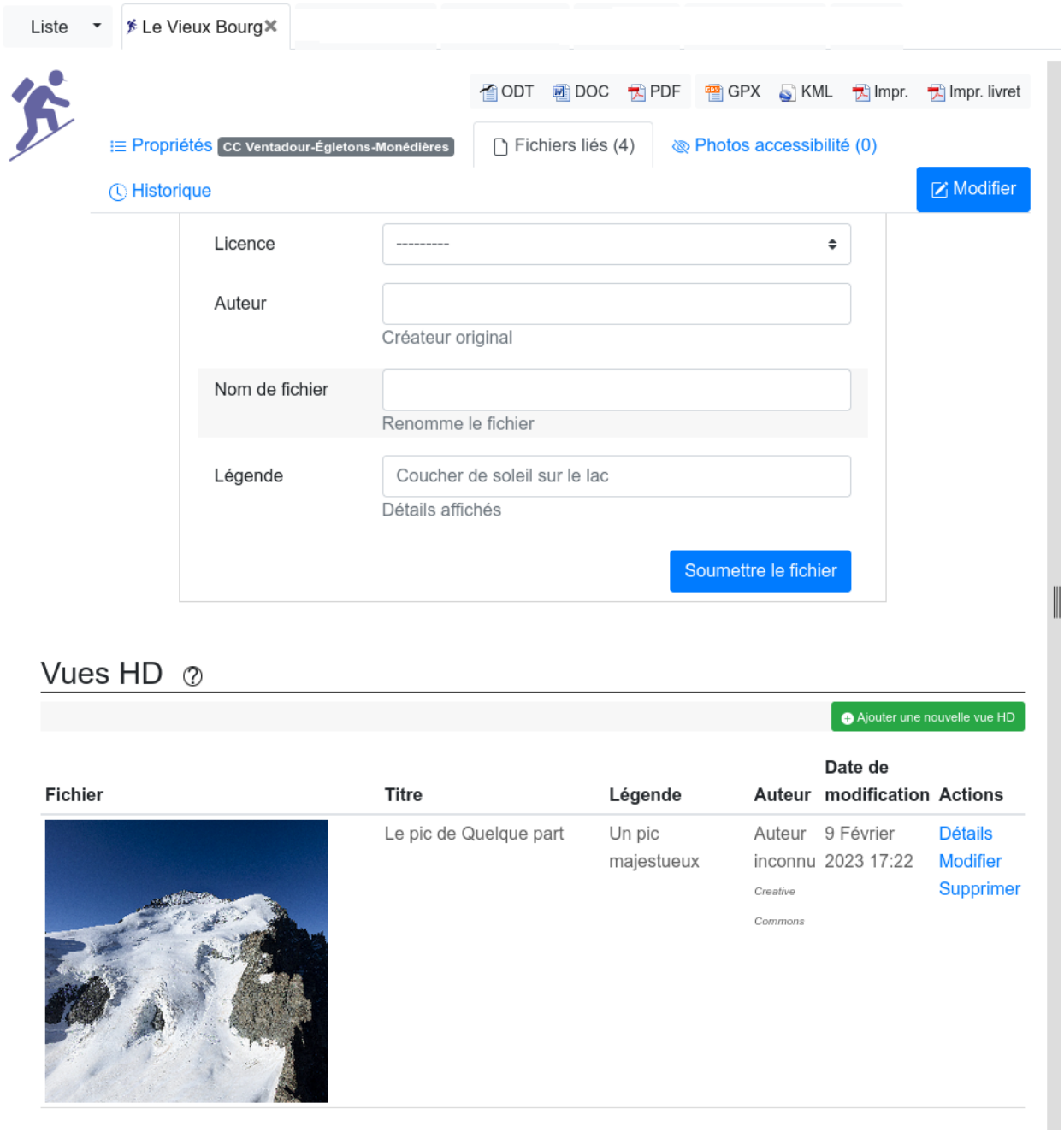
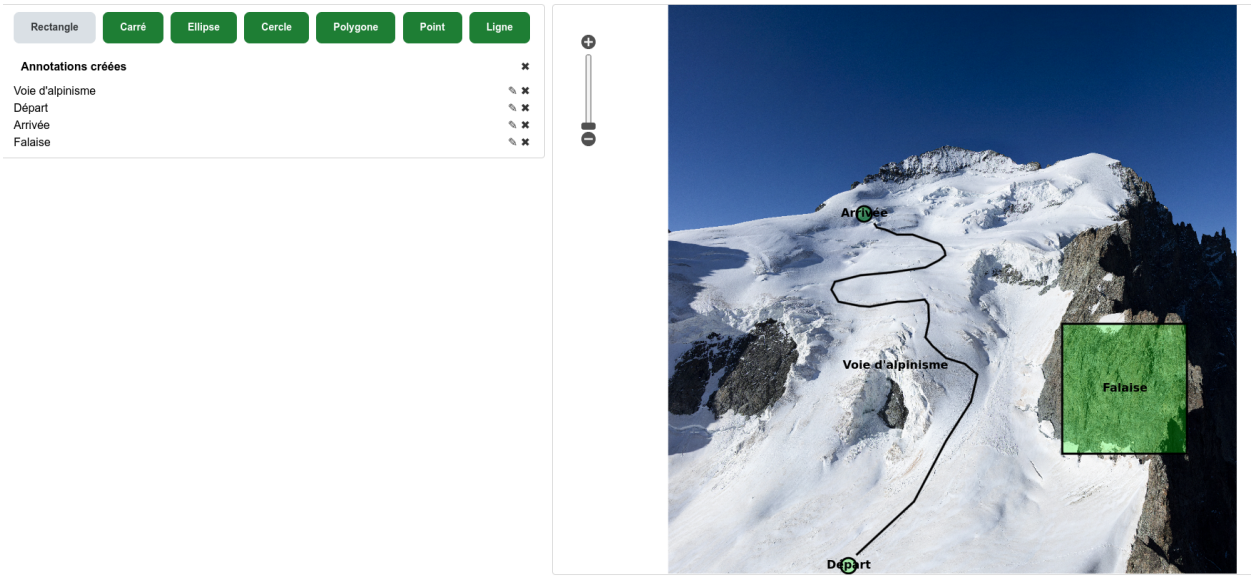


FIG. 2 – Fenêtre d'ajout de fichiers

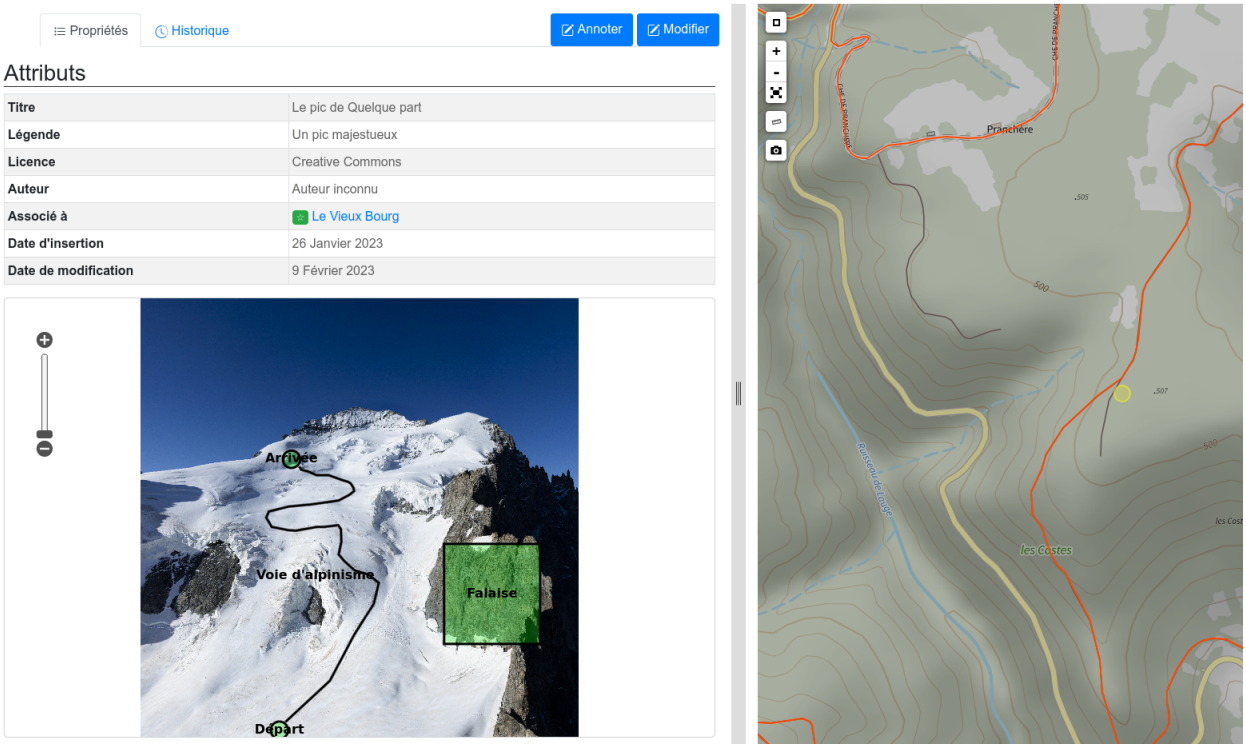


Une fois l'image ajoutée, un formulaire d'annotation permet d'ajouter des objets (points, lignes, polygones, cercles…) et des textes pour enrichir les photos.



Les annotations sont stockées en GeoJSON et peuvent donc être affichées par dessus la photo tuilée dans une librairie javascript de cartographie (comme Leaflet ou GeoJS) au niveau de Geotrek-rando-v3 ou autre. Pour cela, l'APIv2 expose pour chaque Vue HD l'adresse de récupération des tuiles ainsi que les annotations GeoJSON.

La vue HD est également associée à une localisation correspondant à l'emplacement de ce que l'on voit sur l'image.



6.2.2 Photos accessibilité

Cet onglet permet d'ajouter des photos afin d'illustrer les informations liées à l'accessibilité des randonnées.

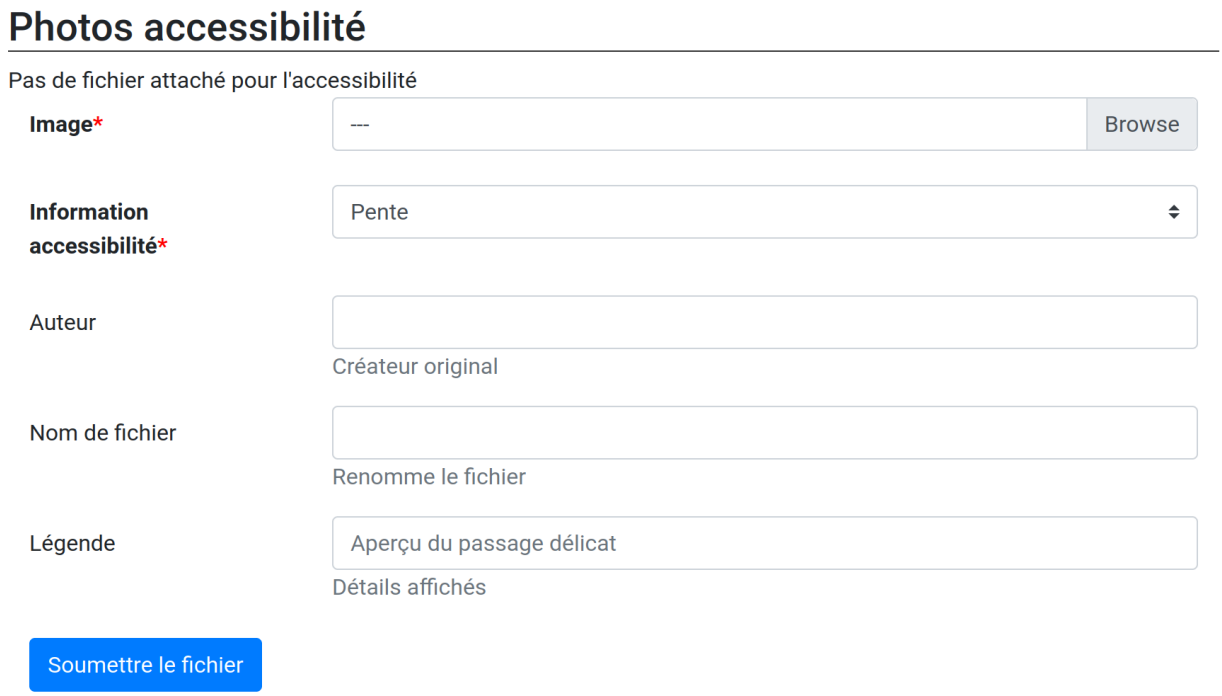


FIG. 3 – Fenêtre d'ajout de photos liées à l'accessibilité

6.2.3 Historique

Tous les objets créés dans Geotrek-admin possèdent un historique donnant des renseignement sur la date de l'évènement, l'utilisateur à l'origine de l'évènement et l'action réalisée (modification, ajout de fichier, etc.)

Date	Utilisateur	Action
11 Janvier 2024 14:54 (2 semaines)	bpo	Modifié
9 Novembre 2021 14:28 (2 ans)	ehe	Modifié
20 Octobre 2020 11:47 (3 ans)	la5c	Modifié
3 Juin 2019 14:38 (4 ans)	gutard	Modifié - Favoriser le fichier
12 Avril 2019 15:37 (4 ans)	ehe	Modifié - Ajout du fichier
12 Avril 2019 15:35 (4 ans)	ehe	Modifié
12 Avril 2019 15:31 (4 ans)	ehe	Modifié
19 Juin 2018 18:13 (5 ans)	gutard	Modifié
20 Février 2017 10:31 (6 ans)	gutard	Modifié
30 décembre 2014 17:42 (9 ans)	gutard	Modifié

Historique complet

FIG. 4 – Historique des modifications sur un objet

CHAPITRE 7

Paramétrage

- *Accès interface de configuration*
- *Paramétrages des modules*
  - *Itinéraires*
  - *Tourisme*
- *Gestion des utilisateurs*
  - *Utilisateurs et permissions*
  - *Groupes*
  - *Structures*
- *Gestion multilingue*
- *Configuration des portails*
- *Intégration des données SIT*
- *Intégration des fonds de cartes*
- *Intégration de pictogrammes*

7.1 Accès interface de configuration

Menu à droite > admin

7.2 Paramétrages des modules

Cette section présente quelques exemples de paramétrages des catégories au sein des modules.

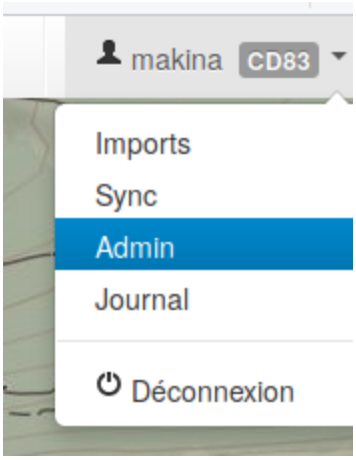


FIG. 1 – Lien vers le module de configuration

7.2.1 Itinéraires

Exemple : ajouter une pratique

— Dans le module de configuration, à la ligne “Pratiques” cliquer sur “+ ajouter”
— Remplir les champs (en gras les champs obligatoires)

Note : La couleur n’est utilisée que pour le mobile actuellement.

Exemple : ajouter une étiquette

Les étiquettes sont des encarts « pré-configurés » pouvant être réutilisés sur de multiples itinéraires. Elles présentent plusieurs avantages : - ne pas avoir à saisir à chaque itinéraire les même informations - permet de filtrer les itinéraires dans la vue liste (catégorie « Autres ») sur Geotrek-Rando.

Pour les configurer, vous devez : - vous rendre dans le module de configuration - dans la section Étiquettes du groupe **COMMUN** cliquer sur + *Ajouter*

Via cette interface vous pourrez créer des étiquettes puis, une fois créées, les rattacher à des itinéraires.
— Créer une étiquette :
Pour mettre en forme le contenu de l’étiquette, il est possible d’utiliser du `HTML`. Pour cela, il est recommandé d’utiliser des outils permettant de formater du contenu et d’obtenir le résultat en HTML directement. Par exemple via l’outil libre Summernote.
— Associer une étiquette à un itinéraire :
Une fois l’étiquette créée il faut l’associer à un itinéraire pour qu’elle soit visible sur le site. Une fois dans votre instance Geotrek Admin, éditez l’itinéraire concerné. Cliquez ensuite sur l’onglet *Avancé* et dans le champ Étiquettes choisissez dans le menu déroulant l’étiquette de votre choix (si vous en avez défini plusieurs).

Astuce :
— L’ajout d’un pictogramme est facultatif, par défaut le pictogramme de l’étiquette sera le même que celui des recommandations dans les « Infos pratiques » de la fiche d’une randonnées (Geotrek Rando).
— Si le champ « Filtre » est coché, l’étiquette sera proposée comme filtre dans Geotrek-Rando.
— Les images (hors pictogramme) utilisées dans le contenu de l’étiquette doivent être des liens web.

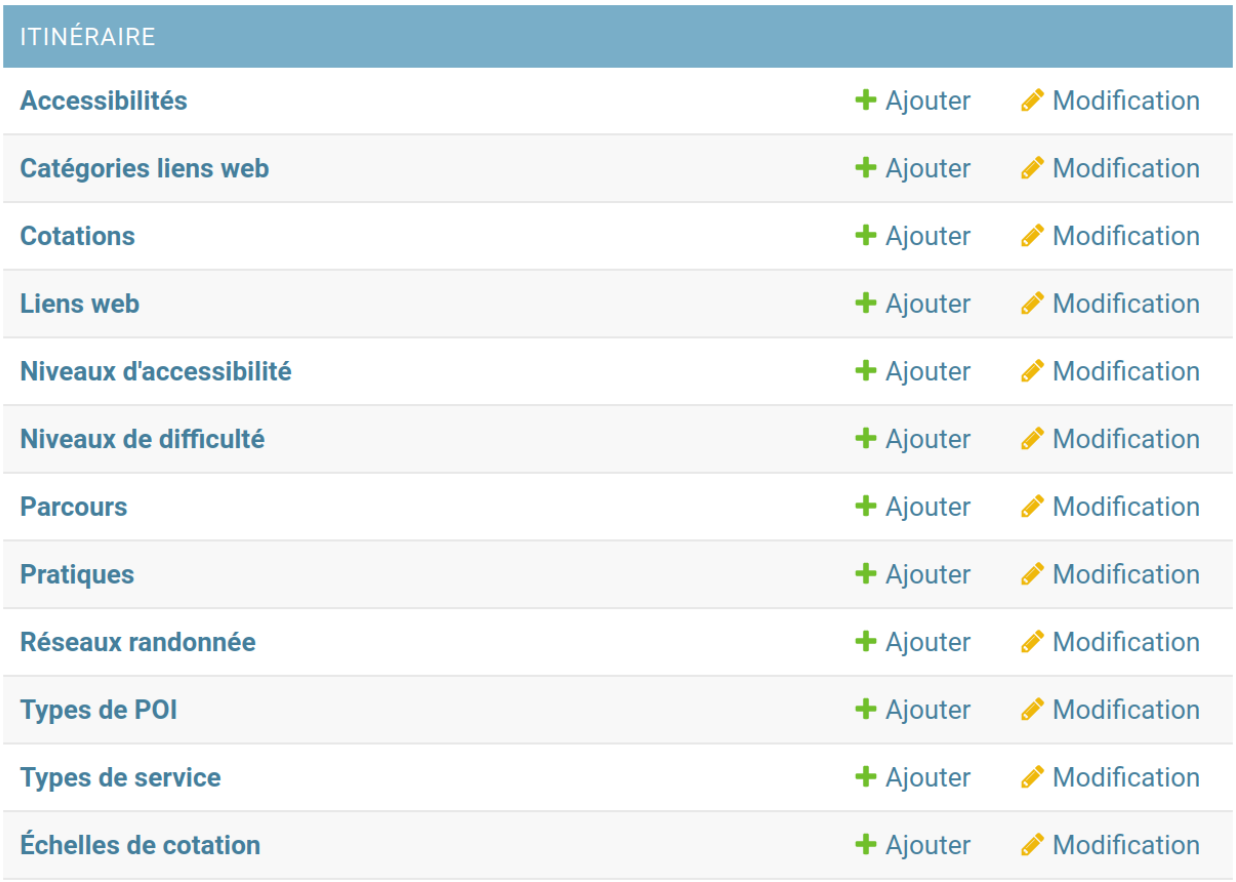


FIG. 2 – Ensemble des champs paramétrables du module Itinéraires

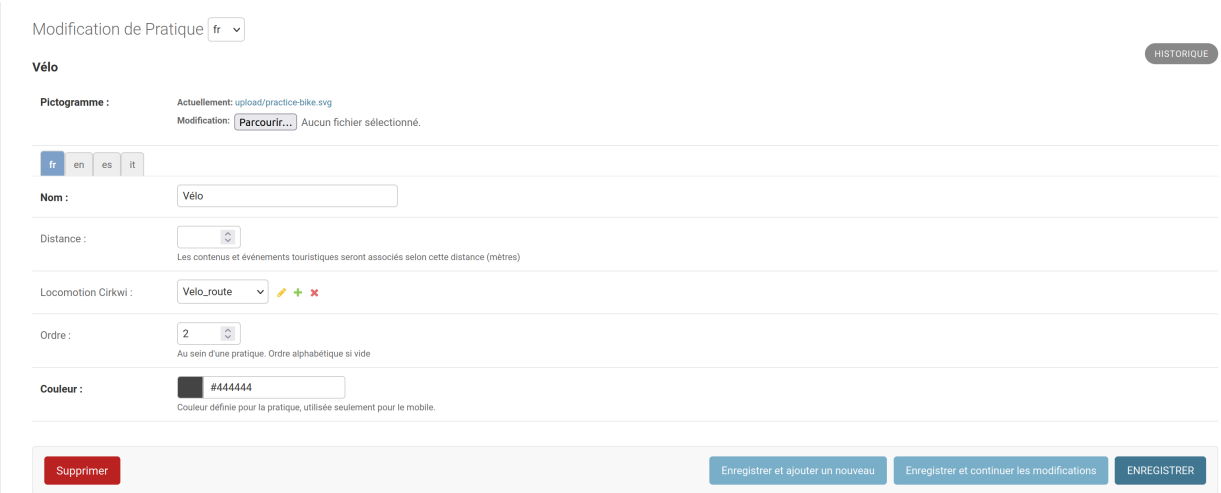


FIG. 3 – Ajout d'une nouvelle pratique

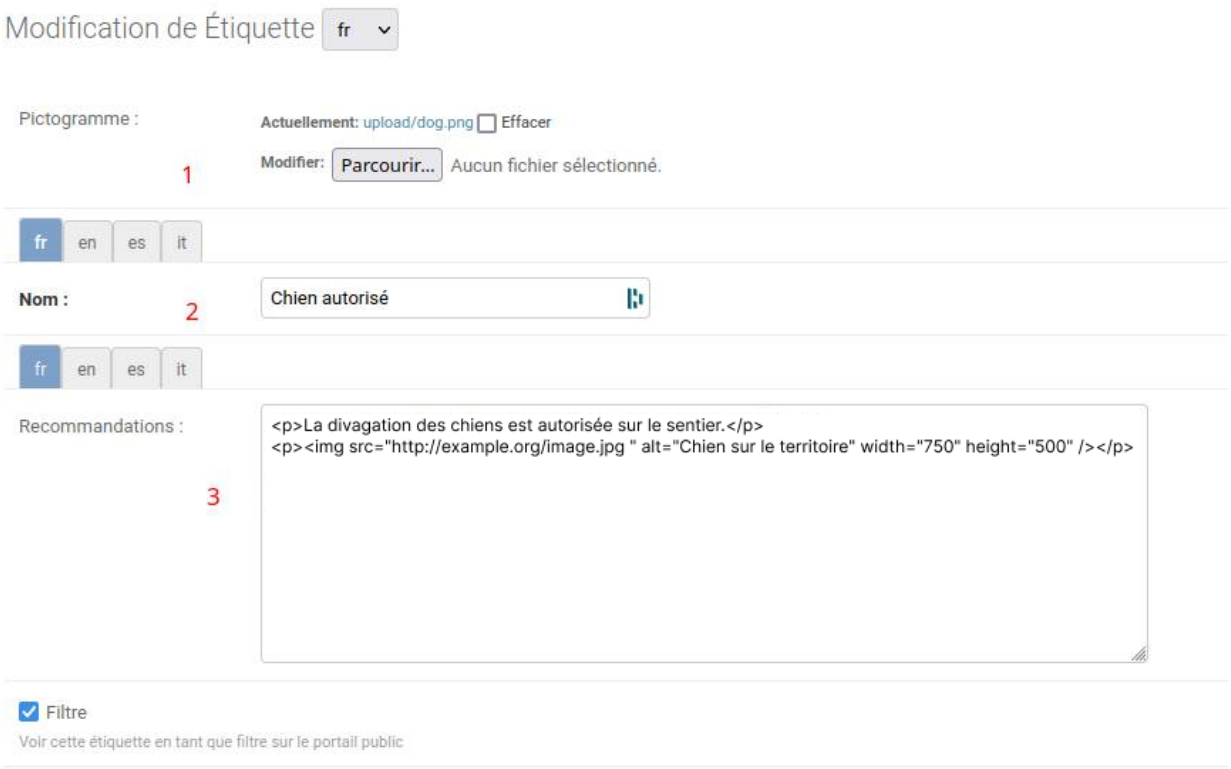


FIG. 4 – Création d'une étiquette

Rendu dans **Geotrek Rando** (onglet *Infos pratiques* d'une fiche randonnée) :

Rendu dans **Geotrek Rando** (partie *Filtres*) :

7.2.2 Tourisme

Exemple : catégorie de contenu touristique et ses sous-types

Pour chaque catégorie il est possible de définir deux listes de sous-types et leur nom.

Édition des sous-types de la catégorie "Hébergements"

7.3 Gestion des utilisateurs

Geotrek-admin s'appuie sur le système d'authentification et de permissions Django .

Les utilisateurs appartiennent à des groupes et les permissions peuvent être attribuées au niveau de l'utilisateur ou du groupe.

Avec les groupes, vous pouvez créer et configurer un profil utilisateur, chacun possédant des permissions spécifiques.

Toute la configuration des utilisateurs, groupes et permissions est disponible depuis le module de configuration, si vous n'avez pas activé *Authentification externe* (voir ci-dessous).

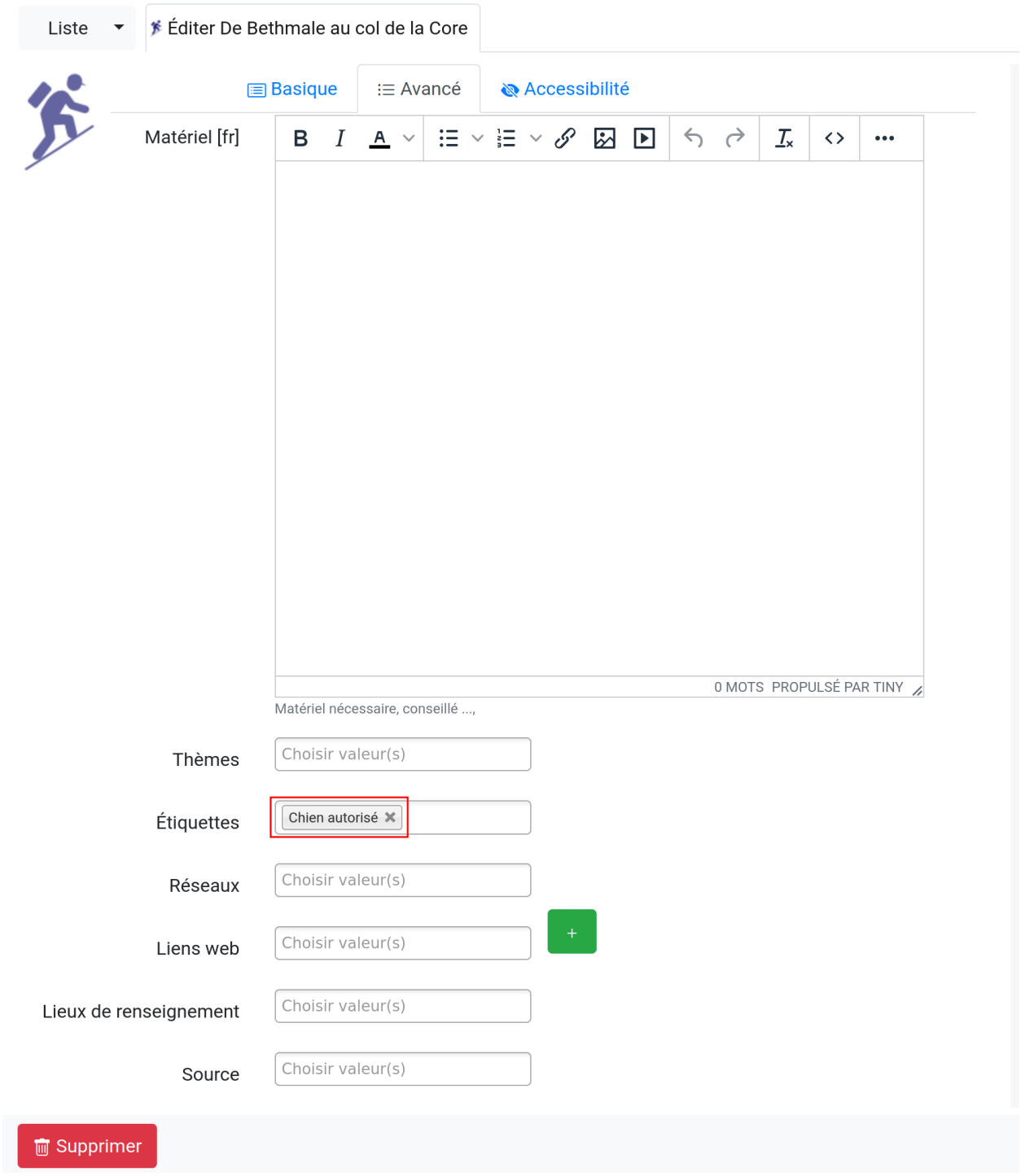


FIG. 5 – Ajout d'une étiquette à un itinéraire

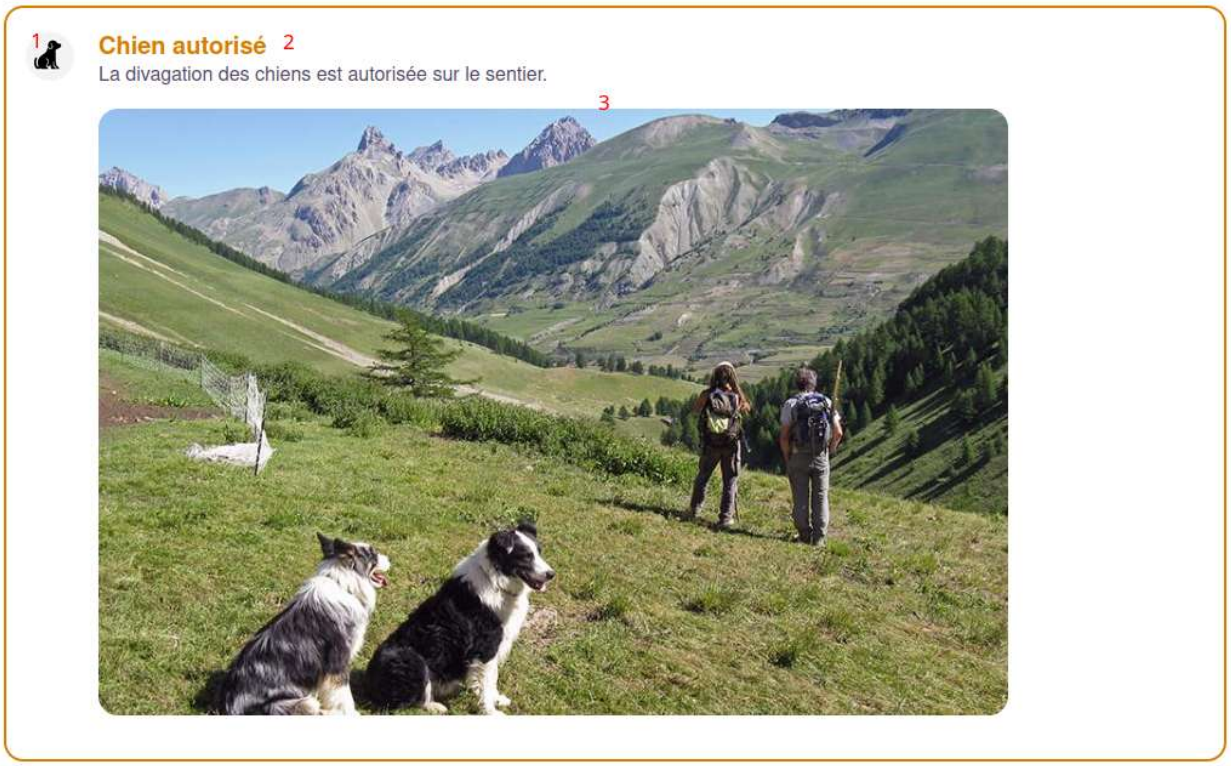


FIG. 6 – Rendu d'une étiquette dans Geotrek-rando

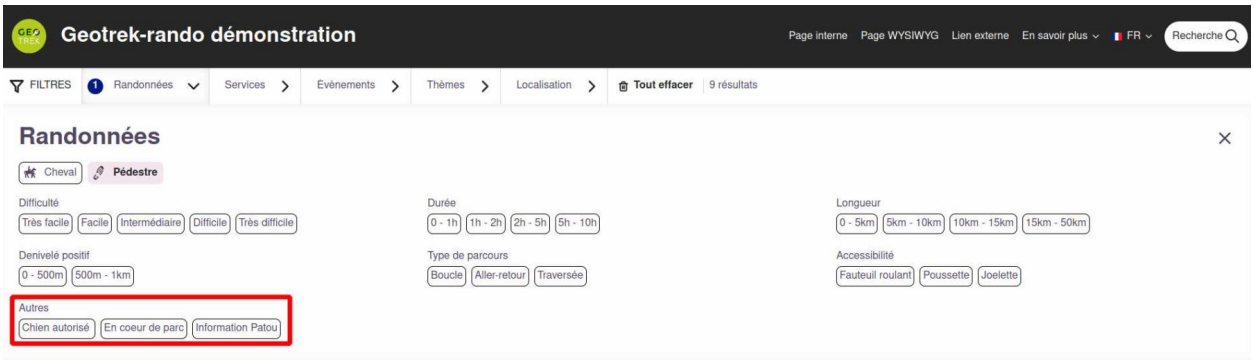


FIG. 7 – Rendu des étiquettes dans les filtres de Geotrek-rando

FIG. 8 – Ensemble des champs paramétrables des modules Contenus et Évènements touristiques

7.3.1 Utilisateurs et permissions

L'application permet d'attribuer des droits à autant de groupes que souhaité. Nous recommandons cependant de privilégier la diffusion la plus large par défaut et de ne restreindre l'application que pour des raisons impératives.

L'application permet de définir des rôles disposant chacun de certains droits.

- visiteurs anonymes (par exemple : tous les internautes)
- partenaires, élus
- agents techniques
- prestataires externes
- administrateurs de l'application
- …

À chaque rôle est associé un ensemble de permissions (consulter telle donnée, la modifier…) qui peuvent être attribuées à un document ou à une catégorie. Les utilisateurs sont ensuite associés à un ou plusieurs rôles.

Un utilisateur donné peut avoir trois niveaux permissions basiques :

- **Actif** : si cette case est coché, l'utilisateur peut se connecter à Geotrek-admin

---

Note : Il est préférable de désactiver un compte lorsqu'un utilisateur n'intervient plus sur Geotrek, plutôt que de le supprimer. En effet supprimer le compte supprimera également par exemple toutes les entrées dans l'historique de Geotrek associées à ce compte.

- **Équipe** : si la case est cochée l'utilisateur pourra accéder au module de configuration de Geotrek-Admin
- **Super-utilisateur** : permet d'octroyer toutes les permissions à un utilisateur sans avoir à les définir explicitement

Un utilisateur peut avoir des permissions spécifiques, définies par type d'objet.

Pour cela, sélectionner les permissions dans l'écran de gauche pour les positionner dans l'écran de droite. Par exemple sur la capture ci-dessous l'utilisateur possède les permissions pour consulter uniquement et exporter les informations relatives aux signalétiques sans possibilité d'accéder aux autres modules ou de modifier les contenus.

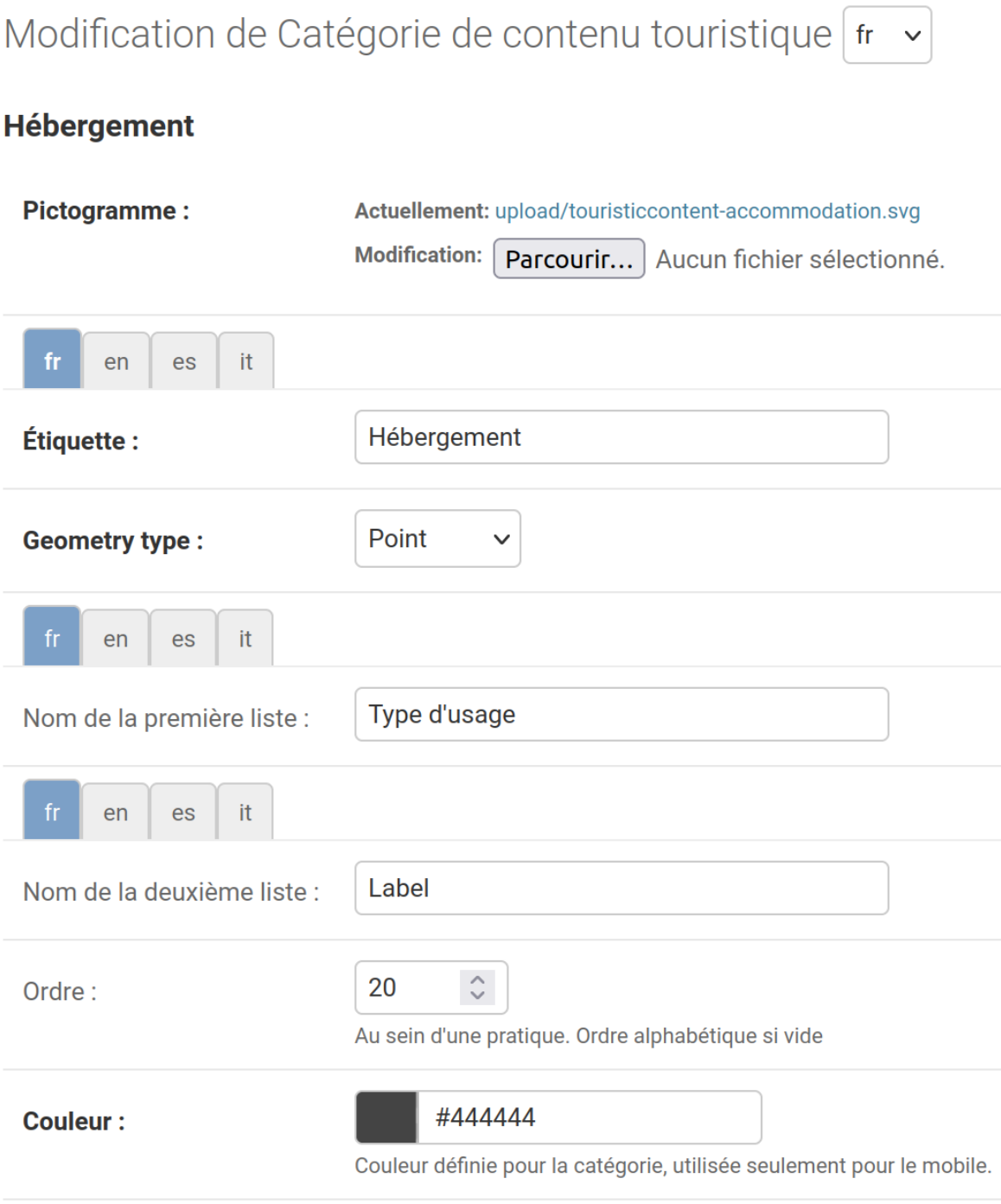


FIG. 9 – Création de catégorie de contenu touristique et ses sous-types

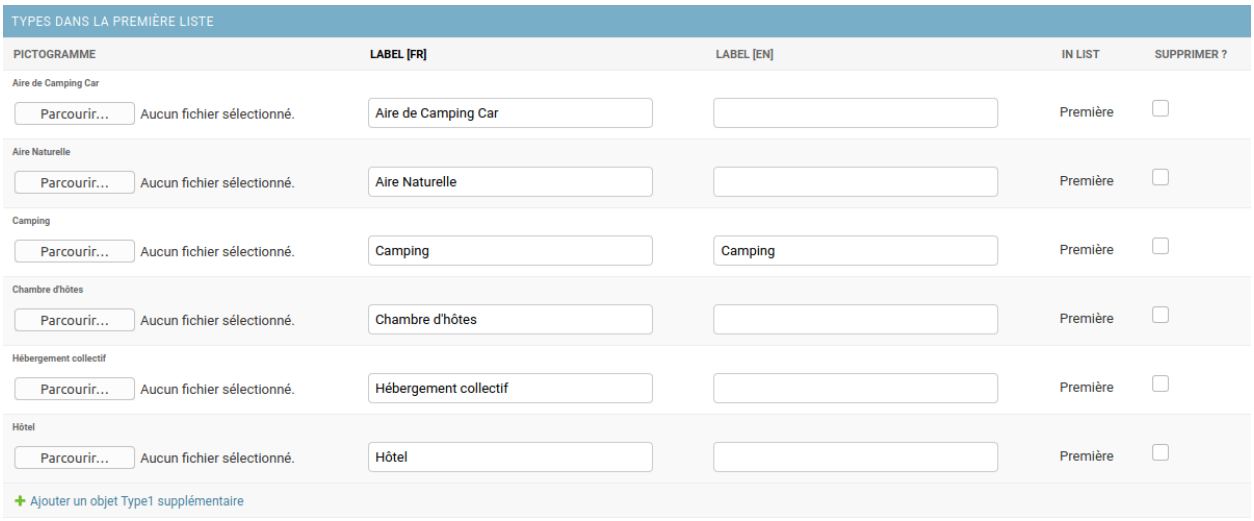


FIG. 10 – Sous-types de la catégorie “Hébergements”

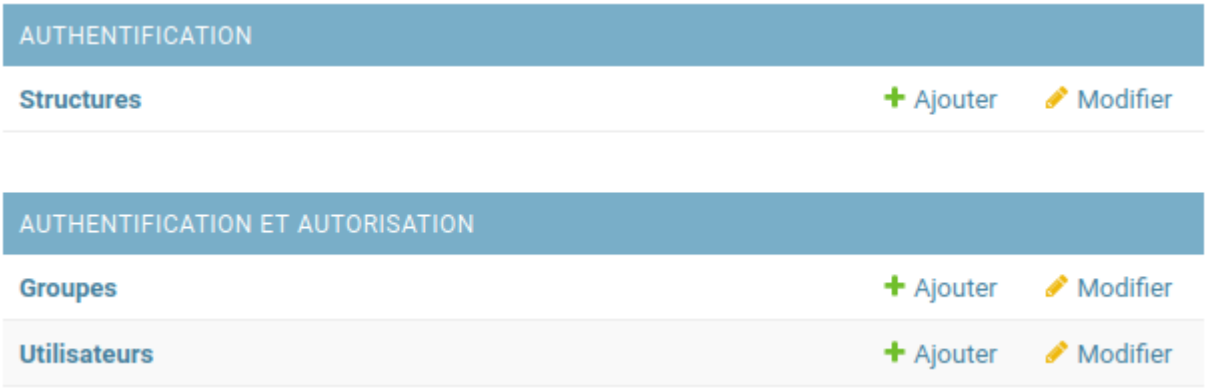


FIG. 11 – Interface de configuration des utilisateurs/groupes

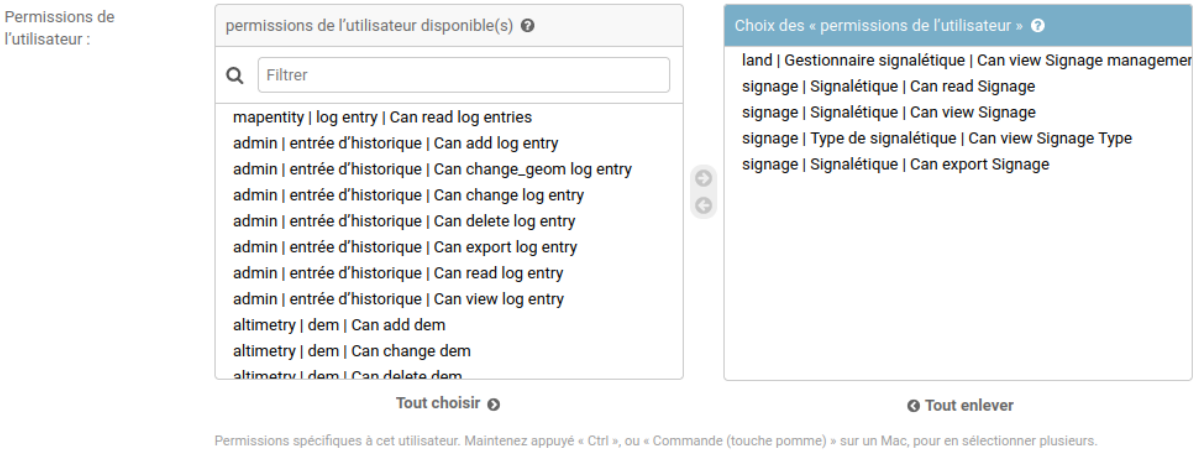


FIG. 12 – Choix des permissions de l’utilisateur

Il existe quatre principaux types de permissions :
- add
- change
- delete
- read / view

Chaque type de donnée est au moins associée à ces quatres actions de base (*ajout*, *modification*, *suppression*, *lecture*). Un type de donnée correspond à un table dans la base de données (*signage_signage*, *trekking_trek*…) »

Voici la signification des actions autorisées dans les permissions :
- *view* : voir les données dans le module de configuration (pour ledonnées dans « category », comme les types de POI, ou les niveaux de difficulté)
- *read* : voir les données dans Geotrek-admin (détail et liste)
- *add* : ajouter une nouvelle donnée (itinéraire, thème…)
- *change* : modifier une donnée
- *change_geom* : modifier la géométrie d'une donnée
- *publish* : publier la donnée
- *export* : exporter les données via l'interface de Geotrek-admin (CSV, JSON…)

7.3.2 Groupes

Les groupes facilitent la gestion des utilisateurs et des permissions.Chaque groupe est configuré avec un certain nombre de permissions.

Dans la vue de modification d'un utilisateur, il est possible d'associer un utilisateur à un ou plusieurs groupes pour bénéficier des permissioncorrespondantes.

Par défaut, six groupes sont disponibles :
- Readers (« Lecteurs »)
- Path managers (« Référents sentiers »)
- Trek managers (« Référents communication »)
- Editors (« Rédacteurs »)
- Geotrek-rando (« Geotrek-rando »)
- Trek and management editors (« Rédacteurs rando et gestion »)

Lorsque l'application est installée, il est possible de modifier les permissions par défaut de ces groupes, d'en créer de nouveaux, etc.

Pour autoriser les utilisateurs à accéder à l'interface AdminSite, accordez-leur le statut équipe. L'interface AdminSite permet aux utilisateurs d'éditer les catégories comme *niveaux de difficulté*, *types de POI*, etc

Il est possible de créer ou de supprimer des comptes administrateurs ou éditeurs. De même qu'il est possible d'ajouter, modifier ou supprimer des itinéraires, tout comme pour les contenus additionnels.

Les éditeurs et les administrateurs peuvent travailler en simultané sur des fiches balades, mais pas en même temps sur la même fiche, car il y aurait sinon un risque de perte d'informations lors de la validation / enregistrement des informations saisies.

7.3.3 Structures

Chaque utilisateur est obligatoirement rattaché à une structure. Lors de l'installation, Geotrek crée une structure par défaut à laquellles premiers utilisateurs seront rattachés.Il est possible d'ajouter de nouvelles structures, reflétant des partenaires territoriaux, entreprises, entités qui seront amenés à travailler à vos côtés sur Geotrek.

Les utilisateurs d'une structure ne peuvent travailler que sur les objets dans Geotrek liés à leur structure. Ils pourront consulter les objets des autres structures mais n'auront pas le droit de les modifier.

Exemple : si on imagine un Geotrek déployé sur l'ensemble du territoire français, il pourrait y avoir des structures correspondant à chaque région. Chaque utilisateur serait rattaché à sa région. Il y aurait alors la garantie qu'un utilisateur de Bretagne ne puisse pas modifier les objets saisis par un utilisateur de Normandie.

Cette notion de structures permet de segmenter les périmètres d'action des utilisateurs et de permettre à différentes entités de travailler sur un même Geotrek-Admin, tout en garantissant une cohérence des données.

Note : Un utilisateur d'une structure pourra tout de même tracer des itinéraires sur des tronçons tracés par une autre structure

---

Note : Pour qu'un utilisateur puisse modifier les objets d'une autre structure il y a deux possibilités :

— celui-ci est super-utilisateur
— celui-ci possède la permission « Can bypass structure », qui permet d'outrepasser la restriction des structures.

Pour définir la structure par défaut, se référer à la section *Default structure*

7.4 Gestion multilingue

La configuration des langues de la plate-forme est réalisée au travers du fichier de configuration principal. Lors de sa mise à jour, la commande de déploiement créé les champs nécessaires qui manquent dans la base de données (exemple : nom_fr, nom_es, nom_it…). Ceci est à distinguer des langues dans lesquelles sont traduits les éléments de l'interface de l'application (Français, Anglais, Italien, Espagnol).

Il est par exemple possible d'intégrer des textes dans différentes langues (à minima français, anglais) dans Geotrek-Admin.

7.5 Configuration des portails

Geotrek permet de configurer un ou plusieurs portails. Ce terme est utilisé pour référencer un site grand public sur lequel seront visibles les objets publiés de Geotrek.

Ainsi, il est possible d'avoir plusieurs Geotrek-Rando branchés sur un seul Geotrek-Admin. Grâce à leur distinction sous forme de portail, il sera alors aisé de choisir sur quel Geotrek-Rando on souhaite faire apparaitre une information.

Avec le widget Geotrek (https://github.com/GeotrekCE/geotrek-rando-widget) il est également possible d'utiliser cette fonctionnalité pour distinguer les contenus à afficher dans un widget ou dans un autre (https://makina-corpus.com/logiciel-libre/developpement-geotrek-widget-finance-parc-naturel-regional-haut-jura).

Pour configurer un ou pluseurs portails, il faut se rendre dans le module de configuration sur la section « Portails cibles ».

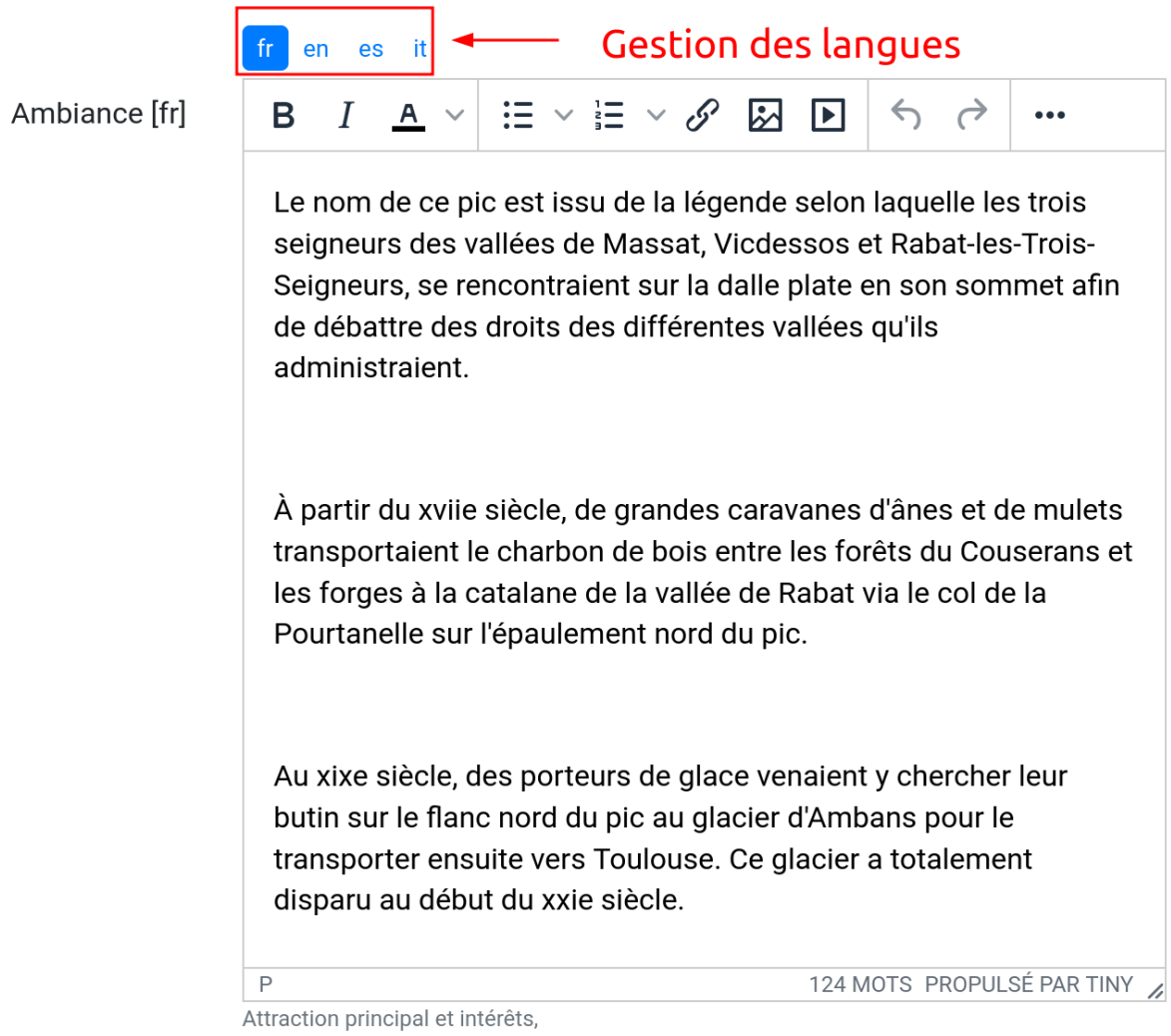


Fig. 13 – Gestion multilingue dans la fiche détail

Fig. 14 – Configuration des portails

Il est possible de choisir de publier sur un ou plusieurs portails les objets suivants : itinéraires, contenus et évènements touristiques, pages statiques. Pour cela il suffit de sélectionner la valeur souhaitée dans le champ « portail » à l'édition de l'objet.

7.6 Intégration des données SIT

Des développements ont déjà été réalisés dans Geotrek pour intégrer des données de divers SIT (Système d'Informations Touristiques), notamment APIDAE, Tourinsoft, LEI, SITLOR… si les flux sont disponibles dans des formats ouverts.

Il convient ensuite de configurer chaque flux dans Geotrek-Admin afin que ceux-ci soient synchronisé avec la plateforme touristique.

La fonctionnalité consiste à absorber le flux XML, afin de le remettre à disposition sous une forme optimisée et épurée. Les données sont alors exposées sous la forme d'une web API, au format GeoJSON. L'obtention d'un flux de données GeoJSON (agnostique et standard) présente de nombreux avantages pour l'intégration dans les applications tierces. Ce format est compatible nativement avec Rando V3, mais également avec la plupart des bibliothèques de cartographie (web et natives mobile).

Pour configurer l'import de SIT, référez vous à cette section *Import data from touristic data systems (SIT)*

7.7 Intégration des fonds de cartes

Il est possible d'intégrer dans Geotrek différents fonds de carte comme :
- OpenStreetMap : https://www.openstreetmap.org/#map=6/46.449/2.210
- OpenTopoMap : https://opentopomap.org/#map=6.49.000/10.000
- ou les données IGN : https://geoservices.ign.fr/services-geoplateforme-diffusion

Pour configurer l'ajout de fonds de plan, référez vous à cette section *Map settings*

7.8 Intégration de pictogrammes

Les pictogrammes contribués dans Geotrek doivent être au format :
- SVG (de préférence, cela permet de conserver la qualité en cas de redimensionnement) ou PNG,
- SVG pour les thèmes (afin de permettre un changement de couleur pour les thèmes sélectionnés),

Il doivent :
- Avoir un viewport carré afin de ne pas être déformés sur le portail,
- Ne pas déborder du cercle inscrit pour les pratiques et les catégories de contenus touristiques, en prévoyant une marge si nécessaire.
- Avoir une dimension minimale de 56x56 pixels en ce qui concerne les PNG

Si vous utilisez Inkscape, vous devez définir une viewBox. Voir la documentation d'Inkscape

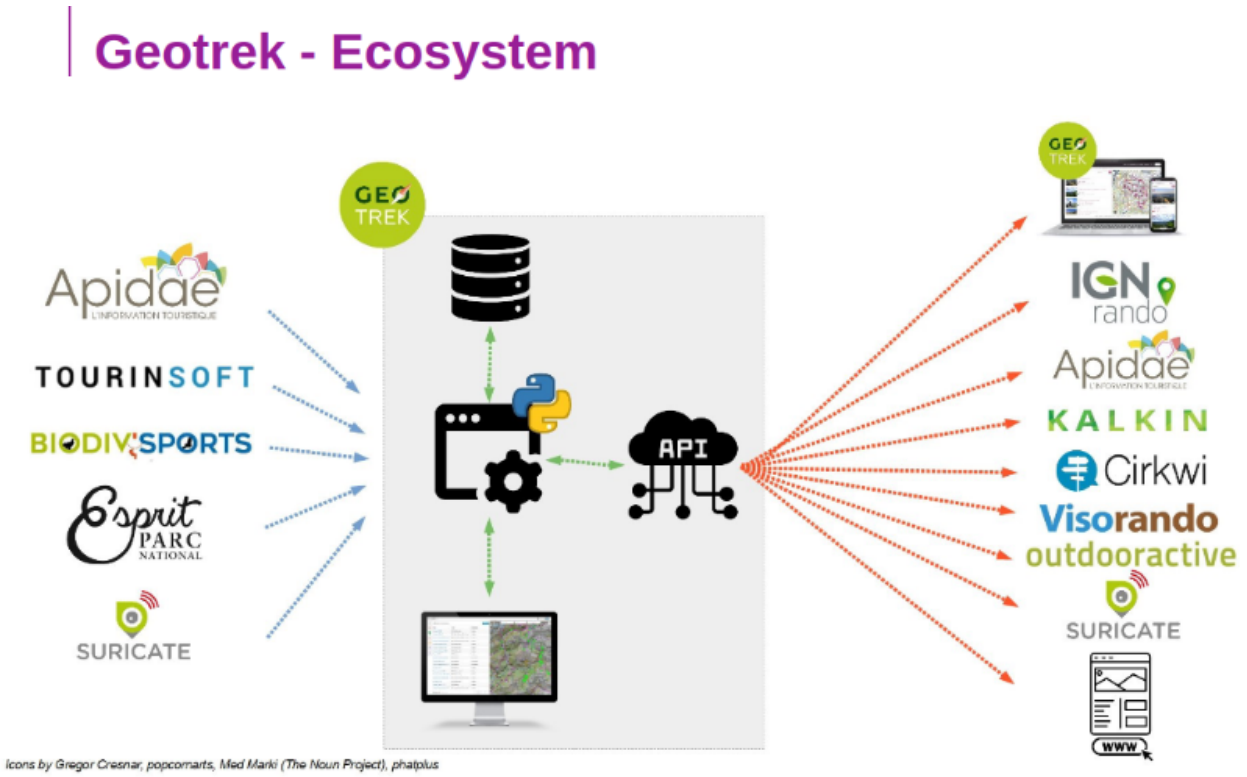
Afin de s'intégrer au mieux dans le design standard, les couleurs suivantes sont recommandées :
- Blanc sur fond transparent pour les pratiques et les catégories de contenus touristiques,
- Gris sur fond transparent pour les thèmes,
- Blanc sur fond orange pour les types de POI.

Voici quelques ressources en ligne proposant des pictogrammes (sous licence libre) :
- https://pictogrammers.com/library/mdi/
- https://thenounproject.com/
- http://map-icons.com/
- https://www.opensymbols.org/
- https://www.svgrepo.com/
- http://www.entypo.com/
- https://icons.getbootstrap.com/
- https://icongr.am/
- https://cocomaterial.com/
- https://icofont.com/
- https://fontello.com/
- https://iconmonstr.com/
- https://fontawesome.com/icons

CHAPITRE 8

APIs

— *API Geotrek*
— *APIs externes*
  — *Geotrek et IGNrando”*
  — *Geotrek et APIDAE*



8.1 API Geotrek

Geotrek dispose d'une API (Application Programming Interface) qui sert à exposer les données stockées dans une instance de Geotrek-admin, dans le but de pouvoir la faire communiquer avec d'autres outils, systèmes et plateforme et ainsi échanger des données.

Cette API, désormais dans sa version 2 permet à toute structure tierce de récupérer des données et de les intégrer dans son système ou ses applications.

À ce jour de nombreux partenaires des structures utilisatrices de l'application Geotrek ont déjà utilisé cette API pour intégrer les données dans leurs outils.

L'API Geotrek est le point central pour permettre les interconnexions avec divers services. Grâce à cette interface de données, Geotrek s'est positionné comme un point central dans un écosystème de solutions du monde de la gestion et promotion des activités de pleine nature.

Pour changer les paramètres d'accès de l'API, référez vous à cette section *API*

8.2 APIs externes

8.2.1 Geotrek et IGNrando"

Geotrek-admin est capable de produire un flux des itinéraires et POIs présents dans sa BDD au format Cirkwi pour pouvoir les importer directement dans IGNrando" (https://makina-corpus.com/sig-webmapping/geotrek-et-lign-ca-fonctionne).

Exemple des randonnées et POIs du Parc national des Ecrins publiées sur IGNrando" depuis Geotrek-admin : https://ignrando.fr/fr/communautes/parc-national-des-ecrins

Depuis cette version, 2 flux sont automatiquement générés par Geotrek-admin au format attendu par l'IGN :

— [URL_GEOTREK-ADMIN]/api/cirkwi/circuits.xml
— [URL_GEOTREK-ADMIN]/api/cirkwi/pois.xml

Il est possible d'exclure les POI du flux pour ne diffuser que les randonnées. Pour cela, ajouter le paramètre `?withoutpois=1` à la fin de l'URL (`http://XXXXX/api/cirkwi/circuits.xml?withoutpois=1`).

Il est possible de filtrer les POI du flux par structure. Pour cela, ajouter le paramètre `?structures=<identifiant_de_la_structure>` à la fin de l'URL (`http://XXXXX/api/cirkwi/pois.xml?structures=2`). Vous pouvez filtrer avec plusieurs structures : en séparant les identifiants par des virgules (`http://XXXXX/api/cirkwi/pois.xml?structures=2,5,3`).

Il est également possible de filtrer les randonnées du flux par structure et par portail. Pour cela, ajouter le paramètre `?structures=<identifiant_de_la_structure>` ou `?portals=<identifian_de_la_structure>` à la fin de l'URL (`http://XXXXX/api/cirkwi/circuits.xml?portals=3`).

Il est également possible d'exclure du flux les randonnées provenant de sources externes à Geotrek-Admin. Ce filtre est notamment nécessaire pour ne pas renvoyer à Cirkwi les randonnées qui en proviennent déjà. Pour cela, ajouter le paramètre `?include_externals=false` à la fin de l'URL (`http://XXXXX/api/cirkwi/circuits.xml?include_externals=false`).

Il est possible de cumuler ces différents filtres, en séparant les valeurs par un `&` (`http://XXXXX/api/cirkwi/circuits.xml?portals=3&structures=1&include_externals=false`).

Il est également possible d'exclure du flux les randonnées provenant de sources externes à Geotrek-Admin. Ce filtre est notamment nécessaire pour ne pas renvoyer à Cirkwi les randonnées qui en proviennent déjà. Pour cela, ajouter le paramètre `?include_externals=false` à la fin de l'URL (`http://XXXXX/api/cirkwi/circuits.xml?include_externals=false`).

Il est possible de cumuler ces différents filtres, en séparant les valeurs par un & (`http://XXXXX/api/cirkwi/circuits.xml?portals=3&structures=1&include_externals=false`).

Le référentiel CIRKWI a été intégré dans 3 tables accessibles dans le module de configuration (à ne pas modifier) :

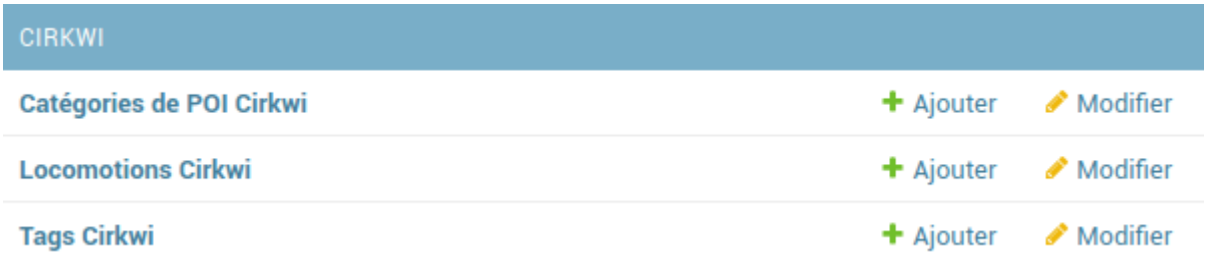


FIG. 1 – Ensemble des champs paramétrables pour le référentiel CIRKWI

Si vous ne souhaitez pas utiliser les valeurs par défaut ou avez créez vos propres typologies, il faut que vous renseigniez les correspondances entre les catégories de votre Geotrek et celles du référentiel IGN (Cirkwi) dans le module de configuration. Comme indiqué ici : https://github.com/GeotrekCE/Geotrek-admin/issues/806.

- Pratique >> locomotion/loisirs
- Accessibilite >> thematiques/tags
- Themes >> thematiques/tags
- Types de POI >> Categories POI

Les correspondances avec les valeurs de ces 3 tables sont donc à renseigner dans les tables Geotrek des Pratiques, Accessibilités, Thèmes et Types de POI.

Ce même flux est aussi utilisable pour alimenter directement la plateforme Cirkwi : https://pro.cirkwi.com/importez-vos-donnees-geotrek-dans-cirkwi/.

Note

Geotrek-admin dispose aussi d'une API générique permettant d'accéder aux contenus d'une instance à l'adresse : `[URL_GEOTREK-ADMIN]/api/v2/`

8.2.2 Geotrek et APIDAE

Il existe plusieurs passerelles entre la plateforme d'informations touristiques APIDAE et Geotrek.

APIDAE vers Geotrek

Actuellement, certains contenus touristiques peuvent être synchronisés automatiquement avec une base APIDAE. Il s'agit des contenus situés dans les catégories suivantes :

- Contenus touristiques (hébergements, restaurants, produits du territoire, lieux de visites…)
- Evénements touristiques (expositions, conférences, sorties…)

Les contenus touristiques peuvent aussi être synchronisés depuis des flux Tourinsoft ou Esprit Parc National.

Il est également possible de mettre en place des passerelles pour importer des POIs, des lieux de renseignement, des aménagements ainsi que des randonnées d'APIDAE vers Geotrek. Il est aussi possible d'enrichir le lien avec les contenus touristiques pour avoir par exemple d'autres catégories.

Pour configurer APIDAE, se référer à cette section *Configure APIDAE (ex-SITRA) import*

Geotrek vers APIDAE

Il existe aussi un lien dans l'autre sens, permettant d'importer automatiquement vers APIDAE les itinéraires existants dans une instance Geotrek.

L'API permet de connecter une instance Geotrek pour importer des itinéraires vers les objets de type « Équipements » dans APIDAE.

Les randonnées VTT, trail, vélo et les tours itinérants sont également intégrés dans la passerelle.

Pour plus d'information, se référer à la documentation en ligne de Sitourisme.

CHAPITRE 9

Pages statiques

— *Créer une page statique*
— *Construire une page statique*
— *Ajouter du contenu dans un bloc*
— *Astuces*

Danger : Depuis Geotrek-Rando V3, le composant Bootstrap n'est plus utilisé pour gérer les différentes tailles d'écran. Cela signifie que la mise en page créée dans Geotrek-Admin n'est pas reproduite sur le site public. Cette documentation n'est valable que pour Geotrek-Rando V2 en attendant sa mise à jour expliquant le fonctionnement actuel. Voir le ticket : https://github.com/GeotrekCE/Geotrek-rando-v3/issues/466

Les pages statiques sont les pages d'information et de contextualisation de votre portail web Geotrek-rando. Comme pourraient l'être les premières pages d'un topo-guide papier. Elles peuvent aussi être consultées dans votre application Geotrek-mobile.

Elles permettent de fournir à l'internaute et futur randonneur des informations génériques : présentation de votre structure, votre projet de randonnée, recommandations, informations pratiques, etc.

Elles sont gérées depuis le module de configuration de Geotrek-admin et sont ensuite publiées sur Geotrek-rando à chaque synchronisation du contenu.

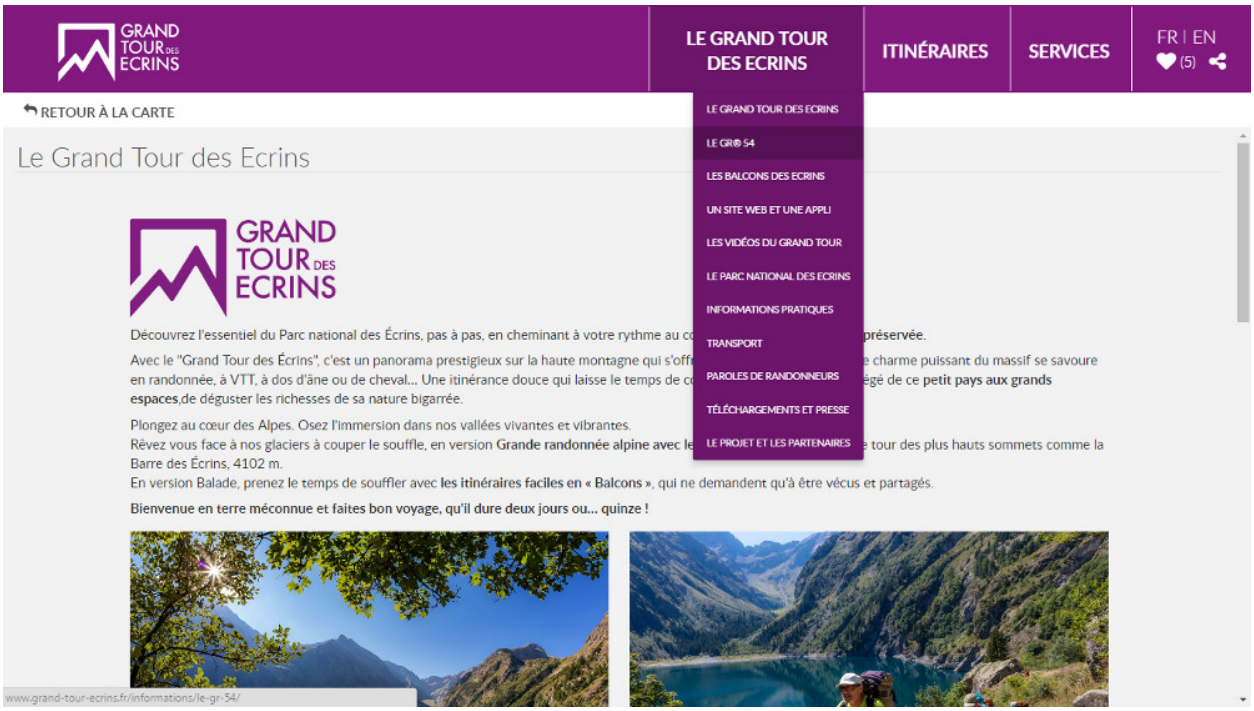
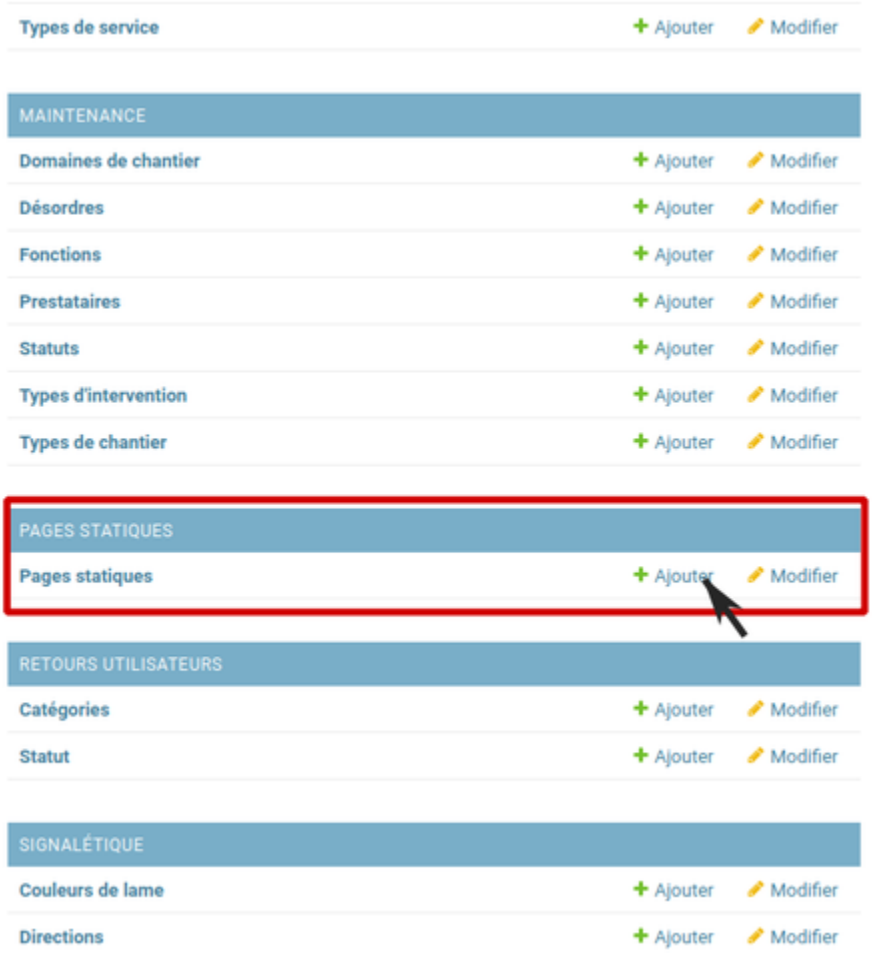


FIG. 1 – Exemple de page statique (http://www.grand-tour-ecrins.fr/informations/le-grand-tour-des-ecrins/)

9.1 Créer une page statique

Depuis le module de configuration de Geotrek-admin, sélectionnez « Pages statiques » dans la rubrique « Flatpages ».



Vous accédez alors à la liste des pages statiques. Cliquer sur « Ajouter Page statique » en haut à droite de l'écran pour créer une première page.

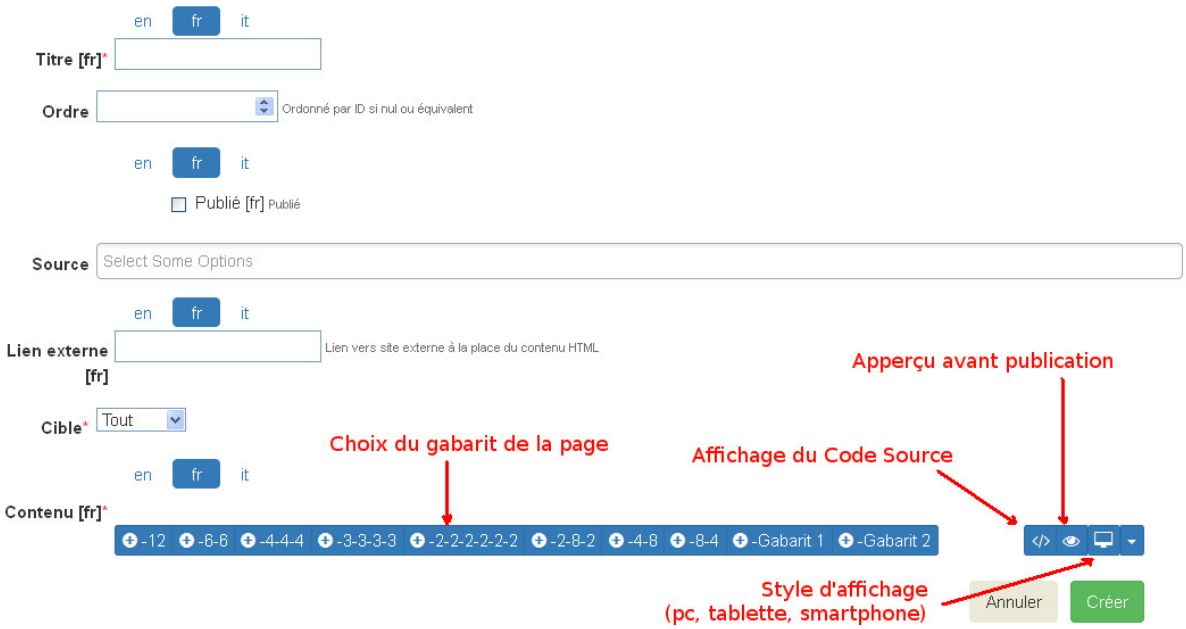
9.2 Construire une page statique

Sélectionnez la langue du contenu que vous souhaitez saisir : en / fr / it…

Saisissez :

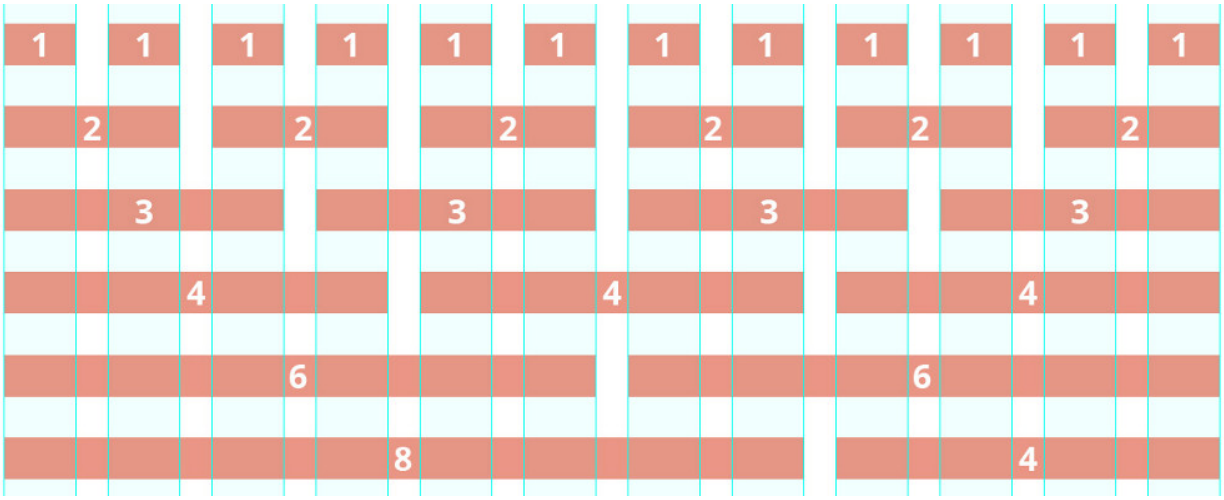
— un titre (sans guillemets, parenthèses, etc.)
— un ordre optionnel (pour définir l'ordre d'apparition dans le menu de votre Geotrek-rando)
— cochez « publié » lorsque vous souhaiterez mettre en ligne votre page
— définissez la « source » (comprendre ici la destination d'affichage et donc votre Geotrek-rando)
— sélectionnez une cible (Geotrek-rando et/ou Geotrek-mobile ou cachée pour créer une page qui ne sera pas listée dans le menu).

Attention, à chaque fois que cela vous est demandé, veillez à sélectionner la langue de votre contenu.

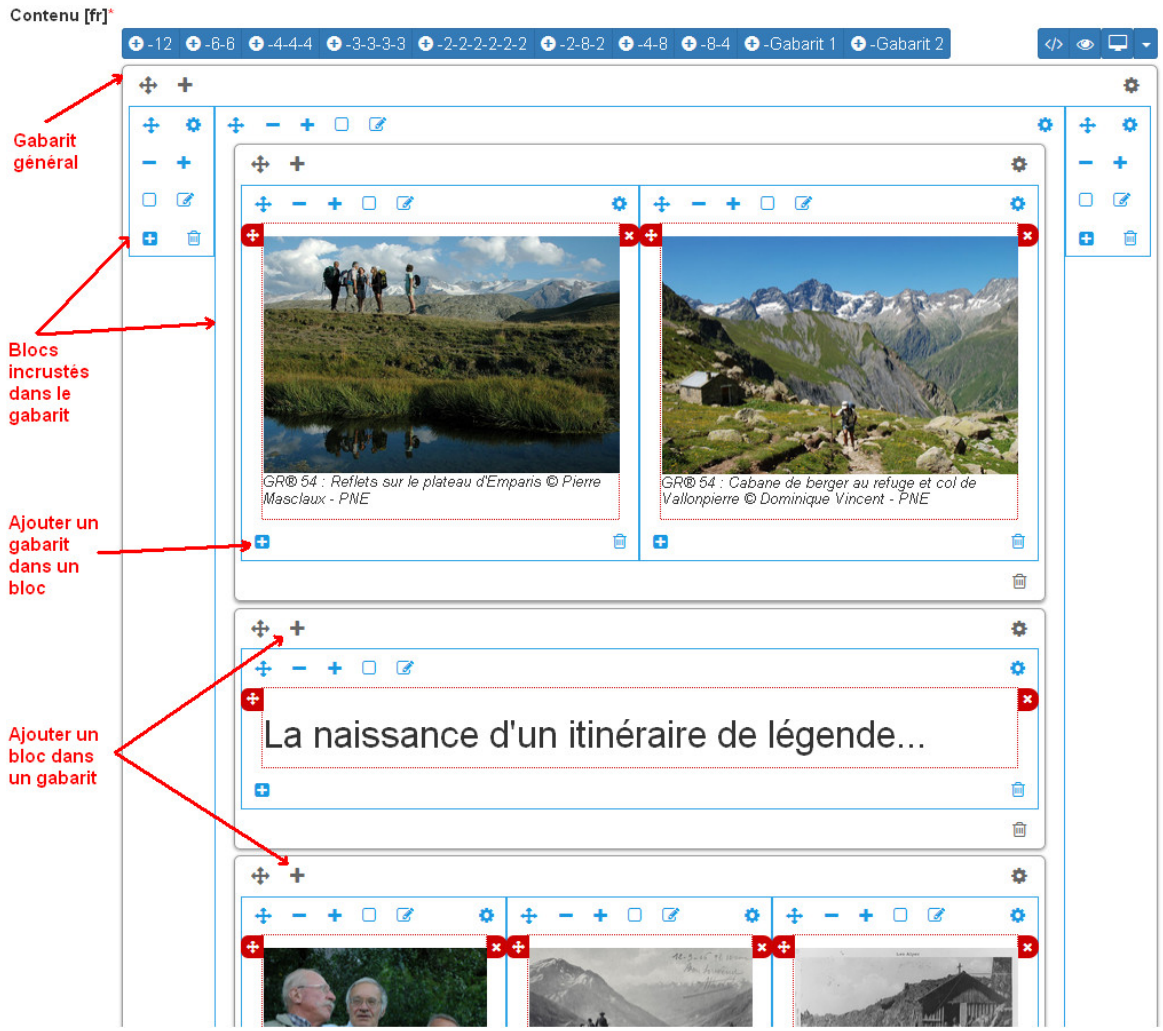


L'interface permet de construire sa page en responsive design, c'est-à-dire qu'il est possible de disposer les blocs de contenu pour s'adapter aux différentes tailles d'écrans des utilisateurs.

Choisissez le gabarit sur lequel vous souhaitez construire votre page : 12 / 6-6 / 4-4-4 / etc. Ce sont des formats prédéfinis d'assemblage de blocs basés sur 12 colonnes qui occupent 100% de la largeur de l'écran (Bootstrap).

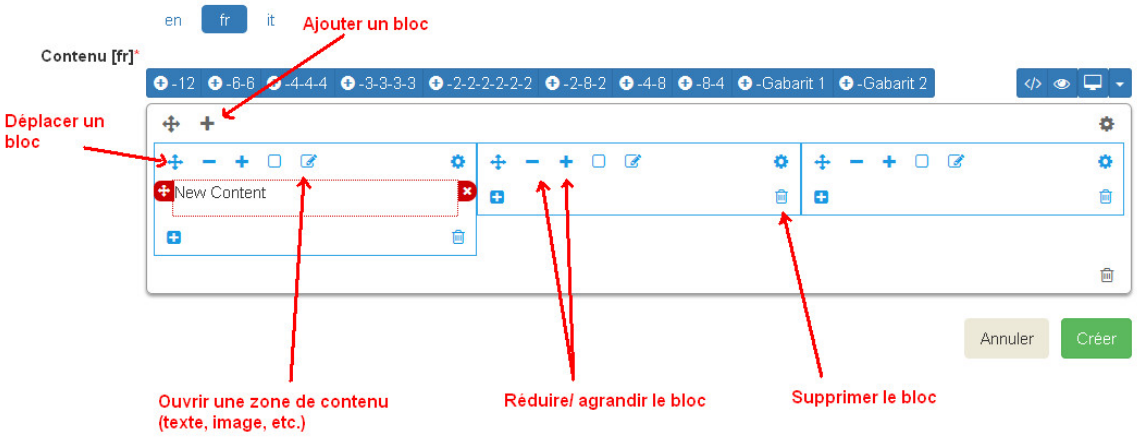


Vous pouvez aussi utiliser ou vous inspirer des 2 gabarits d'exemple (Gabarit 1 et Gabarit 2).



Vous pouvez ajouter autant de gabarits que vous le souhaitez sur une seule page.

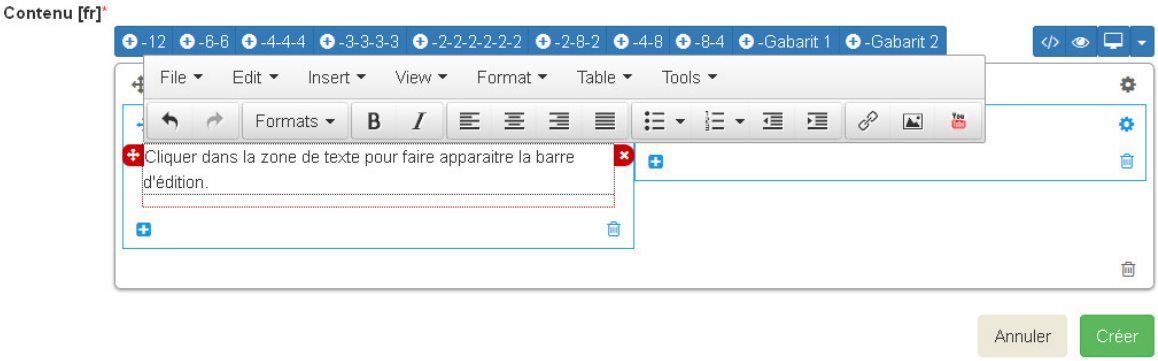
Une fois que vous avez ajusté vos blocs de contenu pour un affichage sur ordinateur (Desktop), vous devez basculer sur l'affichage sur mobile (Phone) pour l'adapter à de plus petits écrans (en cliquant sur les + et - bleus de chaque bloc). Privilégiez alors des blocs sur une colonne faisant 100% de large.



9.3 Ajouter du contenu dans un bloc

En cliquant dans la zone de texte, une barre d'édition apparaît. Sur un format classique comme dans les logiciels de traitement texte, plusieurs menus et outils sont alors disponibles :

- File : (fichier)
- Edit : retour, copier-coller,
- Insert : Insérer une image, un lien, des caractères spéciaux



Insérer une image : cela ouvre une nouvelle fenêtre avec différents champs à remplir :

- Source : insérer l'URL de l'image (idéalement dans le répertoire /custom/public/images/ de votre Geotrek-rando)
- Image description : légender l'image pour optimiser son référencement
- Dimensions : ajuster le format et cocher « Constrain proportions »

Insérer un lien : cela ouvre une nouvelle fenêtre avec différents champs à remplir :

- URL : lien de destination
- Title : texte à afficher pour le lien
- Target : « New window » si vous souhaitez que le lien s'ouvre dans un nouvel onglet

- View : « Show blocks » permet de faire apparaître les différents paragraphes de votre texte. Elles sont utiles à la structure de votre texte.
- Format : gras, italique, souligner, etc. Le sous-menu « Formats » permet de choisir un style prédéfini pour les titres (Heading 1, Heading 2, etc.). Pour que le style s'applique uniquement au titre et non pas à tout le texte, faire un retour à la ligne et vérifier sa prise en compte en activant « Shox blocks ».
- Table : insertion de tableau
- Tools : Afficher le code source de la page

9.4 Astuces

1. Ne jamais utiliser la touche retour du clavier [?] sans avoir le curseur sélectionné dans une zone de texte. Cela équivaut à revenir à la page précédente et vous perdrez tout votre contenu sans le sauvegarder.
2. Pour reproduire une page dans une langue différente : copier le Code Source et coller-le Code Source de votre nouvelle langue. Vous n'aurez plus qu'à traduire votre texte ! Idem pour traduire un contenu dans une autre langue.
3. Si deux de vos pages ont le même numéro d'ordre d'apparition, une seule des deux sera affichée sur la plate-forme numérique.

CHAPITRE 10

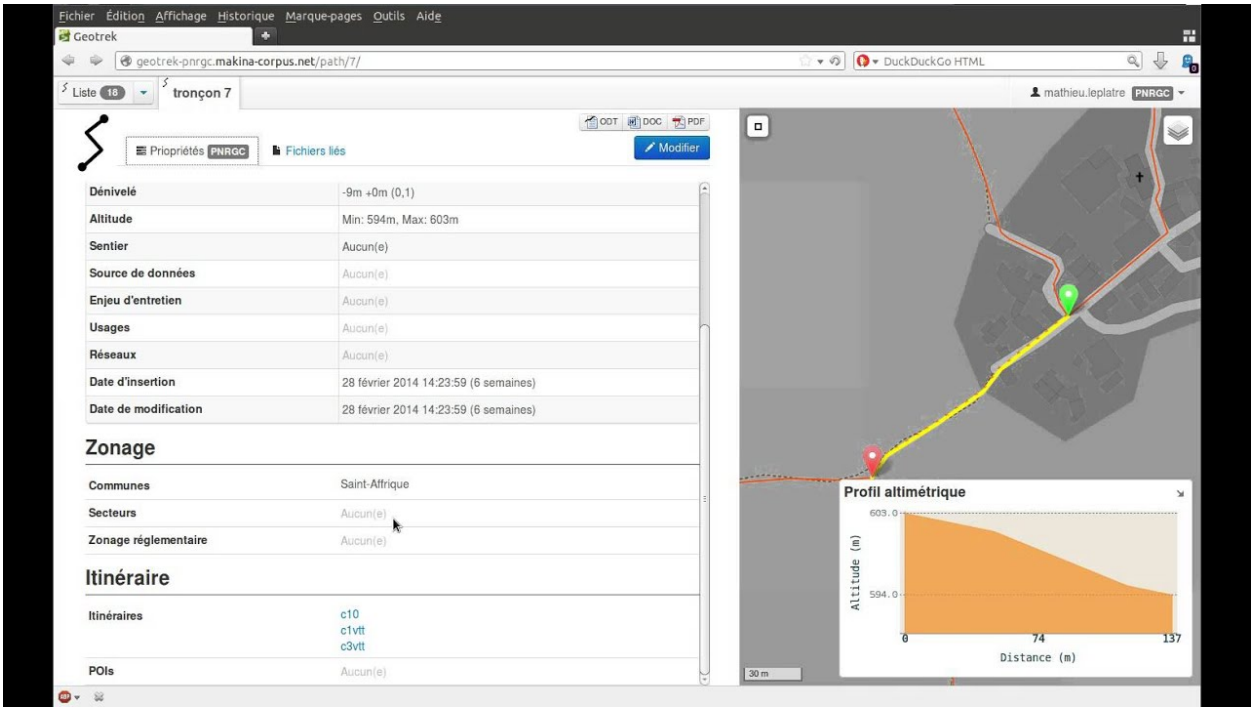
Tutoriels

— *Vidéos*
— *Visualiser les données dans QGIS*
  — *Création de vues SQL pour afficher des couches dans QGIS*
  — *Créer une connexion à la base de données PostgreSQL du projet*
  — *Créer un projet QGIS à partir des vues SQL*

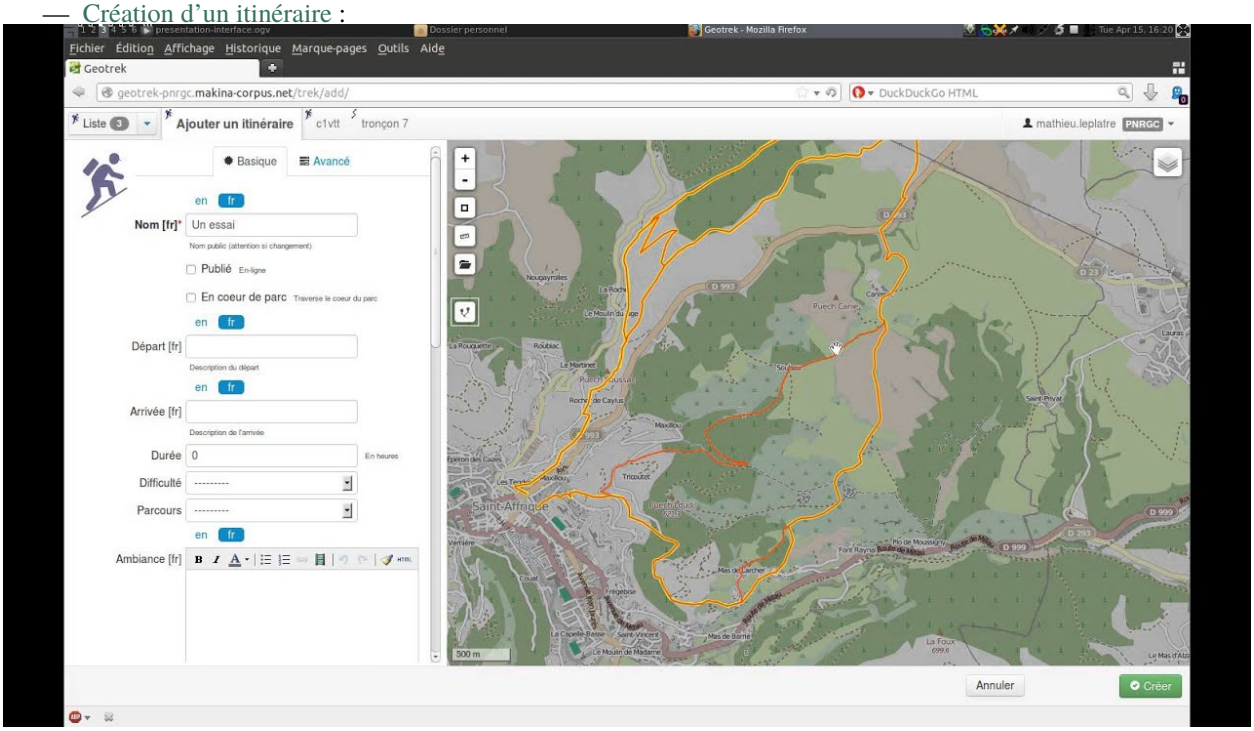
10.1 Vidéos

(In French)

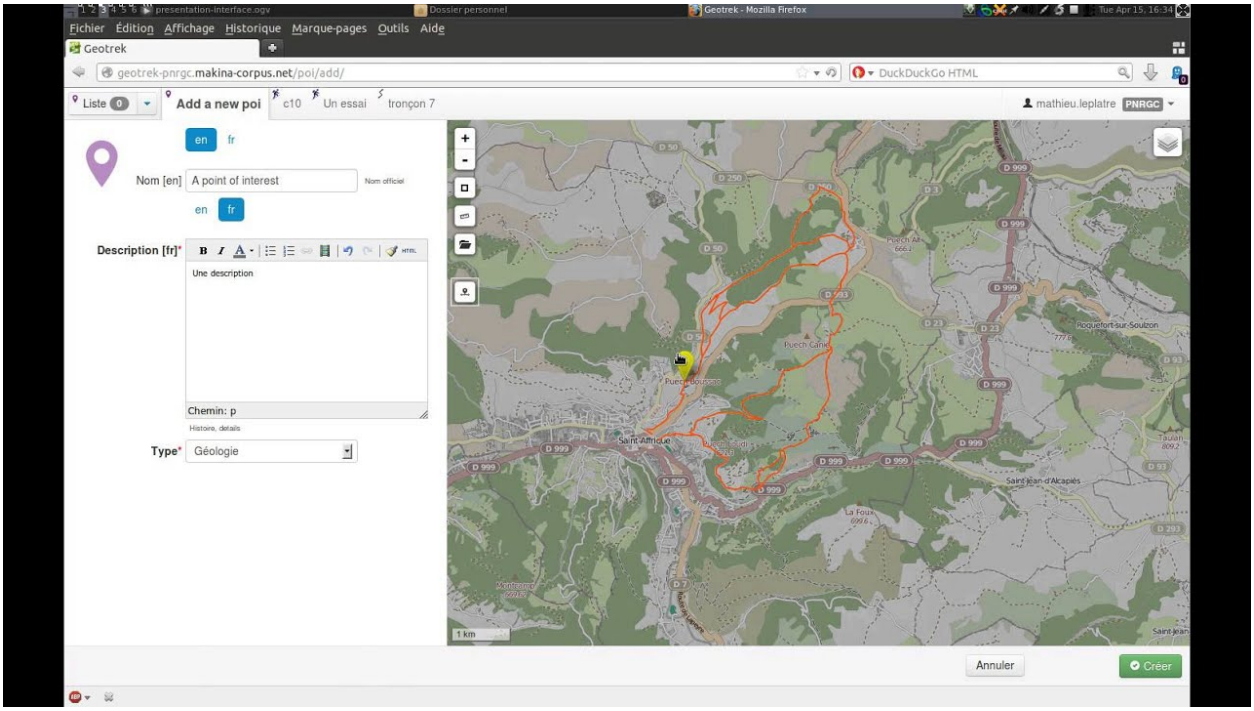
— Présentation de l'interface :



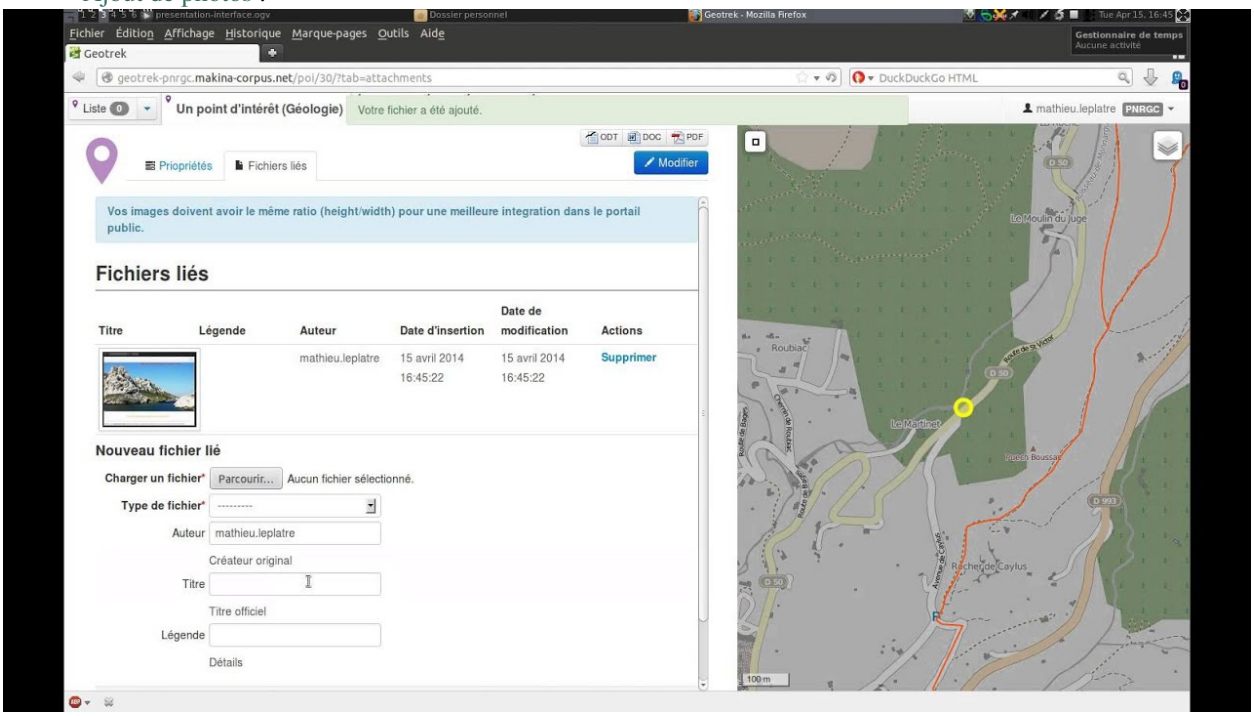
— Création d'un itinéraire :



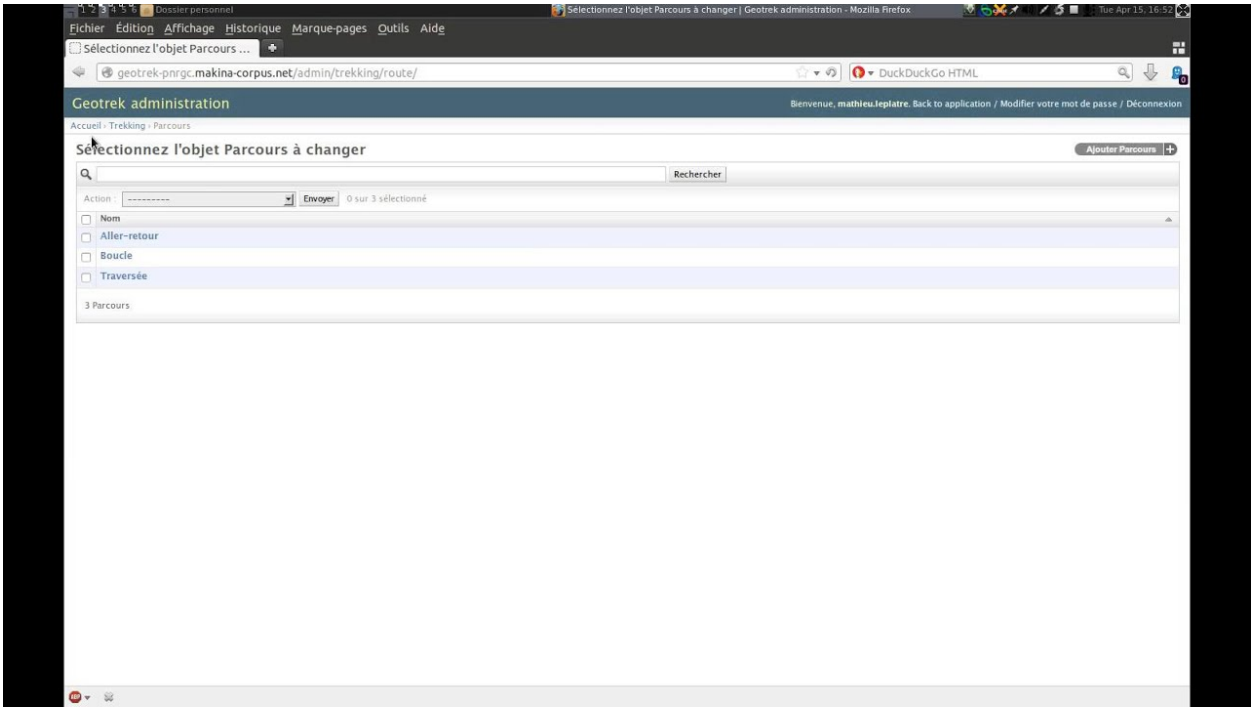
— Création d'un POI :



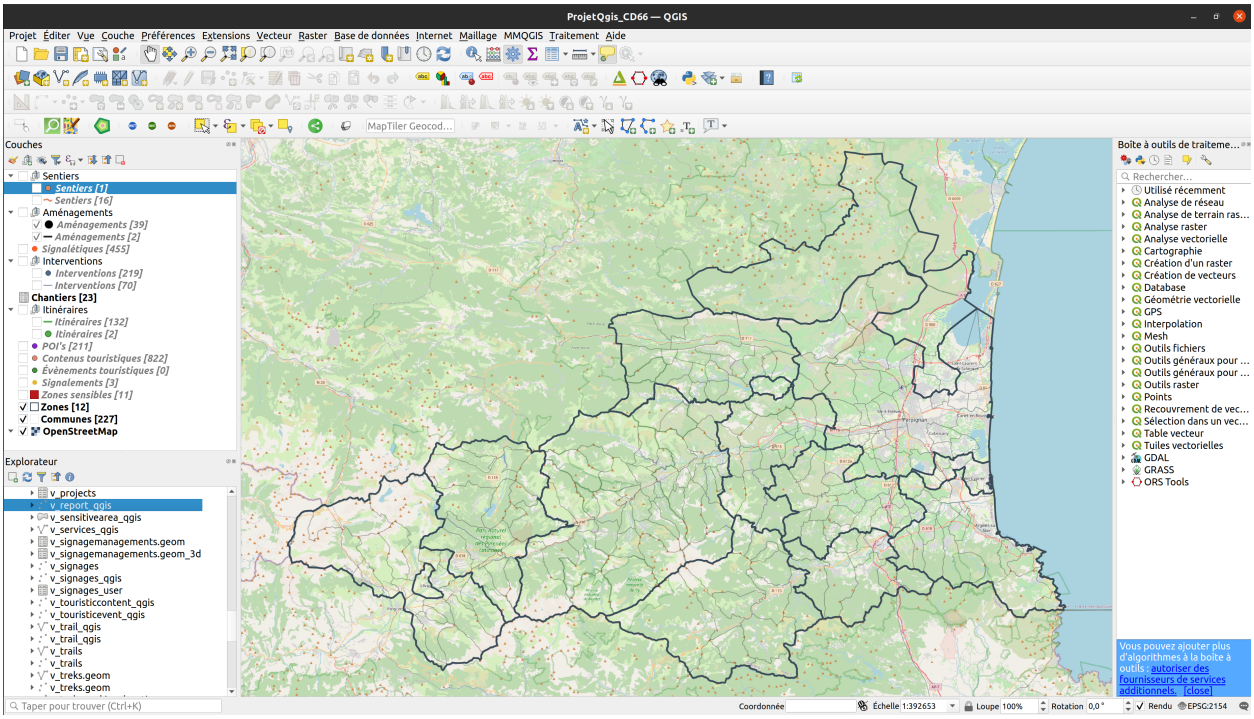
— Ajout de photos :



— Administration :



10.2 Visualiser les données dans QGIS



10.2.1 Création de vues SQL pour afficher des couches dans QGIS

Des vues SQL ont été créées dans la base de données PostgreSQL de Geotrek-admin dans le but de les afficher dans QGIS ou autre (https://makina-corpus.com/django/creer-vues-sql-django-et-afficher-dans-un-sig). Ces vues contiennent les informations essentielles que l'on retrouve dans Geotrek-admin au niveau de chaque module.

Ces vues sont consultables en lecture seule dans QGIS sous forme de couche SIG.

Les modifications se font directement dans Geotrek-admin pour chaque projet, et elles sont répercutées instantanément dans les vues SQL.

10.2.2 Créer une connexion à la base de données PostgreSQL du projet

1. Ouvrir le logiciel QGIS
2. Créer une nouvelle connexion de base de données PostgreSQL

— Dans l'Explorateur > PostgreSQL > Nouvelle connexion
— Renseigner les informations suivantes :
  — Nom de la connexion
  — Hôte
  — Port
  — Base de données
  — SSL mode : permet
  — Nom d'utilisateur
  — Mot de passe
— Cliquer sur « Tester la connexion »
— Si la connexion est réussi, cliquer sur OK pour enregistrer la connexion

10.2.3 Créer un projet QGIS à partir des vues SQL

Afficher une vue SQL sous forme de couche

- Dans l'Explorateur > PostgreSQL > Ouvrir la connexion précédemment créé > Schéma public
- Ajouter les vues : Clic droit sur l'objet > Ajouter la couche au projet
- **Correspondance couches <> vues**
  - Sentiers <> *v_trails*
  - Aménagements <> *v_infrastructures*
  - Signalétiques <> *v_signages*
  - Interventions <> *v_interventions*
  - Chantiers <> *v_projects*

    - Itinéraires <> *v_treks*
    - POI's <> *v_pois*
    - Contenus touristiques <> *v_touristiccontents*
    - Évènements touristiques <> *v_touristicevents*
    - Signalement <> *v_reports*
    - Zones sensibles <> *v_sensitivearea_qgis*
    - Zones <> *v_districts*
    - Communes <> *v_cities*
- Couches supplémentaires (dépend des projets) * Sites outdoor <> *v_outdoor_sites* * Parcours outdoor <> *v_outdoor_courses*

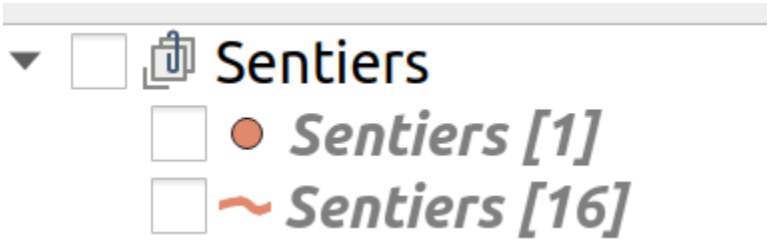
Afficher un fond de plan OpenStreetMap

- Dans l'Explorateur > XYZ Tiles > OpenStreetMap

Créer des groupes de couches

- Dans le panneau des couches > clic droit > Ajouter un groupe

Il peut être utile de créer des groupes de couches dans le cas où certaines couches sont disponibles dans plusieurs types géométriques : exemple pour la couche Sentiers qui peut contenir des lignes et des points



Changer le style d'une couche

- Clic droit sur la couche > Propriétés > Symbologie

Selon le type géométrique de la couche (point, ligne, polygone), il est possible de changer à volonté la couleur de remplissage, la couleur de contour, la taille ou l'épaisseur.

Dimensionner les colonnes de la table attributaire

Le fait de dimensionner la taille des colonnes dans la table attributaire permet une permet lisibilité des noms de champs et des informations contenues à l'intérieur :

- Clic droit sur la couche > Ouvrir la Table d'Attributs > clic droit sur une colonne > Taille autom pour toutes les colonnes

Certains champs texte peuvent être très larges (exemple _Description_ dans la couche **Zones sensibles**). Dans ce cas il est possible d'adapter manuellement la taille de la colonne : * Clic droit sur la couche > Ouvrir la Table d'Attributs > clic droit sur la colonne > Largeur > Entrer une largeur de colonne (exemple : 200)

Afficher le décompte des entités d'une couche

— Clic droit sur la couche > Afficher le nombre d'entités

Zoomer sur l'emprise d'une couche

— Clic droit sur la couche > Zoomer sur la(les) couche(s)

CHAPITRE 11

Installation

— *Ubuntu package*
  — *Requirements*
  — *Information to prepare before installation*
  — *Fresh installation*
  — *Extra steps*
  — *Uninstallation*
— *Docker*
  — *Management commands*

11.1 Ubuntu package

Use these instructions to install Geotrek-admin in an easy way on a dedicated Ubuntu Focal Fossa 20.04 LTS server for production. For another distributions, please use *the Docker installation method*. It requires more technical skills. Lastly, for a developer instance, please follow *the dedicated procedure*.

11.1.1 Requirements

Geotrek is mostly a CPU-bound application due to the complex queries including geometric operations (such as intersection) which are executed on the database. This is especially true in the setup with a Geotrek Rando v3 portal requesting dynamic geometric data through the Geotrek API.

In such a configuration the required system resources should be :

— 4 cores
— 8 Go RAM or more
— 50 Go disk space or more (20 Go + estimated size of attached files like photos, including elements imported from SIT)

If spreading the components on multiple hosts keep in mind the bottleneck will most likely be the CPU and RAM at the database server level.

Software requirements are :

— Ubuntu Focal Fossa 20.04 LTS. Server flavor is recommended but any other flavors work too (desktop…)

An Internet connection with open HTTP and HTTPS destination ports is required.

11.1.2 Information to prepare before installation

These information will be asked during the installation process and are the basic configuration of Geotrek-admin :

— The **domain name** or **IP** to use to access to **Geotrek-admin** web application.
— Rando server name : the **domain name** to use to access to **Geotrek-rando** website (optional, if appropriate).
— PostgreSQL **host, port, user, password and DB name** if you use an external DB server.
— The **SRID** of the projection to use to store geometries. The projection must match your geographic area and coordinates must be in meters.
— The list of **languages** into which translation of contents will be made
— The name or acronym of your **organization**

11.1.3 Fresh installation

Run the following command in a shell prompt on your server :

```
curl https://raw.githubusercontent.com/GeotrekCE/Geotrek-admin/master/tools/install.sh |␣
↪bash
```

If you don't want to use a local database, you can run the following command instead. This will prevent the script to install PostgreSQL server locally. Don't forget to enable PostGIS extension in your remote database before installation.

```
curl https://raw.githubusercontent.com/GeotrekCE/Geotrek-admin/blob/master/tools/install.
↪sh | bash -s - --nodb
```

Then create the application administrator account and connect to the web interface.

```
sudo geotrek createsuperuser
```

If you are not confident with the `install.sh` script, or if you are having troubles, you can do the same operations by hand :

1. Add `deb https://packages.geotrek.fr/ubuntu bionic main` to APT sources list.
2. Add https://packages.geotrek.fr/geotrek.gpg.key to apt keyring.
3. Run `apt-get update`
4. If you want to use a local database, install PostGIS package (before installing Geotrek-admin, not at the same time). If not, you must create database and enable PostGIS extension before.
5. Install the Geotrek-admin package (`sudo apt install geotrek-admin`).

Note : Geotrek-admin is automatically installed in `/opt/geotrek-admin/` directory.

The installation automatically creates an internal `geotrek` linux user, owner of this directory

The Geotrek-admin Python application is located in `/opt/geotrek-admin/lib/python3.6/site-packages/geotrek` directory

---

11.1.4 Extra steps

We highly recommend installing an antivirus software to regularly scan uploaded files located under `/opt/geotrek-admin/var/media/`.

Here is the installation process for ClamAV :

```
apt install clamav
```

Prepare quarantine folder for suspicious files :

```
mkdir /var/lib/clamav/quarantine/
chmod 700 /var/lib/clamav/quarantine/
```

Configure ClamAV via cron, to scan the folder once a day, put suspicious files in quarantine, and raise email alerts, by creating file `/etc/cron.daily/clamscan` with the following content :

```
#!/bin/sh

nice -n 15 ionice -c 3 clamscan --recursive --allmatch --suppress-ok-results --no-
↪summary --infected --scan-mail=no --log=/var/log/clamav/scan-report.$(date -Iseconds) /
↪opt/geotrek-admin/var/media/ |mail -E -s "ClamAV report for $(hostname)" admin@example.
↪com

# Cleanup old files in quarantine (> 90 days)
find /var/lib/clamav/quarantine/ -type f -mtime +90 -delete

# Cleanup old scan reports (> 365 days)
find /var/log/clamav/ -type f -name "scan-report.*" -mtime +365 -delete
```

Make sure to change alert recepient (`admin@example.com` above) and make this cron file executable :

```
chmod 700 /etc/cron.daily/clamscan
```

If a suspicious file is put in quarantine, you will need to manually delete the corresponding attachment from Geotrek-Admin (since the file for this attachment has moved to the quarantine folder, it will no longer be found).

11.1.5 Uninstallation

Run :

```
apt-get remove geotrek-admin
```

Media files will be left in `/opt/geotrek-admin/var` directory. To remove them, run :

```
apt-get purge geotrek-admin
```

To remove dependencies (convertit, screamshooter…), run :

```
apt-get autoremove
```

Note : PostgreSQL and its database will not be removed by these commands. If need be, remove them manually.

11.2 Docker

Docker is an alternative installation method, recommended for experts only. It allows to install several instances of Geotrek-admin on the same serveur, and to install it on other distributions than Ubuntu Linux 18.04.

1. Install Docker and Docker Compose, either from your distribution or from upstream packages (cf. https://docs.docker.com/install/)
2. Download the code from https://github.com/GeotrekCE/Geotrek-admin/releases or checkout it with git from https://github.com/GeotrekCE/Geotrek-admin/
3. Unzip the tarball
4. Copy docker/install folder where you want
5. Edit *docker-compose.yml* to feed your needs if necessary
6. Copy *.env.dist* to *.env* and edit to feed your needs if necessary
7. Create user and database, enable PostGIS extension
8. Run *docker-compose run –rm web update.sh*
9. Run *docker-compose up*
10. Install NGINX (or equivalent) and add a configuration file (taking inspiration from *nginx.conf.in*)

11.2.1 Management commands

Replace `sudo geotrek ...` commands by `cd <install directory>; docker-compose run --rm web ./manage.py ...`

To load minimal data and create an application superuser, run :

```
docker-compose run --rm web load_data.sh
docker-compose run --rm web ./manage.py createsuperuser
```

CHAPITRE 12

Upgrade

— *From Geotrek-admin >= 2.33*
— *From Geotrek-admin <= 2.32*
— *From Geotrek-admin <= 2.69.0*
— *Server migration*
— *PostgreSQL*
  — *Update PostgreSQL / PostGIS on Ubuntu Bionic*

12.1 From Geotrek-admin >= 2.33

Beforehand you shoud update your system's catalog :

```
sudo apt-get update
```

If your current version is <= 2.40.1 you should run instead :

```
sudo apt-get update  --allow-releaseinfo-change
```

To display the installed version and the latest upgradeable version, run :

```
apt list --all-versions geotrek-admin
```

To upgrade only geotrek-admin and its dependencies, run :

```
sudo apt-get install geotrek-admin
```

To upgrade geotrek-admin to a **specific version**, run :

```
sudo apt-get install geotrek-admin=<version>
```

For instance :

```
sudo apt-get install geotrek-admin=2.97.4.ubuntu18.04
```

or

```
sudo apt-get install geotrek-admin=2.98.0.ubuntu20.04
```

Note : all package versions remain available. Even when not listed with `apt list`.

Once geotrek-admin has been upgraded you may want to prevent unwanted upgrade with the whole distribution, you can run :

```
sudo apt-mark hold geotrek-admin
```

12.2 From Geotrek-admin <= 2.32

First of all, make sure your current Geotrek-admin version works correctly. Especially, after an upgrade of the Ubuntu distribution, you will have to run `./install.sh` before proceeding with Geotrek-admin upgrade.

Then, go inside your existing Geotrek-admin installation directory and run the dedicated migration script :

```
curl https://raw.githubusercontent.com/GeotrekCE/Geotrek-admin/blob/master/tools/migrate.
↪sh | bash
```

Check if `SPATIAL_EXTENT` is well set in `/opt/geotrek-admin/var/conf/custom.py` (see Advanced configuration section)

Note : Geotrek-admin is now automatically installed in `/opt/geotrek-admin/` directory and the advanced configuration file moved to `/opt/geotrek-admin/var/conf/custom.py` (with spatial extent, map and modules configuration…).

See advanced configuration documentation for details.

The `etc/settings.ini` file is replaced by basic configuration, updated with `sudo dpkg-reconfigure geotrek-admin` command (database, SRID, languages, server_name, timeout…).

Update your imports, synchronization and backup commands and directories.

12.3 From Geotrek-admin <= 2.69.0

WARNING !

Starting from version 2.70.0, Geotrek now needs PostgreSQL extension “pgrypto”.

Make sure to run the following command **BEFORE** upgrading :

```
su postgres -c "psql -q -d $POSTGRES_DB -c 'CREATE EXTENSION pgcrypto;'"
```

12.4 Server migration

It is a new installation with an additional backup/restore and a file transfert in between. The commands below are examples to adapt to your actual configuration (server names, database configuration). These commands apply to versions >= 2.33. If your version is below 2.33, please check the doc of your version.

Backup settings, media files and database on the old server :

```
sudo -u postgres pg_dump -Fc geotrekdb > geotrekdb.backup
tar cvzf data.tgz geotrekdb.backup /opt/geotrek-admin/var/conf/ /opt/geotrek-admin/var/
↪media/
```

Restore files on the new server :

```
scp old_server_ip:path/to/data.tgz .
tar xvzf data.tgz
```

12.5 PostgreSQL

Geotrek-admin support PostgreSQL 10+ and PostGIS 2.5+ for now. In next release, Django 4.2 will drop PostgreSQL < 12 support. We recommend to upgrade to PostgreSQL 16 and PostGIS 3.4.

You can check your PostgreSQL version with the following command :

```
sudo geotrek check_versions --postgresql
```

If your PostgreSQL version is below 12, you should upgrade your PostgreSQL server. If you can not upgrade for the moment, check release notes before each Geotrek-admin upgrade to ensure compatibility. You will be able to mark hold your Geotrek-admin Ubuntu package to prevent unwanted upgrade.

```
sudo apt-mark hold geotrek-admin
```

In case of unwanted upgrade, you will be able to revert your Geotrek-admin version to last supporting PostgreSQL 10 with, for example :

```
sudo apt-get install geotrek-admin=2.102.1.ubuntu20.04
```

for Ubuntu 20.04, or

```
sudo apt-get install geotrek-admin=2.102.1.ubuntu18.04
```

for Ubuntu bionic

12.5.1 Update PostgreSQL / PostGIS on Ubuntu Bionic

Avertissement : Ubuntu Bionic is already deprecated. We recommend you to install PostgreSQL on a dedicated server, with a most recent version of Ubuntu. If possible, on the same host or datacenter than your Geotrek-admin instance. If you can't, you can follow these instructions to upgrade PostgreSQL and PostGIS on Ubuntu Bionic with official PostgreSQL APT archive repository. The ultimate version published for Bionic is PostgreSQL 14, supported until November 12, 2026.

```
sudo rm -f /etc/apt/sources.list.d/pgdg.list
sudo apt install curl ca-certificates
sudo install -d /usr/share/postgresql-common/pgdg
sudo curl -o /usr/share/postgresql-common/pgdg/apt.postgresql.org.asc --fail https://www.
↪postgresql.org/media/keys/ACCC4CF8.asc
sudo sh -c 'echo "deb [signed-by=/usr/share/postgresql-common/pgdg/apt.postgresql.org.
↪asc] https://apt-archive.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" > /
↪etc/apt/sources.list.d/pgdg.list'
sudo apt update
```

Then, make a database dump. You can see user / database / password in /opt/geotrek-admin/conf/env file.

```
sudo -u postgres pg_dump -Fc --no-acl --no-owner -d <your geotrek database name> > ./
↪backup.dump
```

Now, install newest version of PostgreSQL and PostGIS :

```
sudo apt install postgresql-14-postgis-3
```

Note : Installing many PostgreSQL versions on the same system will use another port than default 5432. You can check the newest port with `pg_lsclusters` command. For next lines, we consider new port is 5433.

Recreate user and database :

```
sudo -u postgres psql -p 5433
```

```
CREATE USER <your geotrek user> WITH ENCRYPTED PASSWORD '<your geotrek user password>';
CREATE DATABASE <your geotrek database> WITH OWNER <your geotrek user>;
\c <your geotrek database>
CREATE EXTENSION postgis;
CREATE EXTENSION postgis_raster;
CREATE EXTENSION pgcrypto;
\q
```

Avertissement : You should report configuration from /etc/postgresql/10/main/pg_hba.conf to /etc/postgresql/14/main/pg_hba.conf. Then restart your postgresql

```
sudo cp /etc/postgresql/10/main/pg_hba.conf /etc/postgresql/14/main/pg_hba.conf
sudo systemctl restart postgresql
```

You can now restore your database dump.

```
pg_restore -h 127.0.0.1 -p 5433 -U <your geotrek user> -d <your geotrek database> ./
↪backup.dump
```

Note : Note you have to use *-h 127.0.0.1* to connect with the *geotrek* user (this user cannot connect with the default unix socket). Connecting with *geotrek* is important for restored entities to have the right owner. Some errors can occurs, around extensions creation or spatial_ref_sys table content. This is normal. We already create these extensions on previous steps.

Avertissement : Any special configuration or tune setting in your *postgresql.conf* will not be reported, you should report configuration yourself in /etc/postgresql/14/postgresql.conf. Then restart your postgresql

```
sudo systemctl restart postgresql
```

Now, you can update your Geotrek-admin configuration to use the new PostgreSQL server, by changing its default port to the new one.

```
sudo dpkg-reconfigure geotrek-admin
```

And change `POSTGRES_PORT` to 5433

You can now upgrade your Geotrek-admin, and check that the right database is used.

Note : If you want to use default 5432 port, you should change it in *postgresql.conf*, restart postgresql service, and change it by reconfiguring Geotrek-admin.

```
sudo geotrek check_versions --postgresql
```

If it shows PostgreSQL 14, you can remove the old PostgreSQL version.

```
sudo apt remove --purge postgresql-10
sudo apt autoremove
```

CHAPITRE 13

Loading data

— *Prerequisites for your data*
  — *Layers*
  — *Core*
  — *Land*
  — *Extras*
— *Load MNT raster*

13.1 Prerequisites for your data

13.1.1 Layers

— WMTS protocol
— WebMercator Projection

13.1.2 Core

— Only LineString geometries
— Simple geometries
— Not overlapping

If possible :
— Connex graph
— Name column
— Data source
Formats : Shapefile or pure SQL dump (CREATE TABLE + INSERT)

13.1.3 Land

— Cities polygons (Shapefile or SQL, simple and valid Multi-Polygons)
— Districts (Shapefile ou SQL, simple and valid Multi-Polygons)
— Restricted Areas (Shapefile ou SQL, simple and valid Multi-Polygons)

13.1.4 Extras

— Languages list
— Structures list (and default one)

13.2 Load MNT raster

In QGIS, you can visualize your DEM, or merge several tiles together (in *Raster* > *Misc* > *Merge*).

Generate a GeoTIFF, and upload both files (`.tif` + `.tfw`) on the server. And use the Geotrek-admin command to load it into PostGIS :

```
sudo geotrek loaddem <PATH>/dem.tif
```

Note : This command makes use of *GDAL* and `raster2pgsql` internally. It therefore supports all GDAL raster input formats. You can list these formats with the command `raster2pgsql -G`.

Note : The elevation data of DEM must be integer values. If the elevation data are floating numbers, you can convert them in integer values with the Raster calculator processing of SAGA in QGis (Processing > Toolbox > SAGA > Raster calculus > Raster calculator) with formula parameter set to `int(a)`.

Note : If you only have a `.tif` file, you can generate the `.tfw` file with the command `gdal_translate -co "TFW=YES" in.tif out.tif`. It will generate a new `.tif` file with its `.tfw` metadata file.

Note : If you want to update the altimetry of the topologies you need to use the option : –update-altimery

CHAPITRE 14

Configuration

— *Basic configuration update*
— *Custom setting file*
— *NGINX configuration*
— *Activate SSL / HTTPS*
— *Mandatory settings*
  — *Spatial reference identifier*
  — *Default Structure*
  — *Dynamic segmentation*
  — *Translations*
  — *Spatial extents*
— *Users management*
— *Database users*

14.1 Basic configuration update

To update basic configuration (server name, database connection, languages, or set workers number or timeout), run :

```
sudo dpkg-reconfigure geotrek-admin
```

The basic configuration is stored in `/opt/geotrek-admin/var/conf/env` file, not to be changed manually. This file also contains the PostgreSQL authentification details, if you need to access your Geotrek-admin database.

14.2 Custom setting file

Geotrek-admin advanced configuration is done in `/opt/geotrek-admin/var/conf/custom.py` file.

The list of all overridable setting and default values can be found there.

After any change in `custom.py`, run :

```
sudo service geotrek restart
```

Sometimes you also have to run :

```
sudo dpkg-reconfigure -u geotrek-admin
```

Note : Don't override the `os.getenv()` settings as they are managed with Basic configuration.

14.3 NGINX configuration

NGINX configuration is controlled by Geotrek-admin and will be erased at each upgrade. Do not modify `/etc/nginx/sites-available/geotrek.conf` or `/etc/nginx/sites-enable/geotrek.conf`. Modify `/opt/geotrek-admin/var/conf/nginx.conf.in` instead. To update `nginx.conf`, then run :

```
sudo dpkg-reconfigure geotrek-admin
```

14.4 Activate SSL / HTTPS

To activate https, you need firstly to change `custom.py` and add :

```
SESSION_COOKIE_SECURE = True
CSRF_COOKIE_SECURE = True
```

After this, edit `nginx.conf.in` to add your certificate.

If you generate it with letsencrypt : You can use certbot to add the certificate in your configuration. But you will have to move the configuration automatically added into `nginx.conf`, to the file `nginx.conf.in` in `/opt/geotrek-admin/var/conf/` directory

You have to move the configuration to the file `nginx.conf.in` because `nginx.conf` is automatically changed during command `dpkg-reconfigure geotrek-admin`.

14.5 Mandatory settings

14.5.1 Spatial reference identifier

```
SRID = 2154
```

Spatial reference identifier of your database. Default 2154 is RGF93 / Lambert-93 - France

It should not be change after any creation of geometries.

Choose wisely with epsg.io for example

14.5.2 Default Structure

```
DEFAULT_STRUCTURE_NAME = "GEOTEAM"
```

Name for your default structure.

This one can be changed, except it's tricky.
— *First change the name in the admin (authent/structure),*
— *Stop your instance admin.*
— *Change in the settings*
— *Re-run the server.*

14.5.3 Dynamic segmentation

```
TREKKING_TOPOLOGY_ENABLED = True
```

Use dynamic segmentation or not.

Dynamic segmentation is used by default when installing Geotrek-admin.

With this mode, linear objects are built and stored related to paths.

Without this mode, linear geometry of objects is built and stored as an independent geographic object without relation to paths.

So if you want to use Geotrek-admin without dynamic segmentation, set TREKKING_TOPOLOGY_ENABLED to false after installation.

Do not change it again to true after setting it to false.

14.5.4 Translations

```
LANGUAGE_CODE = 'fr'
```

Language of your interface.

```
MODELTRANSLATION_LANGUAGES = ('en', 'fr', 'it', 'es')
```

Languages of your project. It will be used to generate fields for translations. (ex : description_fr, description_en)

You won't be able to change it easily, avoid to add any languages and do not remove any.

14.5.5 Spatial extents

Boundingbox of your project : x minimum , y minimum , x max, y max :

```
    4 ^
      |
1     |     3
<-----+----->
      |
      |
    2 v
```

Default values :

```
SPATIAL_EXTENT = (105000, 6150000, 1100000, 7150000)
```

If you extend spatial extent, don't forget to load a new DEM that covers all the zone. If you shrink spatial extent, be sure there is no element in the removed zone or you will no more be able to see and edit it.

In order to check your configuration of spatial extents, a small tool is available at `http://<server_url>/tools/extents/`.

Note : Administrator privileges are required.

14.6 Users management

See *user management section in usage*.

14.7 Database users

It is not safe to use the `geotrek` user in QGIS, or to give its password to many collaborators.

A wise approach, is to create a *read-only* user, or with specific permissions.

With *pgAdmin*, you can create database users like this :

```
CREATE ROLE lecteur LOGIN;
ALTER USER lecteur PASSWORD 'passfacile';
GRANT CONNECT ON DATABASE geotrekdb TO lecteur;
```

And give them permissions by schema :

```
GRANT USAGE ON SCHEMA public TO lecteur;
GRANT USAGE ON SCHEMA geotrek TO lecteur;
GRANT SELECT ON ALL TABLES IN SCHEMA public TO lecteur;
GRANT SELECT ON ALL TABLES IN SCHEMA geotrek TO lecteur;
```

You can also create groups, etc. See PostgreSQL documentation.

CHAPITRE 15

Configuration avancée

- *Application settings*
  - *Spatial extents*
  - *Email settings*
  - *API*
  - *Swagger API documentation*
  - *Share services between several Geotrek instances*
  - *Control number of workers and request timeouts*
  - *External authent*
  - *Custom SQL*
- *Map settings*
  - *Change or add WMTS tiles layers (IGN, OSM, Mapbox…)*
  - *Map layers zoom*
  - *Map layers colors and style*
  - *External raster layers*
  - *Geographical CRUD*
  - *Disable darker map backgrounds*
  - *Map screenshots*
- *Modules and components*
  - *Enable Apps*
  - *Paths*
  - *Signage and Blade*
  - *POI*
  - *Diving*
  - *Land*
  - *Outdoor*
  - *Sensitive areas*
- *Feedback reports settings*
  - *Send acknowledge email*
  - *Suricate support*
  - *Display reports with status defined colors*

- *Use timers to receive alerts for your reports*
- *Anonymize feedback reports*
- *Attachments*
  - *View attachments in the browser*
  - *Resizing uploaded pictures*
  - *Prohibit usage of big pictures and small width / height*
  - *Prohibit usage of certain file types*
- *Interface*
  - *Configure columns displayed in lists views and exports*
  - *Configure form fields in creation views*
  - *Configure form fields required or needed for review or publication*
- *Edition*
  - *WYSIWYG editor configuration*
  - *Max characters count*
  - *Copyright on pictures*
  - *Facebook configuration*
  - *Override translations*
  - *Override public PDF templates*
  - *PDF as booklet*
  - *Custom font in public document template*
  - *Custom colors in public document template*
  - *Primary color in PDF templates*
  - *Custom logos*
- *Settings for Geotrek-rando*
  - *Synchro Geotrek-rando*
  - *Distances*
  - *Limits*
  - *Categories*
- *Settings for Geotrek-mobile*

15.1 Application settings

15.1.1 Spatial extents

In order to check your configuration of spatial extents, a small tool is available at http://server/tools/extents/.

Note : Administrator privileges are required.

15.1.2 Email settings

Geotrek-admin will send emails :

- to administrators when internal errors occur
- to managers when a feedback report is created

Email configuration takes place in `/opt/geotrek-admin/var/conf/custom.py`, where you control recipients emails (`ADMINS`, `MANAGERS`) and email server configuration.

Set configuration settings in `geotrek/settings/custom.py.dist` template file.

You can test your configuration with the following command. A fake email will be sent to the managers :

```
sudo geotrek sendtestemail --managers
```

15.1.3 API

API_IS_PUBLIC

Set to `True` if you want the API V2 to be available for everyone without authentication.

Example :

```
API_IS_PUBLIC = True
```

Default :

```
False
```

Note :

— This API provides access to promotion content (Treks, POIs, Touristic Contents …).
— Set to `False` if Geotrek is intended to be used only for managing content and not promoting them.
— This setting does not impact the Path endpoints, which means that the Paths informations will always need authentication to be display in the API, regardless of this setting.

15.1.4 Swagger API documentation

INSTALLED_APPS for API V2

In order to enable swagger module to auto-document API, in the custom settings file, add the following code :

Enable API V2 documentation :

```
INSTALLED_APPS += ('drf_yasg', )
```

Then run `sudo dpkg-reconfigure -u geotrek-admin`. The API swagger documentation is now availaible here : `<GEOTREK_ADMIN_URL>/api/v2`

15.1.5 Share services between several Geotrek instances

As explained *in the design section*, *Geotrek-admin* relies on several services. They are generic and reusable, and can thus be shared between several instances, in order to save system resources for example.

A simple way to achieve this is to install one instance with everything as usual (*standalone*), and plug the other instances on its underlying services.

Capture and conversion

If you want to use external services, in `.env`, add following variables :

```
CAPTURE_HOST=x.x.x.x
CAPTURE_PORT=XX
CONVERSION_HOST=x.x.x.x
CONVERSION_PORT=XX
```

Then, you can delete all screamshotter and convertit references in `docker-compose.yml`.

Shutdown useless services

Now that your instances point the shared server. You can shutdown the useless services on each instance.

Start by stopping everything :

```
sudo systemctl stop geotrek
```

15.1.6 Control number of workers and request timeouts

By default, the application runs on 4 processes, and timeouts after 30 seconds.

To control those values, edit and fix your `docker-compose.yml` file in web and api section.

To know how many workers you should set, please refer to gunicorn documentation.

15.1.7 External authent

You can authenticate user against a remote database table or view.

To enable this feature, fill these fields in `/opt/geotrek-admin/var/conf/custom.py` :

```
AUTHENT_DATABASE = 'authent'
DATABASES['authent'] = {
    'ENGINE': 'django.contrib.gis.db.backends.postgis',
    'NAME': '<database name>',
    'USER': '<user name>',
    'PASSWORD': '<password>',
    'HOST': '<host>',
    'PORT': '<port>',
}
AUTHENT_TABLENAME = '<table name>'
AUTHENTICATION_BACKENDS = ['geotrek.authent.backend.DatabaseBackend']
```

Expected columns in table/view are :

- `username` : string (*unique*)
- `first_name` : string
- `last_name` : string
- `password` : string (simple md5 encoded, or full hashed and salted password)
- `email` : string
- `level` : integer (1 : readonly, 2 : redactor, 3 : path manager, 4 : trekking manager, 5 : management and trekking editor, 6 : administrator)
- `structure` : string

— `lang` : string (language code)

Note :

— The schema used in `AUTHENT_TABLENAME` must be in the user search_path (`ALTER USER $geotrek_db_user SET search_path=public,userschema;`)
— User management will be disabled from Administration backoffice.
— In order to disable remote login, just comment *AUTHENTICATION_BACKENDS* line in settings file, and restart instance (see paragraph above).
— Geotrek-admin can support many types of users authentication (LDAP, oauth, ...), contact us for more details.

15.1.8 Custom SQL

Put your custom SQL in a file name `/opt/geotrek-admin/var/conf/extra_sql/<app name>/<pre or post>_<script name>.sql`

— app name is the name of the Django application, eg. trekking or tourism
— `pre_...` scripts are executed before Django migrations and `post_...` scripts after
— script are executed in INSTALLED_APPS order, then by alphabetical order of script names

15.2 Map settings

15.2.1 Change or add WMTS tiles layers (IGN, OSM, Mapbox...)

By default, you have two basemaps layers in your Geotrek-admin (OSM and OpenTopoMap)

You can change or add more basemaps layers like this :

`LEAFLET_CONFIG['TILES']`

Specify the tiles URLs this way in your custom Django setting file :

Syntax :

```
LEAFLET_CONFIG['TILES'] = [('NAME_OF_TILE', 'URL', 'COPYRIGHT'), ...]
```

Basic example :

```
LEAFLET_CONFIG['TILES'] = [
('OSM', 'http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', '©␣
↪OpenStreetMap Contributors'),
('OpenTopoMap', 'http://a.tile.opentopomap.org/{z}/{x}/{y}.png', 'Map data:␣
↪© OpenStreetMap contributors, SRTM | Map style: © OpenTopoMap (CC-BY-SA)
↪'),
]
```

Example with IGN and OSM basemaps :

```
LEAFLET_CONFIG['TILES'] = [
(
    'IGN Plan V2',
    '//data.geopf.fr/wmts?SERVICE=WMTS&REQUEST=GetTile&VERSION=1.0.0&
↪LAYER=GEOGRAPHICALGRIDSYSTEMS.PLANIGNV2&STYLE=normal&FORMAT=image/png&
↪TILEMATRIXSET=PM&TILEMATRIX={z}&TILEROW={y}&TILECOL={x}',
    {
```

(suite sur la page suivante)

(suite de la page précédente)

```
        'attribution': 'Plan IGNV2 - Carte © IGN/Geoportail',
        'maxNativeZoom': 16,
        'maxZoom': 22
    }
),
(
    'IGN Orthophotos',
    '//data.geopf.fr/wmts?SERVICE=WMTS&REQUEST=GetTile&VERSION=1.0.0&
↪LAYER=ORTHOIMAGERY.ORTHOPHOTOS&STYLE=normal&FORMAT=image/jpeg&
↪TILEMATRIXSET=PM&TILEMATRIX={z}&TILEROW={y}&TILECOL={x}',
    {
        'attribution': 'Orthophotos - Carte © IGN/Geoportail',
        'maxNativeZoom': 19,
        'maxZoom': 22
    }
),
(
    'OpenStreetMap',
    '//{s}.tile.openstreetmap.org/{z}/{x}/{y}.png',
    {
        'attribution': '&copy; <a href="https://www.openstreetmap.org/
↪copyright">Contributeurs d\'OpenStreetMap</a>',
        'maxNativeZoom': 19,
        'maxZoom': 22
    }
),
(
    'OpenTopoMap',
    '//{s}.tile.opentopomap.org/{z}/{x}/{y}.png',
    {
        'attribution': 'map data: © <a href="https://openstreetmap.org/
↪copyright">OpenStreetMap</a> contributors, <a href="http://
↪viewfinderpanoramas.org">SRTM</a> | map style: © <a href="https://
↪opentopomap.org">OpenTopoMap</a> (<a href="https://creativecommons.org/
↪licenses/by-sa/3.0/">CC-BY-SA</a>)',
        'maxNativeZoom': 17,
        'maxZoom': 22
    }
),
(
    'IGN Scan 25',
    '//data.geopf.fr/private/wmts?apikey=ign_scan_ws&
↪LAYER=GEOGRAPHICALGRIDSYSTEMS.MAPS&EXCEPTIONS=text/xml&FORMAT=image/jpeg&
↪SERVICE=WMTS&VERSION=1.0.0&REQUEST=GetTile&STYLE=normal&TILEMATRIXSET=PM&
↪TILEMATRIX={z}&TILEROW={y}&TILECOL={x}',
    {
        'attribution': 'Plan Scan 25 Touristique - Carte © IGN/Geoportail',
        'maxNativeZoom': 17,
        'maxZoom': 22
    }
),
]
```

You can also configure overlays layers like this :

```
LEAFLET_CONFIG['OVERLAYS'] = [
(
    'IGN Cadastre',
    '//data.geopf.fr/wmts?SERVICE=WMTS&REQUEST=GetTile&VERSION=1.0.0&
↪LAYER=CADASTRALPARCELS.PARCELLAIRE_EXPRESS&STYLE=normal&FORMAT=image/png&
↪TILEMATRIXSET=PM&TILEMATRIX={z}&TILEROW={y}&TILECOL={x}',
    {
        'attribution': 'Cadastre - Carte © IGN/Geoportail',
        'maxNativeZoom': 19,
        'maxZoom': 22
    }
),
]
```

Note : To use some IGN Geoportail WMTS tiles (Scan25, Scan100, etc.), you may need an API key. You can find more information about this on https://geoservices.ign.fr/services-geoplateforme-diffusion.

15.2.2 Map layers zoom

LEAFLET_CONFIG

You can define the max_zoom the user can zoom for all tiles.

Example :

```
LEAFLET_CONFIG= 19
```

Note : It can be interesting when your tiles can't go to a zoom. For example OpenTopoMap is 17.

15.2.3 Map layers colors and style

MAPENTITY_CONFIG for layers color and style

All layers colors can be customized from the settings. See Leaflet reference for vectorial layer style.

Example :

```
MAPENTITY_CONFIG['MAP_STYLES']['path'] = {'color': 'red', 'weight': 5}
```

Example with one parameter :

```
MAPENTITY_CONFIG['MAP_STYLES']['city']['opacity'] = 0.8
```

Note : It can be interesting when your tiles can't go to a zoom. For example OpenTopoMap is 17.

COLORS_POOL

Regarding colors that depend from database content, such as land layers (physical types, work management...) or restricted areas. We use a specific setting that receives a list of colors :

Example :

```
COLORS_POOL['restrictedarea'] = ['#ff00ff', 'red', '#ddddd'...]
```

See the default values in `geotrek/settings/base.py` for the complete list of available styles.

```
MAPENTITY_CONFIG['MAP_STYLES'] = {
    'path': {'weight': 2, 'opacity': 1.0, 'color': '#FF4800'},
    'draftpath': {'weight': 5, 'opacity': 1, 'color': 'yellow', 'dashArray': '8, 8'},
    'city': {'weight': 4, 'color': 'orange', 'opacity': 0.3, 'fillOpacity': 0.0},
    'district': {'weight': 6, 'color': 'orange', 'opacity': 0.3, 'fillOpacity': 0.0,
↪'dashArray': '12, 12'},
    'restrictedarea': {'weight': 2, 'color': 'red', 'opacity': 0.5, 'fillOpacity': 0.5},
    'land': {'weight': 4, 'color': 'red', 'opacity': 1.0},
    'physical': {'weight': 6, 'color': 'red', 'opacity': 1.0},
    'competence': {'weight': 4, 'color': 'red', 'opacity': 1.0},
    'workmanagement': {'weight': 4, 'color': 'red', 'opacity': 1.0},
    'signagemanagement': {'weight': 5, 'color': 'red', 'opacity': 1.0},
    'print': {'path': {'weight': 1},
              'trek': {'color': '#FF3300', 'weight': 7, 'opacity': 0.5,
                       'arrowColor': 'black', 'arrowSize': 10},}
}
```

Color of the different layers on the map :

```
COLORS_POOL = {'land': ['#f37e79', '#7998f3', '#bbf379', '#f379df', '#f3bf79', '#9c79f3',
↪ '#7af379'],
               'physical': ['#f3799d', '#79c1f3', '#e4f379', '#de79f3', '#79f3ba', '
↪#f39779', '#797ff3'],
               'circulation': ['#f3799d', '#79c1f3', '#e4f379', '#de79f3', '#79f3ba', '
↪#f39779', '#797ff3'],
               'competence': ['#a2f379', '#f379c6', '#79e9f3', '#f3d979', '#b579f3', '
↪#79f392', '#f37984'],
               'signagemanagement': ['#79a8f3', '#cbf379', '#f379ee', '#79f3e3', '#79f3d3
↪'],
               'workmanagement': ['#79a8f3', '#cbf379', '#f379ee', '#79f3e3', '#79f3d3'],
               'restrictedarea': ['plum', 'violet', 'deeppink', 'orchid',
                                  'darkviolet', 'lightcoral', 'palevioletred',
                                  'MediumVioletRed', 'MediumOrchid', 'Magenta',
                                  'LightSalmon', 'HotPink', 'Fuchsia']}
```

Color of the different layers on the top right for landing.

Note :

— For land, physical, competence, signagemanagement, workmanagement should have 5 values.
— For restricted Area : add as many color as your number of restricted area type
— **Restart** the application for changes to take effect.

15.2.4 External raster layers

Astuce :

It is possible to add overlay tiles layer on maps. For example, it can be useful to :

- Get the cadastral parcels on top of satellite images
- Home made layers (*with Tilemill or QGisMapserver for example*).
- Like the park center borders, traffic maps, IGN BDTopo® or even the Geotrek paths that are marked as invisible in the database !

LEAFLET_CONFIG['OVERLAYS']

In `custom.py`, just add the following lines :

Example :

```
LEAFLET_CONFIG['OVERLAYS'] = [
('Cadastre', '//data.geopf.fr/wmts?SERVICE=WMTS&REQUEST=GetTile&VERSION=1.0.0&
↪LAYER=CADASTRALPARCELS.PARCELLAIRE_EXPRESS&STYLE=normal&FORMAT=image/png&
↪TILEMATRIXSET=PM&TILEMATRIX={z}&TILEROW={y}&TILECOL={x}', '&copy; Cadastre -␣
↪Carte © IGN/Geoportail')
('Coeur de parc', 'http://serveur/coeur-parc/{z}/{x}/{y}.png', '&copy; PNF'),
]
```

Expected properties :

For `GeoJSON` files, you can provide the following properties :

- `title` : string
- `description` : string
- `website` : string
- `phone` : string
- `pictures` : list of objects with `url` and `copyright` attributes
- `category` : object with `id` and `label` attributes

15.2.5 Geographical CRUD

PATH_SNAPPING_DISTANCE

Minimum distance to merge two paths in unit of SRID

Example :

```
PATH_SNAPPING_DISTANCE = 2.0
```

Note :

- Change the distance. Better to keep it like this.
- Not used when `TREKKING_TOPOLOGY_ENABLED = True`

SNAP_DISTANCE

Distance of snapping for the cursor in pixels on Leaflet map.

Example :

```
SNAP_DISTANCE = 30
```

PATH_MERGE_SNAPPING_DISTANCE

Minimum distance to merge two paths.

Example :

```
PATH_MERGE_SNAPPING_DISTANCE = 2
```

Note :

— Change the distance. Should be higher or the same as `PATH_SNAPPING_DISTANCE`.

— Used when `TREKKING_TOPOLOGY_ENABLED = True`.

TREK_POINTS_OF_REFERENCE_ENABLED

Points of reference are enabled on form of treks.

Example :

```
TREK_POINTS_OF_REFERENCE_ENABLED = True
```

Default :

```
False
```

OUTDOOR_COURSE_POINTS_OF_REFERENCE_ENABLED

Points of reference are enabled on form of otudoor courses.

Example :

```
OUTDOOR_COURSE_POINTS_OF_REFERENCE_ENABLED = True
```

Default :

```
False
```

TOPOLOGY_STATIC_OFFSETS

Land objects are added on other objects (path for example) with offset, avoiding overlay.

Example :

```
TOPOLOGY_STATIC_OFFSETS = {'land': -5, 'physical': 0, 'competence': 5,
↪'signagemanagement': -10, 'workmanagement': 10}
```

Example with more overlays :

```
TOPOLOGY_STATIC_OFFSETS = {'land': -7, 'physical': 0, 'competence': 7,
↪'signagemanagement': -14, 'workmanagement': 14}
```

Note : You should not change it to avoid overlay except if you want to have more overlays.

All settings used to generate altimetric profile :

```
ALTIMETRIC_PROFILE_PRECISION = 25  # Sampling precision in meters
ALTIMETRIC_PROFILE_AVERAGE = 2  # nb of points for altimetry moving average
ALTIMETRIC_PROFILE_STEP = 1  # Step min precision for positive / negative altimetry gain
ALTIMETRIC_PROFILE_BACKGROUND = 'white'
ALTIMETRIC_PROFILE_COLOR = '#F77E00'
```

(suite sur la page suivante)

(suite de la page précédente)

```
ALTIMETRIC_PROFILE_HEIGHT = 400
ALTIMETRIC_PROFILE_WIDTH = 800
ALTIMETRIC_PROFILE_FONTSIZE = 25
ALTIMETRIC_PROFILE_FONT = 'ubuntu'
ALTIMETRIC_PROFILE_MIN_YSCALE = 1200  # Minimum y scale (in meters)
ALTIMETRIC_AREA_MAX_RESOLUTION = 150  # Maximum number of points (by width/height)
ALTIMETRIC_AREA_MARGIN = 0.15
```

Note :
- All these settings can be modified but you need to check the result every time
- The only one modified most of the time is `ALTIMETRIC_PROFILE_COLOR`

15.2.6 Disable darker map backgrounds

`MAPENTITY_CONFIG for map background`

Since IGN map backgrounds are very dense and colourful, a dark opacity is applied. In order to disable, change this MapEntity setting :

Example :

```
MAPENTITY_CONFIG['MAP_BACKGROUND_FOGGED'] = False
```

Default :

```
True
```

15.2.7 Map screenshots

```
SHOW_SENSITIVE_AREAS_ON_MAP_SCREENSHOT = True
SHOW_POIS_ON_MAP_SCREENSHOT = True
SHOW_SERVICES_ON_MAP_SCREENSHOT = True
SHOW_SIGNAGES_ON_MAP_SCREENSHOT = True
SHOW_INFRASTRUCTURES_ON_MAP_SCREENSHOT = True
```

`MAP_CAPTURE_SIZE`

Show objects on maps of PDF

Example :

```
MAP_CAPTURE_SIZE = 800
```

Note :
- Size in pixels of the capture.
- Be careful with your pdfs.
- If you change this value, pdfs will be rendered differently

15.3 Modules and components

15.3.1 Enable Apps

In order to disable a full set of modules, in the custom settings file, add the following code :

```
# Disable infrastructure and maintenance
_INSTALLED_APPS = list(INSTALLED_APPS)
_INSTALLED_APPS.remove('geotrek.infrastructure')
_INSTALLED_APPS.remove('geotrek.maintenance')
INSTALLED_APPS = _INSTALLED_APPS
```

TRAIL_MODEL_ENABLED

In order to remove notion of trails.

Example :

```
TRAIL_MODEL_ENABLED = False
```

Default :

```
True
```

LANDEDGE_MODEL_ENABLED

In order to remove landedge model.

Example :

```
LANDEDGE_MODEL_ENABLED = False
```

Default :

```
True
```

In order to remove zoning combo-boxes on list map :

```
LAND_BBOX_CITIES_ENABLED = False
LAND_BBOX_DISTRICTS_ENABLED = False
LAND_BBOX_AREAS_ENABLED = False
```

TOURISM_ENABLED

In order to hide TouristicContents and TouristicEvents on menu.

Example :

```
TOURISM_ENABLED = False
```

Default :

```
True
```

FLATPAGES_ENABLED

In order to hide Flatpages on menu. Flatpages are used in Geotrek-rando.

Example :

```
FLATPAGES_ENABLED = False
```

Default :

```
True
```

ACCESSIBILITY_ATTACHMENTS_ENABLED

In order to hide the accessibility menu for attachments.
Example :

```
ACCESSIBILITY_ATTACHMENTS_ENABLED = False
```

Default :

```
True
```

Note :
— By doing so, some software upgrades may not be as smooth as usual.
— Never forget to mention this customization if you ask for community support.

15.3.2 Paths

ALLOW_PATH_DELETION_TOPOLOGY

If `False`, it forbids to delete a path when at least one topology is linked to this path.
Example :

```
ALLOW_PATH_DELETION_TOPOLOGY = True
```

Default :

```
False
```

ALERT_DRAFT

If `True`, it sends a message to managers (MANAGERS) whenever a path has been changed to draft.
Example :

```
ALERT_DRAFT = False
```

Default :

```
True
```

ALERT_REVIEW

If `True`, it sends a message to managers (MANAGERS) whenever an object which can be published has been changed to review mode.
Example :

```
ALERT_REVIEW = False
```

Default :

```
True
```

Note : Email configuration takes place in `/opt/geotrek-admin/var/conf/custom.py`, where you control recipients emails (`ADMINS`, `MANAGERS`) and email server configuration.

15.3.3 Signage and Blade

`BLADE_ENABLED` and `LINE_ENABLED` settings (default to `True`) allow to enable or disable blades and lines submodules.

`DIRECTION_ON_LINES_ENABLED` setting (default to `False`) allow to have the *direction* field on lines instead of blades.

BLADE_CODE_TYPE

Type of the blade code (string or integer)

Example :

```
BLADE_CODE_TYPE = INT
```

Note :

- It can be string or integer
- If you have an integer code : `int`
- If you have an string code : `str`

BLADE_CODE_FORMAT

Correspond to the format of blades. Show N3-1 for the blade 1 of the signage N3.

Example :

```
BLADE_CODE_FORMAT = "{signagecode}-{bladenumber}"
```

Note :

- If you want to change : move information under bracket
- You can also remove one element between bracket
- You can do for exemple : `"CD99.{signagecode}.{bladenumber}"`
- It will display : `CD99.XIDNZEIU.01 (first blade of XIDNZEIU)`
- `signagecode` is the code of the signage
- `bladenumber` is the number of the blade

LINE_CODE_FORMAT

Corresponds to the format used in export of lines. Used in csv of signage

Example :

```
LINE_CODE_FORMAT = "{signagecode}-{bladenumber}-{linenumber}"
```

Note :

- Similar with above
- You can do for example : `"CD99.{signagecode}-{bladenumber}.{linenumber}"`
- It will display : `CD99.XIDNZEIU-01.02` (second line of the first blade of XIDNZEIU)
- `signagecode` is the code of the signage
- `bladenumber` is the number of the blade
- `linenumber` is the number of the line

15.3.4 POI

TREK_POI_INTERSECTION_MARGIN

Buffer around treks to intersects POIs (works only without dynamic segmentation)
Example :

```
TREK_POI_INTERSECTION_MARGIN = 500  # meters
```

Default :

```
500
```

15.3.5 Diving

INSTALLED_APPS for Diving

In order to enable diving module, in the custom settings file, add the following code :
Example :

```
INSTALLED_APPS += ('geotrek.diving', )
```

Then run `sudo dpkg-reconfigure -pcritical geotrek-admin`.

You can also insert diving minimal data (default practices, difficulties, levels and group permissions values) :

```
sudo geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/diving/
↪fixtures/basic.json
cp /opt/geotrek-admin/lib/python*/site-packages/geotrek/diving/fixtures/upload/* /opt/
↪geotrek-admin/var/media/upload/
```

You can insert licenses of attachments with this command :

```
sudo geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/common/
↪fixtures/licenses.json
```

15.3.6 Land

You can insert circulation and authorization types with this command :

```
sudo geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/land/fixtures/
↪circulations.json
```

15.3.7 Outdoor

INSTALLED_APPS for Outdoor

In order to enable Outdoor module, in the custom settings file, add the following code :
Example :

```
INSTALLED_APPS += ('geotrek.outdoor', )
```

Then run `sudo dpkg-reconfigure -pcritical geotrek-admin`.

You can also insert Outdoor minimal data :

```
sudo geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/outdoor/
↪fixtures/basic.json
```

After installing Outdoor module, you have to add permissions to your user groups on outdoor sites and courses.

Note :

— Outdoor module is not compatible with PostGIS <= 2.4 that is included in Ubuntu 18.04.

— You should either upgrade to Ubuntu 20.04 or upgrade postGIS to 2.5 with https://launchpad.net/~ubuntugis/+archive/ubuntu/ppa

---

15.3.8 Sensitive areas

INSTALLED_APPS for Sensitive areas

In order to enable sensitivity module, in the custom settings file, add the following code :

Example :

```
INSTALLED_APPS += ('geotrek.sensitivity', )
```

You can insert rules of sensitive area with these commands :

```
sudo geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/sensitivity/
↪fixtures/rules.json
cp -r /opt/geotrek-admin/lib/python*/site-packages/geotrek/sensitivity/fixtures/upload/
↪rules/ /opt/geotrek-admin/var/media/upload/
```

Settings

The following settings are related to sensitive areas :

SENSITIVITY_DEFAULT_RADIUS

Default radius of sensitivity bubbles when not specified for species

Example :

```
SENSITIVITY_DEFAULT_RADIUS = 100  # meters
```

Default :

```
100
```

SENSITIVE_AREA_INTERSECTION_MARGIN

Buffer around treks to intersects sensitive areas

Example :

```
SENSITIVE_AREA_INTERSECTION_MARGIN = 500  # meters
```

Default :

```
500
```

Import from https ://biodiv-sports.fr

In user interface, in the top-right menu, clic on « Imports » and choose « Biodiv'Sports ».

On command line, run

```
sudo geotrek import geotrek.sensitivity.parsers.BiodivParser
```

Import from shapefile

In user interface, in the top-right menu, go to Imports and choose « Shapefile zone sensible espèce » or « Shapefile zone sensible réglementaire ».

Note : The file must be a zip containing all the shapefile extensions (.shp, .shx, .prj, .dbf, .cpg)

On command line, run :

```
sudo geotrek import geotrek.sensitivity.parsers.SpeciesSensitiveAreaShapeParser <file.
↪shp>
```

or :

```
sudo geotrek  import geotrek.sensitivity.parsers.RegulatorySensitiveAreaShapeParser
↪<file.shp>.
```

Attributes for « zones espèces sensibles » are :

- `espece` : species name. Mandatory. A species with this name must have been previously created.
- `contact` : contact (text or HTML format). Optional.
- `descriptio` : description (text or HTML format). Optional.

Attributes for « zones sensibles réglementaires » are :

- `name` : zone name.
- `contact` : contact (text or HTML format). Optional.
- `descriptio` : description (text or HTML format). Optional.
- `periode` : month numbers of zone occupation, separated by comas, without spaces (ex. « 6,7,8 » for june, july and august)
- `pratiques` : sport practices names, separated by comas, without spaces (ex. « Terrestre,Aérien »). A sport practice with this name must have been previously created.
- `url` : card url. Optional.

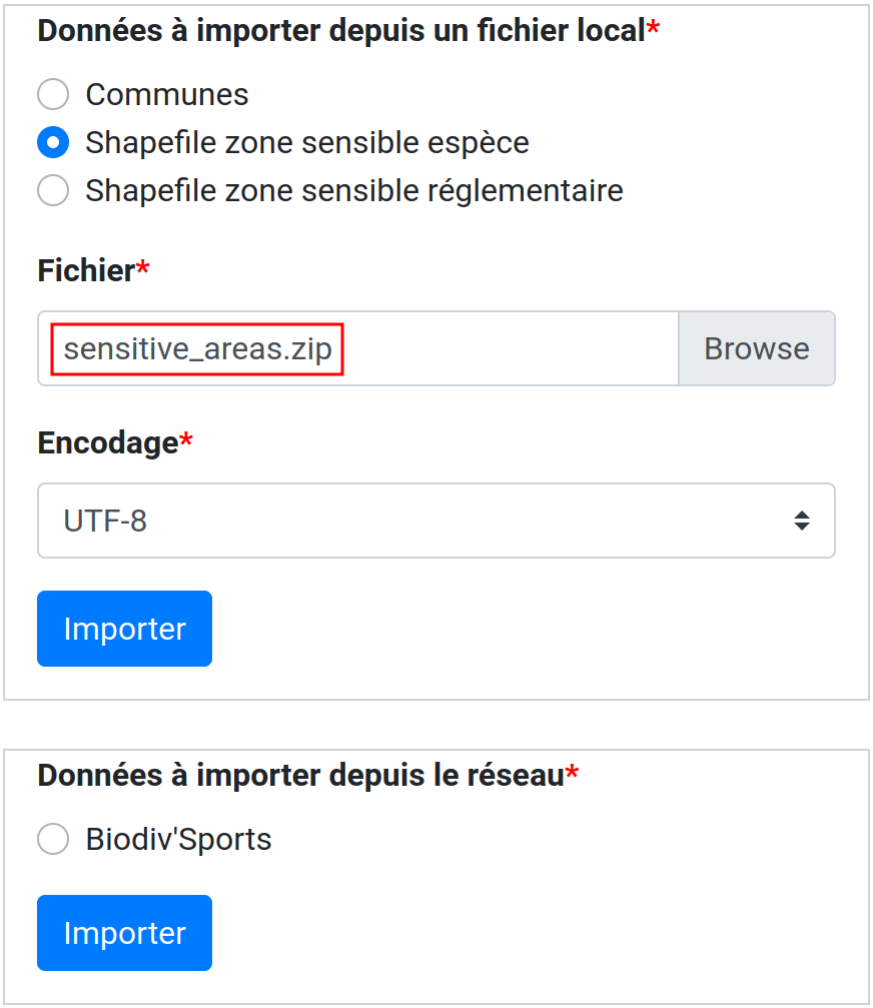


FIG. 1 – Import shapefile in user interface

Sync to Geotrek-rando

Just run :

```
sudo geotrek sync_rando <parameters>
```

If sensitivity module is enabled, sensitive areas will be automatically synchronized.

15.4 Feedback reports settings

15.4.1 Send acknowledge email

SEND_REPORT_ACK

If `False`, no email will be sent to the sender of any feedback on Geotrek-rando website.

Example :

```
SEND_REPORT_ACK = True
```

Default :

```
False
```

15.4.2 Suricate support

Suricate is the French national database gathering such reports. It exposes an API for external software to connect to. For Geotrek to connect to Suricate, you need to request two pairs of API keys allowing access.

Geotrek reports can work together with Suricate API, using one of three modes. Proceed through a mode full configuration before proceeding to the next mode.

1 - No Suricate (default)

This mode sends no report data to Suricate.

To initialize Report forms (Geotrek-admin, Geotrek-rando-V2, Geotrek-rando-V3) load lists for categories, activities, statuses and problem magnitude :

```
geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/feedback/fixtures/
↪basic.json
```

To make these lists available for your Geotrek-rando-V2, run `sync_rando` (see *synchronization*)

2 - Suricate Standard

This mode simply forwards all reports to Suricate, using the Standard API to post reports.

Set your account settings in `custom.py` :

```
SURICATE_REPORT_ENABLED = True

SURICATE_REPORT_SETTINGS = {
    'URL': '<Suricate Standard API Url>',
    'ID_ORIGIN': '<Suricate origin ID>',
    'PRIVATE_KEY_CLIENT_SERVER': '<your private key client / server>',
```

(suite sur la page suivante)

(suite de la page précédente)

```
    'PRIVATE_KEY_SERVER_CLIENT': '<your private key server / client>',
}
```

3 - Suricate Management (Workflow)

This mode allows to retrieve reports and related data directly from Suricate, using the Management API to get data. It is used to process and manage reports, using the Intervention module and following a predefined worklow, while sending all progress to Suricate. It implies enabling Suricate Report mode as well.

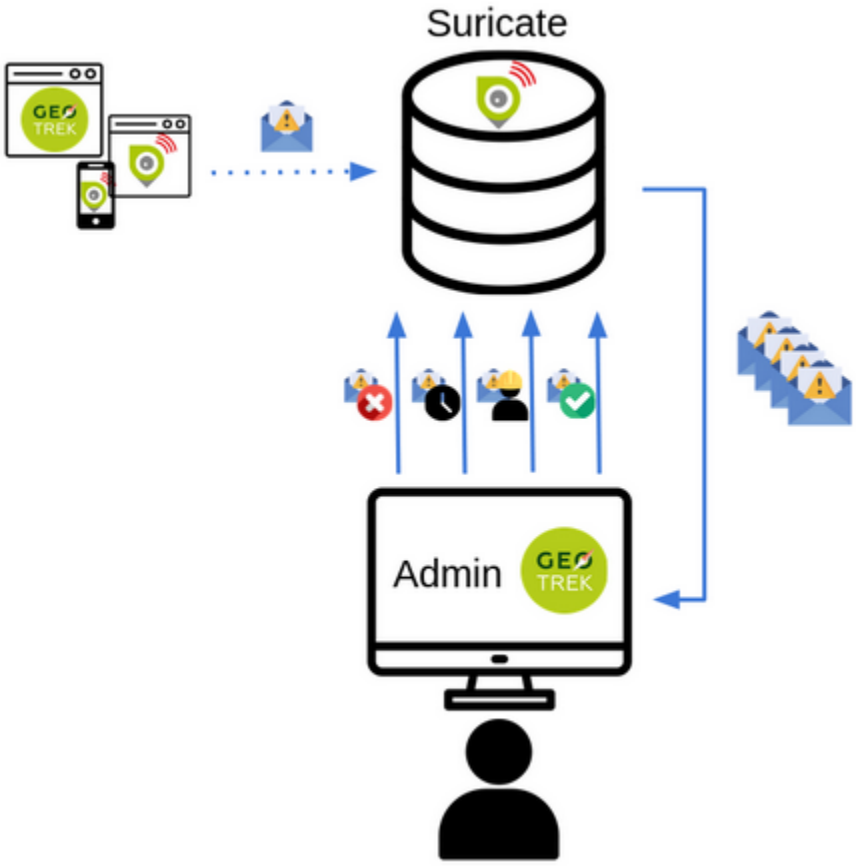


FIG. 2 – Suricate workflow

Suricate Workflow mode defines a strict process, composed of several steps representing the lifecycle of a user report, from creation to closing. A report is always characterized with a status, depicting how far in the process the report is, and displayed using a specific color on the map.

Reports

A report consists of the following information :

- A GPS position
- A message describing the problem
- A category : environment, security, usage conflit, signages
- A magnitude : usage is possible, difficult, or impossible
- A practice : trekking, cycling, horse-riding…
- Up to three pictures

Stakeholders and responsibility

This workflow defines three stakeholders categories :

- The sentinel : the person who submitted the report. They do not have a Geotrek user account nor intervene in the workflow, but they are kept updated on the processing of their report via semi-automatic e-mails.
- Supervisors : they are assigned (a) report(s) for treatment. They are tasked with planning an Intervention on Geotrek and enter information about it.
- The manager : they maintain a global view of all reports on the territory, assign reports to supervisors, handle messaging to the sentinel, and confirm reports resolution.

Any Geotrek user account can be used as a supervisor, as long as they have proper access and modification rights on both Report and Intervention modules. There can only be one Manager.

Report processing

Every night, Geotrek fetches new reports and updates through Suricate API. The manager receives an e-mail listing new reports (with "Filed" status). They can visualize them on Geotrek.

1 - Qualification

The manager has three options when handling a newly filed report :

- Classify : The report isn't relevant. The manager sets the report to "Classified" and enters a message for the sentinel, explaining their choice. The report is considered closed.
- Reject treatment : The report does not involve an area or an equipment under responsibility of the workflow users, but could be handled by another member of the Suricate community. The report is excluded from Geotrek workflow but is still accessible to the community via other applications using Suricate API.
- Assignation : The manager selects a supervisor from a drop-down selector, and enters a message with instructions or further information. The supervisor receives an e-mail notifying them about the newly assigned report, along with the manager's message. * The manager also enters a message destined to the sentinel, to notify them that the report is about to be handled. The report is set to status "Waiting". Only after assignation can we proceed to the following steps.

2 - Planification

The supervisor logs onto Geotrek and creates an Intervention linked to the assigned report, with a planification date. The intervention has status "Plannified". If too many days have passed between report assignation and intervention creation, the report is automatically set to "Late intervention" status, marked with color red, and the supervisor receives a reminder by e-mail.

3 - Resolution

The supervisor sets their intervention to "Resolved" status. The manager receives an e-mail notifying that a report is ready to be closed. If too many days have passed between intervention creation and intervention resolution, the report is automatically set to "Late resolution" status, marked with color red, and the supervisor receives a reminder e-mail.

4 - Closing

Following the intervention's resolution, the manager has to confirm the report was handled and sets it to "Resolved". They enter a message for the sentinel to inform them that the report's processing is over. The report is considered closed.

5 - GPS relocalisation

At any point, the manager or the supervisor can re-define GPS location for the report. Relocating it outside of the district marked as workflow responsibility area causes the treatment to be rejected (see part 1 Qualification). Furthermore, it is now possible to display the report layer on other Geotrek modules, for instance to compare positions between reports and signages or treks.

6 - Reports visibility

When a supervisor logs in to Geotrek, they can only see reports that are currently assigned to them. Both the manager and administrators can see all existing reports.

7 - Predefined messages

As we have seen above, the manager enters messages destined to the sentinel or to supervisors. These messages can be predefined in the administration interface and picked from a drop-down selector, then modified before sending. It is possible to automatically retrieve in a message the intervention date and the username of the supervisor that handled it.

Workflow configuration

Even though the workflow is a strict process, the following items are customisable.

Through administration interface :

- Colors for each status
- Selecting the manager
- Selecting the workflow responsibility area
- Predefined messages

Through application configuration :

- API keys to connect to Suricate
- Enabling of Workflow mode or any other mode
- Enabling/disabling status colors on map
- Duration of timers setting reports to “late” statuses

Synchronization and network losses

Communication between Suricate and Geotrek operates as follows :

- Suricate to Geotrek : new information is fetched once a night
- Geotrek to Suricate : every report update on Geotrek is immediately forwarded to Suricate

Maintaining synchronization between Suricate and Geotrek confronts us to the challenges of distributed software architecture. At any point, the connection between both applications can be lost, meaning that Suricate and Geotrek will no longer agree on a report’s status. Most of the time, this is simply due to temporary network failure. A system is in place to compensate for such failures. If a request to Suricate API fails, it is stored in the database and resent a few hours later. In case of a long term loss of connection, Django commands are available for an administrator to run some connection tests and resend stored information once connection is recovered.

For technical documentation refer to : https://geotrek.ecrins-parcnational.fr/ressources/technique/2023-02-Geotrek-Suricate-configuration.pdf

You can find the same detailled explanation on the workflow in this article in french : https://makina-corpus.com/geotrek/gestion-territoires-naturels-geotrek-traitement-signalements-suricate

- Set your settings in `custom.py` :

```
SURICATE_WORKFLOW_ENABLED = True

SURICATE_MANAGEMENT_SETTINGS = {
    'URL': '<Suricate Management API Url>',
    'ID_ORIGIN': '<Suricate origin ID>',
    'PRIVATE_KEY_CLIENT_SERVER': '<your private key client / server>',
    'PRIVATE_KEY_SERVER_CLIENT': '<your private key server / client>',
}

SURICATE_WORKFLOW_SETTINGS = {
    "SURICATE_RELOCATED_REPORT_MESSAGE": "This report is not located in Workflow␣
↪responsiblity area.",
    "SKIP_MANAGER_MODERATION": False
}
```

You can use the following command to test your connection settings :

```
geotrek sync_suricate -v 2 --connection-test
```

Load lists for activities and/or report statuses from Suricate :

```
geotrek sync_suricate --activities --statuses -v 2
```

Load alerts from Suricate (located in your bounding box) :

```
geotrek sync_suricate -v 2 --no-notification
```

To make these lists available for your Geotrek-rando, run `sync_rando` (see *synchronization*)

— Then load extra required statuses for Reports and Interventions :

```
geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/feedback/fixtures/
↪management_workflow.json
geotrek loaddata /opt/geotrek-admin/lib/python*/site-packages/geotrek/maintenance/
↪fixtures/basic.json
```

— Go to the configuration site and :

— if you want to include the moderation steps (`SKIP_MANAGER_MODERATION = False`), select a user as Workflow Manager (*/admin/feedback/workflowmanager/*). Their role is to assign reports to other users.

— select a district as Workflow District (*/admin/feedback/workflowdistrict/*). This zone defines the area of reponsibility for reports. Reports relocated outside of the district will be excluded from workflow.

— create predefined emails (*/admin/feedback/predefinedemail/*) to notify Suricate Sentinels and Administrators. You can use *##intervention_end_date##* and *##supervisor##* in the messages" body to automatically replace with the report's linked Intervention date and author. The Extended Username field will be dsiplayed (see User Profile under */admin/auth/user/*).

— Make sure Users involved in the workflow have proper permissions to create and update Reports and Interventions (*/admin/auth/user/*)

Note :

— Be aware that, when enabling Suricate Management mode, Suricate becomes the master database for reports. This means **reports created in Geotrek-admin will not be saved to the database, they will only be sent to Suricate**.

— Reports are only saved when synchronized back from Suricate, when the synchronization command is run.

Make sure to run these three commands daily to maintain synchronization and update reports (thanks to *cron* for instance) :

```
geotrek retry_failed_requests_and_mails
geotrek check_timers
geotrek sync_suricate
```

15.4.3 Display reports with status defined colors

ENABLE_REPORT_COLORS_PER_STATUS

Go to the Configuration site and select colors to display for each status (*/admin/feedback/reportstatus/*).

Example :

```
ENABLE_REPORT_COLORS_PER_STATUS = True
```

Default :

```
False
```

15.4.4 Use timers to receive alerts for your reports

Astuce :

— It is possible to enable receiving email alerts for reports that have remained in the same status for too long.
— For instance, I can create two report statuses « To program » with timer days set to 10 and « Programmed » with timer days set to 0.
— If a report has had status « To program » for 10 days, an email alert will be sent. If its status is changed to « Programmed » within these 10 days, this will cancel the alert.
— The email alert will be sent to the assigned user for this report, or to managers (setting *MANAGERS*) if there is no assigned user.

To enable the alerts :

— Go to the Configuration module and set « Timer days » to some integer other than 0 in relevant statuses (*/admin/feedback/reportstatus/*)
— Select the « Uses timers » checkbox on reports that you wish to receive alerts for (in report update form)
— Make sure to run this commands daily to send email alerts and clear obsolete timers (thanks to *cron* for instance) :

```
geotrek check_timers
```

15.4.5 Anonymize feedback reports

To be compliant to GDPR, you cannot keep personnal data infinitely, and should notice your users on how many time you keep their email.

A Django command is available to anonymize reports, by default older than 365 days.

```
geotrek erase_emails
```

Or if you want to erase emails for reports older than 90 days

```
geotrek erase_emails --days 90
```

15.5 Attachments

15.5.1 View attachments in the browser

MAPENTITY_CONFIG for medias

Attached files are downloaded by default by browser, with the following line, files will be opened in the browser :
Example :

```
MAPENTITY_CONFIG['SERVE_MEDIA_AS_ATTACHMENT'] = False
```

Default :

```
True
```

15.5.2 Resizing uploaded pictures

PAPERCLIP_RESIZE_ATTACHMENTS_ON_UPLOAD

Attached pictures can be resized at upload by enabling this parameter :
Example :

```
PAPERCLIP_RESIZE_ATTACHMENTS_ON_UPLOAD = True
```

Default :

```
False
```

These corresponding height/width parameters can be overriden to select resized image size :

```
PAPERCLIP_MAX_ATTACHMENT_WIDTH = 1280
PAPERCLIP_MAX_ATTACHMENT_HEIGHT = 1280
```

15.5.3 Prohibit usage of big pictures and small width / height

PAPERCLIP_MAX_BYTES_SIZE_IMAGE

If you want to prohibit the usage of heavy pictures :
Example :

```
PAPERCLIP_MAX_BYTES_SIZE_IMAGE = 50000 # Bytes
```

If you want to prohibit the usage of small pictures in pixels :

```
PAPERCLIP_MIN_IMAGE_UPLOAD_WIDTH = 100
PAPERCLIP_MIN_IMAGE_UPLOAD_HEIGHT = 100
```

These three settings will also not allow downloading images from the parsers.

15.5.4 Prohibit usage of certain file types

Paperclip will only accept attachment files matching a list of allowed extensions. Here is the default value for this setting, which you can extend if needed :

```
PAPERCLIP_ALLOWED_EXTENSIONS = [
    'jpeg',
    'jpg',
    'mp3',
    'mp4',
    'odt',
    'pdf',
    'png',
    'svg',
    'txt',
    'gif',
    'tiff',
    'tif',
    'docx',
```

(suite sur la page suivante)

(suite de la page précédente)

```
    'webp',
    'bmp',
    'flac',
    'mpeg',
    'doc',
    'ods',
    'gpx',
    'xls',
    'xlsx',
    'odg',
]
```

It will verify that the mimetype of the file matches the extension.

PAPERCLIP_EXTRA_ALLOWED_MIMETYPES

You can add extra allowed mimetypes for a given extension with the following syntax :
Example :

```
PAPERCLIP_EXTRA_ALLOWED_MIMETYPES['gpx'] = ['text/xml']
```

PAPERCLIP_ALLOWED_EXTENSIONS

You can also entirely deactivate these checks with the following :
Example :

```
PAPERCLIP_ALLOWED_EXTENSIONS = None
```

Note : These two settings will also not allow downloading images from the parsers.

15.6 Interface

15.6.1 Configure columns displayed in lists views and exports

For each module, use the following syntax to configure columns to display in the main table.

```
COLUMNS_LISTS['<module>_view'] = ['list', 'of', 'columns']
```

For each module, use the following syntax to configure columns to export as CSV or SHP.

```
COLUMNS_LISTS['<module>_export'] = ['list', 'of', 'columns']
```

Another setting exists to enable a more detailed export of jobs costs in the interventions module. When enabling this settings, interventions list exports will contain a new column for each job's total cost.

ENABLE_JOBS_COSTS_DETAILED_EXPORT

Enable a more detailed export
Example :

```
ENABLE_JOBS_COSTS_DETAILED_EXPORT = True
```

Default :

```
False
```

Custom columns available

A (nearly ?) exhaustive list of attributes available for display and export as columns in each module.

```
COLUMNS_LISTS["path_view"] = [
    "length_2d",
    "valid",
    "structure",
    "visible",
    "min_elevation",
    "max_elevation",
    "date_update",
    "date_insert",
    "stake",
    "networks",
    "comments",
    "departure",
    "arrival",
    "comfort",
    "source",
    "usages",
    "draft",
    "trails",
    "uuid",
]
COLUMNS_LISTS["trail_view"] = [
    "departure",
    "arrival",
    "category",
    "length",
    "structure",
    "min_elevation",
    "max_elevation",
    "date_update",
    "length_2d",
    "date_insert",
    "comments",
    "uuid",
]
COLUMNS_LISTS["landedge_view"] = [
    "eid",
    "min_elevation",
    "max_elevation",
    "date_update",
    "length_2d",
    "date_insert",
    "owner",
    "agreement",
    "uuid",
]
```

(suite sur la page suivante)

(suite de la page précédente)

```
COLUMNS_LISTS["circulationedge_view"] = [
    "eid",
    "min_elevation",
    "max_elevation",
    "date_update",
    "length_2d",
    "date_insert",
    "uuid",
]
COLUMNS_LISTS["physicaledge_view"] = [
    "eid",
    "date_insert",
    "date_update",
    "length",
    "length_2d",
    "min_elevation",
    "max_elevation",
    "uuid",
]
COLUMNS_LISTS["competenceedge_view"] = [
    "eid",
    "date_insert",
    "date_update",
    "length",
    "length_2d",
    "min_elevation",
    "max_elevation",
    "uuid",
]
COLUMNS_LISTS["signagemanagementedge_export"] = [
    "eid",
    "date_insert",
    "date_update",
    "length",
    "length_2d",
    "min_elevation",
    "max_elevation",
    "uuid",
    "provider"
]
COLUMNS_LISTS["workmanagementedge_export"] = [
    "eid",
    "date_insert",
    "date_update",
    "length",
    "length_2d",
    "min_elevation",
    "max_elevation",
    "uuid",
]
COLUMNS_LISTS["infrastructure_view"] = [
    "condition",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "cities",
    "structure",
    "type",
    "description",
    "accessibility",
    "date_update",
    "date_insert",
    "implantation_year",
    "usage_difficulty",
    "maintenance_difficulty",
    "published",
    "uuid",
    "eid",
    "provider",
    "access"
]
COLUMNS_LISTS["signage_view"] = [
    "code",
    "type",
    "condition",
    "structure",
    "description",
    "date_update",
    "date_insert",
    "implantation_year",
    "printed_elevation",
    "coordinates",
    "sealing",
    "access",
    "manager",
    "published",
    "uuid",
]
COLUMNS_LISTS["intervention_view"] = [
    "begin_date",
    "end_date",
    "type",
    "target",
    "status",
    "stake",
    "structure",
    "subcontracting",
    "status",
    "disorders",
    "length",
    "material_cost",
    "min_elevation",
    "max_elevation",
    "heliport_cost",
    "contractor_cost",
    "date_update",
    "date_insert",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "description",
]
COLUMNS_LISTS["project_view"] = [
    "structure",
    "begin_year",
    "end_year",
    "constraint",
    "global_cost",
    "type",
    "date_update",
    "domain",
    "contractors",
    "project_owner",
    "project_manager",
    "founders",
    "date_insert",
    "comments",
]
COLUMNS_LISTS["trek_view"] = [
    "structure",
    "departure",
    "arrival",
    "duration",
    "description_teaser",
    "description",
    "gear",
    "route",
    "difficulty",
    "ambiance",
    "access",
    "accessibility_infrastructure",
    "advised_parking",
    "parking_location",
    "public_transport",
    "themes",
    "practice",
    "min_elevation",
    "max_elevation",
    "length_2d",
    "date_update",
    "date_insert",
    "accessibilities",
    "accessibility_advice",
    "accessibility_covering",
    "accessibility_exposure",
    "accessibility_level",
    "accessibility_signage",
    "accessibility_slope",
    "accessibility_width",
    "ratings_description",
    "ratings",
    "points_reference",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "source",
    "reservation_system",
    "reservation_id",
    "portal",
    "uuid",
    "eid",
    "eid2",
    "provider"
]
COLUMNS_LISTS["poi_view"] = [
    "structure",
    "description",
    "type",
    "min_elevation",
    "date_update",
    "date_insert",
    "uuid",
]
COLUMNS_LISTS["service_view"] = [
    "structure",
    "min_elevation",
    "type",
    "length_2d",
    "date_update",
    "date_insert",
    "uuid",
]
COLUMNS_LISTS["dive_view"] = [
    "structure",
    "description_teaser",
    "description",
    "owner",
    "practice",
    "departure",
    "disabled_sport",
    "facilities",
    "difficulty",
    "levels",
    "depth",
    "advice",
    "themes",
    "source",
    "portal",
    "date_update",
    "date_insert",
]
COLUMNS_LISTS["touristic_content_view"] = [
    "structure",
    "description_teaser",
    "description",
    "category",
    "contact",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "email",
    "website",
    "practical_info",
    "accessibility",
    "label_accessibility",
    "type1",
    "type2",
    "source",
    "reservation_system",
    "reservation_id",
    "date_update",
    "date_insert",
    "uuid",
    "eid",
    "provider"
]
COLUMNS_LISTS["touristic_event_view"] = [
    "structure",
    "themes",
    "description_teaser",
    "description",
    "meeting_point",
    "start_time",
    "end_time",
    "duration",
    "begin_date",
    "contact",
    "email",
    "website",
    "end_date",
    "organizers",
    "speaker",
    "type",
    "accessibility",
    "capacity",
    "portal",
    "source",
    "practical_info",
    "target_audience",
    "booking",
    "date_update",
    "date_insert",
    "uuid",
    "eid",
    "provider",
    "bookable",
    "cancelled",
    "cancellation_reason"
    "place",
    'preparation_duration',
    'intervention_duration',
    'price'
```

(suite sur la page suivante)

(suite de la page précédente)

```
]
COLUMNS_LISTS["feedback_view"] = [
    "email",
    "comment",
    "activity",
    "category",
    "problem_magnitude",
    "status",
    "related_trek",
    "uuid",
    "eid",
    "external_eid",
    "locked",
    "origin"
    "date_update",
    "date_insert",
    "created_in_suricate",
    "last_updated_in_suricate",
    "assigned_user",
    "uses_timers"
]
COLUMNS_LISTS["sensitivity_view"] = [
    "structure",
    "species",
    "published",
    "publication_date",
    "contact",
    "pretty_period",
    "category",
    "pretty_practices",
    "description",
    "date_update",
    "date_insert",
]
COLUMNS_LISTS["outdoor_site_view"] = [
    "structure",
    "name",
    "practice",
    "description",
    "description_teaser",
    "ambiance",
    "advice",
    "accessibility",
    "period",
    "labels",
    "themes",
    "portal",
    "source",
    "information_desks",
    "web_links",
    "eid",
    "orientation",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "wind",
    "ratings",
    "managers",
    "type",
    "description",
    "description_teaser",
    "ambiance",
    "period",
    "orientation",
    "wind",
    "labels",
    "themes",
    "portal",
    "source",
    "managers",
    "min_elevation",
    "date_insert",
    "date_update",
    "uuid",
]
COLUMNS_LISTS["outdoor_course_view"] = [
    "structure",
    "name",
    "parent_sites",
    "description",
    "advice",
    "equipment",
    "accessibility",
    "eid",
    "height",
    "ratings",
    "gear",
    "duration"
    "ratings_description",
    "type",
    "points_reference",
    "uuid",
]
COLUMNS_LISTS["path_export"] = [
    "structure",
    "valid",
    "visible",
    "name",
    "comments",
    "departure",
    "arrival",
    "comfort",
    "source",
    "stake",
    "usages",
    "networks",
    "date_insert",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "date_update",
    "length_2d",
    "length",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["trail_export"] = [
    "structure",
    "name",
    "comments",
    "departure",
    "arrival",
    "category",
    "certifications",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["landedge_export"] = [
    "eid",
    "land_type",
    "owner",
    "agreement",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "length_2d",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["circulationedge_export"] = [
    "eid",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "circulation_type",
    "authorization_type",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "length_2d",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["physicaledge_export"] = [
    "eid",
    "physical_type",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "length_2d",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["competenceedge_export"] = [
    "eid",
    "organization",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "length_2d",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["signagemanagementedge_export"] = [
    "eid",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "organization",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "length_2d",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["workmanagementedge_export"] = [
    "eid",
    "organization",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "length_2d",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["infrastructure_export"] = [
    "name",
    "type",
    "condition",
    "access",
    "description",
    "accessibility",
    "implantation_year",
    "published",
    "publication_date",
    "structure",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "slope",
    "usage_difficulty",
    "maintenance_difficulty"
    "uuid",
    "eid",
    "provider"
]
COLUMNS_LISTS["signage_export"] = [
    "structure",
    "name",
    "code",
    "type",
    "condition",
    "description",
    "implantation_year",
    "published",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "lat_value",
    "lng_value",
    "printed_elevation",
    "sealing",
    "access",
    "manager",
    "length",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "uuid",
    "eid",
    "provider"
]
COLUMNS_LISTS["intervention_export"] = [
    "name",
    "begin_date",
    "end_date",
    "type",
    "target",
    "status",
    "stake",
    "disorders",
    "total_manday",
    "project",
    "subcontracting",
    "width",
    "height",
    "length",
    "area",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "structure",
    "description",
    "date_insert",
    "date_update",
    "material_cost",
    "heliport_cost",
    "contractor_cost",
    "total_cost_mandays",
    "total_cost",
    "cities",
    "districts",
    "areas",
    "length",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
]
COLUMNS_LISTS["project_export"] = [
    "structure",
    "name",
    "period",
    "type",
    "domain",
    "constraint",
    "global_cost",
    "interventions",
    "interventions_total_cost",
    "comments",
    "contractors",
    "project_owner",
    "project_manager",
    "founders",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
]
COLUMNS_LISTS["trek_export"] = [
    "eid",
    "eid2",
    "structure",
    "name",
    "departure",
    "arrival",
    "duration",
    "duration_pretty",
    "description",
    "description_teaser",
    "gear",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "networks",
    "advice",
    "ambiance",
    "difficulty",
    "information_desks",
    "themes",
    "practice",
    "accessibilities",
    "accessibility_advice",
    "accessibility_covering",
    "accessibility_exposure",
    "accessibility_level",
    "accessibility_signage",
    "accessibility_slope",
    "accessibility_width",
    "ratings_description",
    "ratings",
    "access",
    "route",
    "public_transport",
    "advised_parking",
    "web_links",
    "labels",
    "accessibility_infrastructure",
    "parking_location",
    "points_reference",
    "related",
    "children",
    "parents",
    "pois",
    "review",
    "published",
    "publication_date",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "source",
    "portal",
    "length_2d",
    "length",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
    "provider"
]
COLUMNS_LISTS["poi_export"] = [
    "structure",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "eid",
    "name",
    "type",
    "description",
    "treks",
    "review",
    "published",
    "publication_date",
    "structure",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "length",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["service_export"] = [
    "eid",
    "type",
    "length",
    "ascent",
    "descent",
    "min_elevation",
    "max_elevation",
    "slope",
    "uuid",
]
COLUMNS_LISTS["dive_export"] = [
    "eid",
    "structure",
    "name",
    "departure",
    "description",
    "description_teaser",
    "advice",
    "difficulty",
    "levels",
    "themes",
    "practice",
    "disabled_sport",
    "published",
    "publication_date",
    "date_insert",
    "date_update",
    "areas",
    "source",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "portal",
    "review",
    "uuid",
]
COLUMNS_LISTS["touristic_content_export"] = [
    "structure",
    "eid",
    "name",
    "category",
    "type1",
    "type2",
    "description_teaser",
    "description",
    "themes",
    "contact",
    "email",
    "website",
    "practical_info",
    "accessibility",
    "label_accessibility",
    "review",
    "published",
    "publication_date",
    "source",
    "portal",
    "date_insert",
    "date_update",
    "cities",
    "districts",
    "areas",
    "approved",
    "uuid",
    "provider"
]
COLUMNS_LISTS["touristic_event_export"] = [
    "structure",
    "eid",
    "name",
    "type",
    "description_teaser",
    "description",
    "themes",
    "begin_date",
    "end_date",
    "duration",
    "meeting_point",
    "start_time",
    "end_time",
    "contact",
    "email",
    "website",
    "organizers",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "speaker",
    "accessibility",
    "capacity",
    "booking",
    "target_audience",
    "practical_info",
    "date_insert",
    "date_update",
    "source",
    "portal",
    "review",
    "published",
    "publication_date",
    "cities",
    "districts",
    "areas",
    "approved",
    "uuid",
    "provider",
    "bookable",
    "cancelled",
    "cancellation_reason"
    "place",
    'preparation_duration',
    'intervention_duration',
    'price'
]
COLUMNS_LISTS["feedback_export"] = [
    "comment",
    "activity",
    "category",
    "problem_magnitude",
    "status",
    "related_trek",
    "uuid",
    "eid",
    "external_eid",
    "locked",
    "origin"
    "date_update",
    "date_insert",
    "created_in_suricate",
    "last_updated_in_suricate",
    "assigned_user",
    "uses_timers"
]
COLUMNS_LISTS["sensitivity_export"] = [
    "species",
    "published",
    "description",
    "contact",
    "pretty_period",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "pretty_practices",
]
COLUMNS_LISTS["outdoor_site_export"] = [
    "structure",
    "name",
    "practice",
    "description",
    "description_teaser",
    "ambiance",
    "advice",
    "accessibility",
    "period",
    "labels",
    "themes",
    "portal",
    "source",
    "information_desks",
    "web_links",
    "eid",
    "orientation",
    "wind",
    "ratings",
    "managers",
    "type",
    "description",
    "description_teaser",
    "ambiance",
    "period",
    "orientation",
    "wind",
    "labels",
    "themes",
    "portal",
    "source",
    "managers",
    "min_elevation",
    "date_insert",
    "date_update",
    "uuid",
]
COLUMNS_LISTS["outdoor_course_export"] = [
    "structure",
    "name",
    "parent_sites",
    "description",
    "advice",
    "equipment",
    "accessibility",
    "eid",
    "height",
    "ratings",
    "gear",
```

(suite sur la page suivante)

(suite de la page précédente)

```
    "duration"
    "ratings_description",
    "type",
    "points_reference",
    "uuid",
]
```

15.6.2 Configure form fields in creation views

HIDDEN_FORM_FIELDS

For each module, use the following syntax to configure fields to hide in the creation form.
Example :

```
HIDDEN_FORM_FIELDS['<module>'] = ['list', 'of', 'fields']
```

Hideable form fields

An exhaustive list of form fields hideable in each module.

```
HIDDEN_FORM_FIELDS["path"] = [
        "departure",
        "name",
        "stake",
        "comfort",
        "arrival",
        "comments",
        "source",
        "networks",
        "usages",
        "valid",
        "draft",
        "reverse_geom",
    ],
HIDDEN_FORM_FIELDS["trek"] = [
        "structure",
        "name",
        "review",
        "published",
        "labels",
        "departure",
        "arrival",
        "duration",
        "difficulty",
        "gear",
        "route",
        "ambiance",
        "access",
        "description_teaser",
        "description",
```

(suite sur la page suivante)

(suite de la page précédente)

```
        "points_reference",
        "accessibility_infrastructure",
        "advised_parking",
        "parking_location",
        "public_transport",
        "advice",
        "themes",
        "networks",
        "practice",
        "accessibilities",
        "accessibility_advice",
        "accessibility_covering",
        "accessibility_exposure",
        "accessibility_level",
        "accessibility_signage",
        "accessibility_slope",
        "accessibility_width",
        "web_links",
        "information_desks",
        "source",
        "portal",
        "children_trek",
        "eid",
        "eid2",
        "ratings",
        "ratings_description",
        "reservation_system",
        "reservation_id",
        "pois_excluded",
        "hidden_ordered_children",
    ],
HIDDEN_FORM_FIELDS["trail"] = [
        "departure",
        "arrival",
        "comments",
        "category",
    ],
HIDDEN_FORM_FIELDS["landedge"] = [
        "owner",
        "agreement"
    ],
HIDDEN_FORM_FIELDS["infrastructure"] = [
        "condition",
        "access",
        "description",
        "accessibility",
        "published",
        "implantation_year",
        "usage_difficulty",
        "maintenance_difficulty"
    ],
HIDDEN_FORM_FIELDS["signage"] = [
```

(suite sur la page suivante)

(suite de la page précédente)

```
        "condition",
        "description",
        "published",
        "implantation_year",
        "code",
        "printed_elevation",
        "manager",
        "sealing",
        "access"
    ],
HIDDEN_FORM_FIELDS["intervention"] = [
        "disorders",
        "description",
        "type",
        "subcontracting",
        "end_date",
        "length",
        "width",
        "height",
        "stake",
        "project",
        "material_cost",
        "heliport_cost",
        "contractor_cost",
    ],
HIDDEN_FORM_FIELDS["project"] = [
        "type",
        "domain",
        "end_year",
        "constraint",
        "global_cost",
        "comments",
        "project_owner",
        "project_manager",
        "contractors",
    ],
HIDDEN_FORM_FIELDS["site"] = [
        "parent",
        "review",
        "published",
        "practice",
        "description_teaser",
        "description",
        "ambiance",
        "advice",
        "period",
        "orientation",
        "wind",
        "labels",
        "themes",
        "information_desks",
        "web_links",
```

(suite sur la page suivante)

(suite de la page précédente)

```
        "portal",
        "source",
        "managers",
        "eid"
    ],
HIDDEN_FORM_FIELDS["course"] = [
        "review",
        "published",
        "description",
        "advice",
        "equipment",
        "accessibility",
        "height",
        "children_course",
        "eid",
        "gear",
        "duration"
        "ratings_description",
    ]
HIDDEN_FORM_FIELDS["poi"] = [
        "review",
        "published",
        "description",
        "eid",
    ],
HIDDEN_FORM_FIELDS["service"] = [
        "eid",
    ],
HIDDEN_FORM_FIELDS["dive"] = [
        "review",
        "published",
        "practice",
        "advice",
        "description_teaser",
        "description",
        "difficulty",
        "levels",
        "themes",
        "owner",
        "depth",
        "facilities",
        "departure",
        "disabled_sport",
        "source",
        "portal",
        "eid",
    ],
HIDDEN_FORM_FIELDS["touristic_content"] = [
        'label_accessibility'
        'type1',
        'type2',
        'review',
```

(suite sur la page suivante)

(suite de la page précédente)

```
        'published',
        'accessibility',
        'description_teaser',
        'description',
        'themes',
        'contact',
        'email',
        'website',
        'practical_info',
        'approved',
        'source',
        'portal',
        'eid',
        'reservation_system',
        'reservation_id'
    ],
HIDDEN_FORM_FIELDS["touristic_event"] = [
        'review',
        'published',
        'description_teaser',
        'description',
        'themes',
        'end_date',
        'duration',
        'meeting_point',
        'start_time',
        'end_time',
        'contact',
        'email',
        'website',
        'organizers',
        'speaker',
        'type',
        'accessibility',
        'capacity',
        'booking',
        'target_audience',
        'practical_info',
        'approved',
        'source',
        'portal',
        'eid',
        "bookable",
        'cancelled',
        'cancellation_reason'
        'place',
        'preparation_duration',
        'intervention_duration',
        'price'
    ],
HIDDEN_FORM_FIELDS["report"] = [
        "email",
```

(suite sur la page suivante)

(suite de la page précédente)

```
        "comment",
        "activity",
        "category",
        "problem_magnitude",
        "related_trek",
        "status",
        "locked",
        "uid",
        "origin",
        "assigned_user",
        "uses_timers"
    ],
HIDDEN_FORM_FIELDS["sensitivity_species"] = [
        "contact",
        "published",
        "description",
    ],
HIDDEN_FORM_FIELDS["sensitivity_regulatory"] = [
        "contact",
        "published",
        "description",
        "pictogram",
        "elevation",
        "url",
        "period01",
        "period02",
        "period03",
        "period04",
        "period05",
        "period06",
        "period07",
        "period08",
        "period09",
        "period10",
        "period11",
        "period12",
    ],
HIDDEN_FORM_FIELDS["blade"] = [
        "condition",
        "color",
    ],
HIDDEN_FORM_FIELDS["report"] = [
        "comment",
        "activity",
        "category",
        "problem_magnitude",
        "related_trek",
        "status",
        "locked",
        "uid",
        "origin"
    ],
```

(suite sur la page suivante)

(suite de la page précédente)

```
HIDDEN_FORM_FIELDS["circulationedge"] = [
    ]
```

15.6.3 Configure form fields required or needed for review or publication

Set `error_on_publication` to avoid publication without completeness fields and `error_on_review` if you want this fields to be required before sending to review.

COMPLETENESS_LEVEL

Configure completeness level

Example :

```
COMPLETENESS_LEVEL = 'warning'
```

COMPLETENESS_FIELDS

For each module, configure fields to be needed or required on review or publication

Example :

```
COMPLETENESS_FIELDS = {
'trek': ['practice', 'departure', 'duration', 'difficulty', 'description_teaser'],
'dive': ['practice', 'difficulty', 'description_teaser'],
}
```

15.7 Edition

15.7.1 WYSIWYG editor configuration

Text form fields are enhanced using TinyMCE.

Its configuration can be customized using advanced settings (see above paragraph).

TINYMCE_DEFAULT_CONFIG

For example, in order to control which buttons are to be shown, and which tags are to be kept when cleaning-up, add this bloc :

Example :

```
TINYMCE_DEFAULT_CONFIG = {
'theme_advanced_buttons1': 'bold,italic,forecolor,separator,code',
'valid_elements': "img,p,a,em/i,strong/b",
}
```

Note :

— This will apply to all text fields.

— For more information on configuration entries available, please refer to the official documentation of *TinyMCE version 3*.

15.7.2 Max characters count

MAPENTITY_CONFIG for characters

Add MAX_CHARACTERS setting to be able to define a maximum number of characters for text fields (to be used with django-mapentity >= 8.1).

Example :

```
MAPENTITY_CONFIG['MAX_CHARACTERS'] = 1500
```

Note :

— This will apply to all text fields.

— See this issue for details.

15.7.3 Copyright on pictures

THUMBNAIL_COPYRIGHT_FORMAT

If you want copyright added to your pictures, change this parameter like so :

Example :

```
THUMBNAIL_COPYRIGHT_FORMAT = "{title} {author}"
```

Note :

— This will apply to all text fields.

— For more information on configuration entries available, please refer to the official documentation of *TinyMCE version 3*.

You can also add {legend} : "{title}-:-{author}-:-{legend}"

THUMBNAIL_COPYRIGHT_SIZE

Change the size of thumbnail

Example :

```
THUMBNAIL_COPYRIGHT_SIZE = 15
```

15.7.4 Facebook configuration

When a content is shared to Facebook in Geotrek-rando V2, it needs static html files built by synchronization (thanks to option `--rando-url`).

In Facebook developper dashboard, create a Facebook app dedicated to Geotrek-rando and activate it.

FACEBOOK_APP_ID

In `custom.py` set Facebook App ID :
Example :

```
FACEBOOK_APP_ID = '<your Facebook AppID>'
```

You can also override these settings :

```
FACEBOOK_IMAGE = '/images/logo-geotrek.png'
FACEBOOK_IMAGE_WIDTH = 200
FACEBOOK_IMAGE_HEIGHT = 200
```

15.7.5 Override translations

Translations are managed by https://weblate.makina-corpus.net/ where you can contribute. But you can also override default translation files available in each module (for example those from trekking module available in `/opt/geotrek-admin/lib/python3.6/site-packages/geotrek/trekking/locale/fr/LC_MESSAGES/django.po`).

Don't edit these default files, use them to find which words you want to override.

Create the custom translations destination folder :

— Create a `django.po` file in `/opt/geotrek-admin/var/conf/extra_locale` directory.
— You can do one folder and one `django.po` file for each language (example `/opt/geotrek-admin/var/conf/extra_locale/fr/LC_MESSAGES/django.po` for French translation overriding)

Override the translations that you want in these files.

Example of content for the French translation overriding :

```
# MY FRENCH CUSTOM TRANSLATION
# Copyright (C) YEAR THE PACKAGE'S COPYRIGHT HOLDER
# This file is distributed under the same license as the PACKAGE package.
# FIRST AUTHOR <EMAIL@ADDRESS>, YEAR.
```

(suite sur la page suivante)

(suite de la page précédente)

```
#
msgid ""
msgstr ""
"Report-Msgid-Bugs-To: \n"
"POT-Creation-Date: 2018-11-15 15:32+0200\n"
"PO-Revision-Date: 2018-11-15 15:33+0100\n"
"Last-Translator: \n"
"Language-Team: LANGUAGE <LL@li.org>\n"
"MIME-Version: 1.0\n"
"Content-Type: text/plain; charset=UTF-8\n"
"Content-Transfer-Encoding: 8bit\n"
"Project-Id-Verésion: PACKAGE VERSION\n"
"Plural-Forms: nplurals=2; plural=(n > 1);\n"
"Project-Id-Version: \n"
"X-Generator: Poedit 1.5.4\n"

msgid "City"
msgstr "Région"

msgid "District"
msgstr "Pays"
```

Apply changes (French translation in this example) :

```
cd /opt/geotrek-admin/var/conf/extra_locale
sudo chown geotrek. fr/LC_MESSAGES/
sudo geotrek compilemessages
sudo service geotrek restart
```

15.7.6 Override public PDF templates

PDF are generated from HTML templates, using Django templating. Treks, touristic contents, touristic events, outdoor sites and courses can be exported in PDF files.

- Treks : `geotrek/trekking/templates/trekking/trek_public_pdf.html`
- Touristic contents : `geotrek/tourism/templates/tourism/touristiccontent_public_pdf.html`
- Touristic events : `geotrek/tourism/templates/tourism/touristicevent_public_pdf.html`
- Outdoor sites : `geotrek/outdoor/templates/outdoor/site_public_pdf.html`
- Outdoor courses : `geotrek/outdoor/templates/outdoor/course_public_pdf.html`

Overriden templates have to be located in `/opt/geotrek-admin/var/conf/extra_templates/<appname>`, with `<appname> = trekking` or `tourism`. To override trekking PDF for example, copy the file `geotrek/trekking/templates/trekking/trek_public_pdf.html` to `/opt/geotrek-admin/var/conf/extra_templates/trekking/trek_public_pdf.html`. Or add inside your file :

```
{% extends "trekking/trek_public_pdf.html" %}
```

These templates derive from base templates, which content is organized in blocks. To override for example the description block of trek PDF, copy and change the `{% block description }...{% endblock description %}` in your `/opt/geotrek-admin/var/conf/extra_templates/trekking/trek_public_pdf.html`.

It is also possible to use color defined for practice for pictogram by adding in your `/opt/geotrek-admin/var/conf/extra_templates/trekking/trek_public_pdf.html` file :

```
{% block picto_attr %}style="background-color: {{ object.practice.color }};"{% endblock␣
↪picto_attr %}
```

CSS can be overriden like html templates : copy them to `var/media/templates/trekking/` or `var/media/templates/tourism/` folder `/opt/geotrek-admin/var/conf/extra_templates/trekking/trek_public_pdf.css` for example.

You can also create a template for each portal.

Add a folder `portal_{id_portal}` (portal ids are located in the portal url path `/admin/common/targetportal/{id_portal}`) in `/opt/geotrek-admin/var/conf/extra_templates/<appname>`, as the first template, and add at the top of your file :

```
{% extends "trekking/trek_public_pdf.html" %}
```

The template for a specific portal will use the modification made on the overriden template in `/opt/geotrek-admin/var/conf/extra_templates/<appname>` (except if you change specific block)

Note : This modification is not mandatory, if you have multiple portal and you want to modify the template of only one portal, you create one folder for this specific portal

---

You might need to use your own images in the PDF templates.

Add your own images in `/opt/geotrek-admin/var/conf/extra_static/images/`.

You can then use these images in your PDF templates with `{% static 'images/file.jpg' %}`, after adding `{% load static %}` at the top of the file.

Example of a customised template (`/opt/geotrek-admin/var/conf/extra_templates/trekking/trek_public_pdf.html`) with a customised logo and URL :

```
{% extends "trekking/trek_public_pdf.html" %}
{% load static %}

{% block logo %}
   <img src="{% static 'images/logo-gte.jpg' %}" alt="Grand tour des Ecrins">
{% endblock %}
{% block url %}
   <div class="main">Grand tour des Ecrins</div>
   <div class="geo"><a href="https://www.grand-tour-ecrins.fr">grand-tour-ecrins.fr</a></
↪div>
{% endblock url %}
```

Note : The default template may change in the future versions. You will be in charge of porting the modification to your copy.

Test your modifications by exporting a trek or a content to PDF from Geotrek-admin application. To get your modifications available for Rando application, launch the `sync_rando` command.

15.7.7 PDF as booklet

USE_BOOKLET_PDF

Use booklet for PDF

Example :

```
USE_BOOKLET_PDF = True
```

Default :

```
False
```

Note :

— During the synchro, pois details will be removed, and the pages will be merged.

— It is possible to customize the pdf, with trek_public_booklet_pdf.html.

15.7.8 Custom font in public document template

In order to use custom fonts in trek PDF, it is necessary to install the font files on the server.

Microsoft fonts like *Arial* and *Verdana* can be installed via the package manager :

```
sudo apt-get install ttf-mscorefonts-installer
```

For specific fonts, copy the `.ttf` (or `.otf`) files into the folder `/usr/local/share/fonts/custom/` as root, and run the following command :

```
fc-cache
```

For more information, check out Ubuntu documentation.

15.7.9 Custom colors in public document template

MAPENTITY_CONFIG for custom colors in PDF

Trek export geometries are translucid red by default. In order to control the apparence of objects in public trek PDF exports, use the following setting :

Example :

```
MAPENTITY_CONFIG['MAP_STYLES']['print']['path'] = {'weight': 3}
```

See *Leaflet* reference documentation for detail about layers apparence.

15.7.10 Primary color in PDF templates

PRIMARY_COLOR

You can override PRIMARY_COLOR to change emphase text in PDF export.
Example :

```
PRIMARY_COLOR = "#7b8c12"
```

Note : Beware of contrast, white colour is used for text so we advise you to avoid light colour.

15.7.11 Custom logos

You might also need to deploy logo images in the following places :
- var/conf/extra_static/images/favicon.png
- var/conf/extra_static/images/logo-login.png
- var/conf/extra_static/images/logo-header.png

15.8 Settings for Geotrek-rando

15.8.1 Synchro Geotrek-rando

With Geotrek-rando V2, there is a synchronization mechanism to expose Geotrek-admin contents in json files generated automatically.

Avertissement : This is no more used in Geotrek-rando V3.

SYNC_RANDO_ROOT

Path on your server where the data for Geotrek-rando website will be generated
Example :

```
SYNC_RANDO_ROOT = os.path.join(VAR_DIR, 'data')
```

Note :
- If you want to modify it, do not forget to import os at the top of the file.
- Check import Python , if you need any information

SYNC_RANDO_OPTIONS

Options of the sync_rando command in Geotrek-admin interface.

Example :

```
SYNC_RANDO_OPTIONS = {}
```

15.8.2 Distances

TOURISM_INTERSECTION_MARGIN

Distance to which tourist contents, tourist events, treks, pois, services will be displayed

Example :

```
TOURISM_INTERSECTION_MARGIN = 500 # meters
```

Default :

```
500
```

Note : This distance can be changed by practice for treks in the admin.

DIVING_INTERSECTION_MARGIN

Distance to which dives will be displayed.

Example :

```
DIVING_INTERSECTION_MARGIN = 500 # meters
```

Default :

```
500
```

15.8.3 Limits

TREK_EXPORT_POI_LIST_LIMIT

Limit of the number of POIs on treks pdf.

Example :

```
TREK_EXPORT_POI_LIST_LIMIT = 14
```

Note : 14 is already a huge amount of POI, but it's possible to add more

TREK_EXPORT_INFORMATION_DESK_LIST_LIMIT

Limit of the number of information desks on treks pdf.

Example :

```
TREK_EXPORT_INFORMATION_DESK_LIST_LIMIT = 14
```

Note : You can put -1 if you want all the information desks

15.8.4 Categories

SPLIT_TREKS_CATEGORIES_BY_PRACTICE

On the Geotrek-rando V2 website, treks practices will be displayed separately

Example :

```
SPLIT_TREKS_CATEGORIES_BY_PRACTICE = False
```

Default :

```
True
```

Note : Field order for each practices in admin will be taken in account

---

SPLIT_TREKS_CATEGORIES_BY_ACCESSIBILITY

On the Geotrek-rando V2 website, accessibilites will be displayed separately

Example :

```
SPLIT_TREKS_CATEGORIES_BY_ACCESSIBILITY = False
```

Default :

```
True
```

SPLIT_TREKS_CATEGORIES_BY_ITINERANCY

On the Geotrek-rando V2 website, if a trek has a children it will be displayed separately

Example :

```
SPLIT_TREKS_CATEGORIES_BY_ITINERANCY = False
```

Default :

```
True
```

SPLIT_DIVES_CATEGORIES_BY_PRACTICE

On the Geotrek-rando V2 website, dives practices will be displayed separately

Example :

```
SPLIT_DIVES_CATEGORIES_BY_PRACTICE = True
```

Default :

```
False
```

HIDE_PUBLISHED_TREKS_IN_TOPOLOGIES

On the Geotrek-rando V2 website, treks near other are hidden

Example :

```
HIDE_PUBLISHED_TREKS_IN_TOPOLOGIES = False
```

Default :

```
True
```

TREK_WITH_POIS_PICTURES

It enables correlated pictures on Geotrek-rando V2 to be displayed in the slideshow

Example :

```
TREK_WITH_POIS_PICTURES = False
```

Default :

```
True
```

ONLY_EXTERNAL_PUBLIC_PDF

On Geotrek-rando V2 website, only PDF imported with filetype « Topoguide »will be used and not autogenerated.

Example :

```
ONLY_EXTERNAL_PUBLIC_PDF = False
```

Default :

```
True
```

Order of all the objects without practices on Geotrek-rando website :

```
TREK_CATEGORY_ORDER = 1
ITINERANCY_CATEGORY_ORDER = 2
DIVE_CATEGORY_ORDER = 10
TOURISTIC_EVENT_CATEGORY_ORDER = 99
```

Note :

— All the settings about order are the order inside Geotrek-rando website.
— Practices of diving, treks and categories of touristic contents are taken in account

15.9 Settings for Geotrek-mobile

SYNC_MOBILE_ROOT

Path on your server where the datas for mobile will be saved.

Example :

```
SYNC_MOBILE_ROOT = os.path.join(VAR_DIR, 'mobile')
```

Note :

— If you want to modify it, do not forget to import os at the top of the file.
— Check import Python , if you need any information

SYNC_MOBILE_OPTIONS

Options of the sync_mobile command.

Example :

```
SYNC_MOBILE_OPTIONS = {'skip_tiles': False}
```

Default :

```
True
```

MOBILE_NUMBER_PICTURES_SYNC

Number max of pictures that will be displayed and synchronized for each object (trek, POI, etc.) in the mobile app.

Example :

```
MOBILE_NUMBER_PICTURES_SYNC = 3
```

MOBILE_TILES_URL

URL's Tiles used for the mobile.

Example with OpenTopoMap :

```
MOBILE_TILES_URL = ['https://{s}.tile.opentopomap.org/{z}/{x}/{y}.png']
```

Example with IGN :

```
MOBILE_TILES_URL = ['https://data.geopf.fr/wmts?SERVICE=WMTS&
↪REQUEST=GetTile&VERSION=1.0.0&LAYER=GEOGRAPHICALGRIDSYSTEMS.PLANIGNV2&
↪STYLE=normal&FORMAT=image/png&TILEMATRIXSET=PM&TILEMATRIX={z}&TILEROW={y}&
↪TILECOL={x}']
```

MOBILE_LENGTH_INTERVALS

Intervals of the mobile for the length filter.

Example :

```
MOBILE_LENGTH_INTERVALS =  [
{"id": 1, "name": "< 10 km", "interval": [0, 9999]},
{"id": 2, "name": "10 - 30", "interval": [9999, 29999]},
{"id": 3, "name": "30 - 50", "interval": [30000, 50000]},
{"id": 4, "name": "> 50 km", "interval": [50000, 999999]}
]
```

Note :

— Interval key is in meters.
— You can add new intervals

MOBILE_ASCENT_INTERVALS

Intervals of the mobile for the ascent filter.

Example :

```
MOBILE_ASCENT_INTERVALS = [
{"id": 1, "name": "< 300 m", "interval": [0, 299]},
{"id": 2, "name": "300 - 600", "interval": [300, 599]},
{"id": 3, "name": "600 - 1000", "interval": [600, 999]},
{"id": 4, "name": "> 1000 m", "interval": [1000, 9999]}
]
```

Note : Do the same as above

MOBILE_DURATION_INTERVALS

Intervals of the mobile for the duration filter.

Example :

```
MOBILE_DURATION_INTERVALS = [
{"id": 1, "name": "< 1 heure", "interval": [0, 1]},
{"id": 2, "name": "1h - 2h30", "interval": [1, 2.5]},
{"id": 3, "name": "2h30 - 5h", "interval": [2.5, 5]},
{"id": 4, "name": "5h - 9h", "interval": [5, 9]},
{"id": 5, "name": "> 9h", "interval": [9, 9999999]}
]
```

Note : Check `MOBILE_LENGTH_INTERVALS` section to use it, here interval correspond to 1 unit of hour

ENABLED_MOBILE_FILTERS

List of all the filters enabled on mobile.

Example :

```
ENABLED_MOBILE_FILTERS = [
'practice',
'difficulty',
'durations',
'ascent',
'lengths',
'themes',
'route',
'districts',
'cities',
'accessibilities',
]
```

Note : Remove any of the filters if you don't want one of them. It's useless to add other one.

CHAPITRE 16

Maintenance

— *Application backup*
— *Application restore*
— *PostgreSQL optimization*
— *Access your database securely on your local machine (QGIS)*
— *Manage Cache*
— *Major evolutions from version 2.33*

16.1 Application backup

Database

```
sudo -u postgres pg_dump --no-acl --no-owner -Fc geotrekdb > `date +%Y%m%d%H%M`-database.
↪backup
```

Media files

```
tar -zcvf `date +%Y%m%d%H%M`-media.tar.gz /opt/geotrek-admin/var/media/
```

Configuration

```
tar -zcvf `date +%Y%m%d%H%M`-conf.tar.gz /opt/geotrek-admin/var/conf/
```

16.2 Application restore

If you restore Geotrek-admin on a new server, you will have to install PostgreSQL and PostGIS and create a database user first. Otherwise go directly to the database creation step.

Example for Ubuntu 18 :

```
sudo apt install postgresql-10 postgresql-10-postgis-2.5
sudo -u postgres psql -c "CREATE USER geotrek PASSWORD 'geotrek';"
```

Create an empty database (`geotrekdb` in this example) :

```
sudo -u postgres psql -c "CREATE DATABASE geotrekdb OWNER geotrek;"
sudo -u postgres psql -d geotrekdb -c "CREATE EXTENSION postgis;"
sudo -u postgres psql -d geotrekdb -c "CREATE EXTENSION postgis_raster;"
sudo -u postgres psql -d geotrekdb -c "CREATE EXTENSION pgcrypto;"
```

Restore backup :

```
pg_restore -U geotrek -h localhost --clean --no-acl --no-owner -d geotrekdb 20200510-
↪geotrekdb.backup
```

If errors persist, rename your database and recreate a fresh one, then restore.

Extract media and configuration files :

```
tar -zxvf 20200510-media.tar.gz
tar -zxvf 20200510-conf.tar.gz
```

Follow *Fresh installation* method. Choose to manage database by yourself.

16.3 PostgreSQL optimization

— Increase `shared_buffers` and `work_mem` according to your RAM
— Log long queries
— Use pg activity for monitoring

16.4 Access your database securely on your local machine (QGIS)

Instead of opening your database to the world (by opening the 5432 port for example), you can use SSH tunnels.

16.5 Manage Cache

— You can purge application cache with command or in admin interface

```
sudo geotrek clearcache --cache_name default --cache_name fat --cache_name api_v2h ori
```

16.6 Major evolutions from version 2.33

From version 2.33, Geotrek-admin is packaged in a debian package. This mean several things :

- a system user `geotrek` is created on install ;
- base code is located in `/opt/geotrek-admin` folder ;
- `geotrek` is the new command, replacing `bin/django`, and must be run in root (system user `geotrek` is used after) ;
- there is no more `settings.ini` but an `env` file with environment variables ;
- configuration files (custom.py et env), parsers and all customisation files (templates and translations) are now located in `/opt/geotrek-admin/var/conf` ;
- we advise you to configure data synchronization in `/opt/geotrek-admin/var`

CHAPITRE 17

Synchronization

— *Manual synchronization*
— *Automatic synchronization*
— *Synchronization options*
— *Synchronization filtered by source and portal*
— *Synchronization filtered by touristic content categories*
— *Synchronization with a distant Geotrek-rando server*
— *Geotrek-mobile app v3*

17.1 Manual synchronization

To create data for Geotrek-rando (public web portal) and Geotrek-mobile (mobile phone app), just run this command :

```
sudo geotrek sync_rando /opt/geotrek-admin/var/data
```

The parameter is the destination directory for synchronized data. If you choose another directory, make sure the parent of this directory is writable by geotrek user. Otherwise you will get a PermissionError message.

If Geotrek-admin is not accessible on localhost :80, you have to use the `--url` option. To make output less or more verbose, you can use the `--verbose` option.

Since version 2.4.0 of Geotrek-admin, you can also launch the command `sync_rando` from the web interface. You can add synchronization options with advanced configuration setting `SYNC_RANDO_OPTIONS = {}`.

For example, if you add this line in `/opt/geotrek-admin/var/conf/custom.py` you will skip generation of map tiles files during the synchronisation : `SYNC_RANDO_OPTIONS = {'skip_tiles': True}`

17.2 Automatic synchronization

You can set up automatic synchronization by creating a file `/etc/cron.d/geotrek_sync` that contains :

```
0 3 * * * root /usr/sbin/geotrek sync_rando /opt/geotrek-admin/var/data
```

This example will automatically synchronize data a 3 am every day.

Note : it is required to give the full path to the geotrek command since cron set the PATH only to *bin :/usr/bin*.

17.3 Synchronization options

```
Options:
  -v VERBOSITY, --verbosity=VERBOSITY
                        Verbosity level; 0=minimal output, 1=normal output,
                        2=verbose output
  -u URL, --url=URL     Base URL of Geotrek-admin (eg. http://geotrek.myorganization.com)
  -r URL, --rando-url=URL
                        Base URL of public Geotrek-rando website, used for static html␣
↪versions of objects pages
                        generated for Facebook in meta folder of data API
  -s SOURCE, --source=SOURCE
                        Filter by source(s)
  -P PORTAL, --portal=PORTAL
                        Filter by portal(s)
  -p, --skip-pdf        Skip generation of PDF files
  -t, --skip-tiles      Skip generation of map tiles files for Geotrek-mobile app v2
  -d, --skip-dem        Skip generation of Digital Elevation Model files for 3D view
  -e, --skip-profile-png
                        Skip generation of PNG elevation profile
  -w, --with-touristicevents
                        Include touristic events by trek in global.zip for Geotrek-
↪mobile v2
  -c CONTENT_CATEGORIES, --with-touristiccontent-categories=CONTENT_CATEGORIES
                        Include touristic contents by trek in global.zip for Geotrek-
↪mobile v2
                        (filtered by category ID ex: --with-touristiccontent-categories=
↪"1,2,3")
  -g, --with-signages   Include published signages
  -i, --with-infrastructures
                        Include published infrastructures
```

Geotrek-mobile v3 uses its own synchronization command (see below). If you are not using Geotrek-mobile v2 anymore, it is recommanded to use `-t` option to don't generate big offline tiles directories, not used elsewhere than in Geotrek-mobile v2. Same for `-w` and `-c` option, only used for Geotrek-mobile v2.

17.4 Synchronization filtered by source and portal

You can filter treks, touristic contents, touristic events and static pages by source(s). For example, if you created 3 sources records named `source A`, `source B` and `source C` and you want to only export data from `source A` and `source B` to your web public portal, you can synchronize with :

```
sudo geotrek sync_rando --source "source A,source B" dataAB
```

Multiple sources are separated with comas (without space before or after coma). Do not forget to add double quotes after and before the parameter if there are spaces in source names. You can run several commands to export several sources combinations into several directories and use them to publish several distinct web portals.

You can do exactly the same with `Target_Portal` field value. It will include objects associated to the selected portal + those without portal.

```
sudo geotrek sync_rando --portal "portal A" dataA
```

17.5 Synchronization filtered by touristic content categories

In Geotrek-mobile v2, you can choose to also include touristic content per trek. You must specify ID categories :

```
sudo geotrek sync_rando --with-touristiccontent-categories="1,3"
```

Multiple categories are separated with comas (without space before or after coma).

17.6 Synchronization with a distant Geotrek-rando server

If your server hosts both Geotrek-admin and Geotrek-rando, you just have to configure Geotrek-rando so it uses the directory chosen above. Be sure NGINX or Apache will have access rights to read these data.

If you have separated servers, you have to copy files, for example with `rsync` command :

```
rsync /path/of/generated/data other-server:/path/of/generated/data
```

17.7 Geotrek-mobile app v3

The Geotrek-mobile app v3 has its own API and synchronization command called `sync_mobile`.

It has similar parameters as `sync_rando` :

```
sudo geotrek sync_mobile [-h] [--languages LANGUAGES] [--portal PORTAL]
                         [--skip-tiles] [--url URL] [--indent INDENT]
                         [--version] [-v {0,1,2,3}] [--settings SETTINGS]
                         [--pythonpath PYTHONPATH] [--traceback]
                         [--no-color] [--force-color]
                         path
```

CHAPITRE 18

Import data

- *Import paths*
- *Import data from touristic data systems (SIT)*
  - *Configure APIDAE (ex-SITRA) import*
  - *Import Treks from APIDAE*
  - *Import from LEI*
  - *Configure Marque Esprit Parc import*
  - *Sensitive areas import*
  - *Multiple imports*
  - *Importing from multiple sources with deletion*
  - *Generic settings for your parser*
  - *Start import from command line*
  - *Start import from Geotrek-admin UI*
- *Import data from a remote Geotrek instance*
- *Import other datas from a file*
  - *Import DEM (altimetry)*
  - *Import POIs*
  - *Import Infrastructure*
  - *Import Dive*
  - *Import Signage*
  - *Import Cities*
  - *Import Districts*
  - *Import Restricted areas*
- *Exploitation commands*
  - *Delete attachment from disk*
  - *Remove duplicate paths*
  - *Merge segmented paths*
  - *Unset structure on categories*
  - *Reorder topologies*
  - *Automatic commands*

18.1 Import paths

Danger : With dynamic segmentation, importing paths is very risky if paths are already present in the same area in Geotrek, it is only safe for an area where no path is already created.

Indeed, if you import paths where there are existing paths, treks, POIs or trails linked topology might be impacted.

Before import paths layer, it is important to prepare them. Paths must be :
- valid geometry
- simple geometry (no intersection)
- all intersections must cut the paths
- not double or covering others

We use QGis to clean a path layer, with plugin Grass. Here are the operations :
- check the SRID (must be the same as in Geotrek)
- vectors → geometric tools → « collect geometries »
- vectors → geometric tools → « group »
- **clean geometries**
  - search « v_clean » in « Processing toolbox »
  - select following options in cleaning tool : break, snap, duplicate (ou rmdup), rmline, rmdangle, chdangle, bpol, prune
  - in threshold enter 2,2,2,2,2,2,2,2 (2 meters for each option)
- **delete duplicate geometries**
  - search « duplicate » in « Processing toolbox »
- **regroup lines**
  - search « v.build.polyline » in « Processing toolbox »)
  - select « first » in « Category number mode »

There are two ways to import path : importing your shapefile with command line, or via QGis following this blog post.

To import a shapefile containing your paths, use the command `loadpaths` :

```
sudo geotrek loadpaths {Troncons.shp} \
    --srid=2154 --comments-attribute IT_VTT IT_EQ IT_PEDEST \
    --encoding latin9 -i
```

18.2 Import data from touristic data systems (SIT)

18.2.1 Configure APIDAE (ex-SITRA) import

To import touristic content from APIDAE (ex-SITRA), edit `/opt/geotrek-admin/var/conf/parsers.py` file with the following content :

```
from geotrek.tourism.parsers import TouristicContentApidaeParser

class HebergementParser(TouristicContentApidaeParser):
    label = "Hébergements"
    api_key = 'xxxxxxxx'
    project_id = 9999
    selection_id = 99999
    category = "Hébergement"
```

(suite sur la page suivante)

(suite de la page précédente)

```
    type1 = ["Camping"]
    type2 = ["3 étoiles", "Tourisme et Handicap"]  # just remove this line if no type2
```

Then set up appropriate values :

- `label` at your convenience,
- `api_key`, `project_id` and `selection_id` according to your APIDAE (ex-SITRA) configuration
- `category`, `type1` and `type2` (optional) to select in which Geotrek category/type imported objects should go
- You can add `delete = True` in your class if you want to delete objects in Geotrek databases that has been deleted in your Apidae selection. It will only delete objects that match with your class settings (category, types, portal, provider…)
- You can also use the class `HebergementParser` if you only import accomodations
- See https://github.com/GeotrekCE/Geotrek-admin/blob/master/geotrek/tourism/parsers.py for details about Parsers

You can duplicate the class. Each class must have a different name. Don't forget the u character before strings if they contain non-ascii characters.

To apply changes, you may have to run `sudo service geotrek restart`.

18.2.2 Import Treks from APIDAE

A parser implementation is available to import Treks from APIDAE. Use it by defining a subclass of `` `geotrek.trekking.parsers.ApidaeTrekParser `` in your `var/conf/parsers.py` configuration file as shown above.

You'll have to configure how to access your APIDAE data : `api_key`, `project_id` and `selection_id` (those are setting attributes from the APIDAE base parser).

The `practices_mapped_with_activities_ids` and `practices_mapped_with_default_activities_ids` attributes define default mapping with the trekking module data fixture. You may override this to match your own types of Trek Practice.

18.2.3 Import from LEI

To import touristic content or touristic event from LEI , create (or update) `/opt/geotrek-admin/var/conf/parsers.py` file with the following content :

```
from geotrek.tourism.parsers import LEITouristicContentParser, LEITouristicEventParser

class XXXLEIContentParser(LEITouristicContentParser):
    label = "LEI TouristicContent"
    url = "https://url.asp"

class XXXLEIEventParser(LEITouristicEventParser):
    label = "LEI TouristicEvent"
    url = "https://url.asp"
```

18.2.4 Configure Marque Esprit Parc import

To import touristic content from Esprit Parc national database, create (or update) `/opt/geotrek-admin/var/conf/parsers.py` file with the following content :

```
from geotrek.tourism.parsers import EspritParcParser

class XXXEspritParcParser(EspritParcParser):
    label = "Marque Esprit Parc"
    url = "https://gestion.espritparcnational.com/ws/?f=getProduitsSelonParc&codeParc=XXX
↪"
```

Then set up appropriate values :

— `XXX` by unique national park code (ex : PNE)

You can duplicate the class. Each class must have a different name. Don't forget the u character before strings if they contain non-ascii characters.

In this case categories and types in Geotrek database have to be the same as in Esprit parc database. Otherwise missing categories and types will be created in Geotrek database.

Imported contents will be automatically published and approved.

If you use an url that filters a unique category, you can change its name. Example to get only Honey products and set the Geotrek category and type in which import them :

```
class MielEspritParcParser(EspritParcParser):
    label = "Miel Esprit Parc national"
    url = "https://gestion.espritparcnational.com/ws/?f=getProduitsSelonParc&
↪codeParc=XXX&typologie=API"
    constant_fields = {
        'category': "GeotrekCategoryName",
        'published': True,
        'approved': True,
        'deleted': False,
    }
    m2m_constant_fields = {
        'type1': "GeotrekTypeName",
    }
```

URL to get Esprit parc types : *https ://gestion.espritparcnational.com/ws/ ?f=getTypologieProduits*.

18.2.5 Sensitive areas import

When sensitive areas module is enabled, Geotrek provides 3 parsers to import data :

— **Import sensitive areas from http ://biodiv-sports.fr** (`geotrek.sensitivity.parsers.BiodivParser`). By default this parser imports all sensitive areas in configured spatial extent.

— **Import species sensitive areas from a zipped shapefile**. Imported field names are : `espece` (required), `contact` and `descriptio`. Species with corresponding names have to be created manually before import.

— **Import regulatory sensitive areas from a zipped shapefile**. Imported field names are : `nom` (required), `contact`, `descriptio`, `periode` (month numbers separated with comas), `pratiques` (separated with comas), and `url`. Practices with corresponding names have to be created manually before import.

You can start imports from « Import » menu or from command line. You can override them in your `var/conf/parsers.py` file.

18.2.6 Multiple imports

When you need to import data for the same object found in 2 different parsers, you can to force the aggregation of both values in many to many relationship case. It can be interesting with portals for example.

Param for the aggregation : `m2m_aggregate_fields`

Here is an example with 2 parsers :

```
class Portal_1Parser(XXXParser):
    portal = "portal_1"

class AggregateParser(XXXParser):
    portal = "portal_2"
    m2m_aggregate_fields = ["portal"]
```

Then, when you import the first parser `Portal_1Parser`, you get multiple objects with `portal_1` as portal. If any object of the `Portal_1Parser` is also in `AggregateParser`, fields in `m2m_aggregate_fields` will have their values not be replaced but aggregated. Then your object in both portals will have as portal : `portal_1, portal_2`

— Here in this example whenever you import the first parser `Portal_1Parser`, portals are replaced because `m2m_aggregate_fields` is not filled. Then, be careful to import parsers in the right order or add the param `m2m_aggregate_fields` on all parsers.

If you need to cancel the aggregation of portals, remove param `m2m_aggregate_fields`.

18.2.7 Importing from multiple sources with deletion

When importing data for the same model using two (or more) different sources, the `provider` field should be used to differenciate between sources, allowing to enable object deletion with `delete = True` without causing the last parser to delete objects created by preceeding parsers.

In the following example, `Provider_1Parser` and `Provider_2Parser` will each import their objects, set the `provider` field on these objects, and only delete objects that disappeared from their respective source since last parsing.

```
class Provider_1Parser(XXXXParser):
    delete = True
    provider = "provider_1"

class Provider_2Parser(XXXParser):
    delete = True
    provider = "provider_2"
```

Danger : It is recommended to use `provider` from the first import - Do not add a `provider` field to preexisting parsers that already imported objects, or you will have to manually set the same value for `provider` on all objects already created by this parser.

Danger : If a parser does not have a `provider` value, it will not take providers into account, meaning that it could delete objects from preceeding parsers even if these other parsers do have a `provider` themselves.

The following example would cause `NoProviderParser` to delete objects from `Provider_2Parser` and `Provider_1Parser`.

```
class Provider_1Parser(XXXXParser):
    delete = True
    provider = "provider_1"

class Provider_2Parser(XXXParser):
    delete = True
    provider = "provider_2"

class NoProviderParser(XXXParser):
    delete = True
    provider = None       (default)
```

18.2.8 Generic settings for your parser

This settings may be overriden when you define a new parser :

- `label` parser display name (default : `None`)
- `model` import content with this model (default : `None`)
- `filename` file imported if no url (default : `None`)
- `url` flow url imported from if no filename (default : `None`)
- `simplify_tolerance` (default : `0`) # meters
- `update_only` don't create new contents (default : `False`)
- `delete` (default : `False`)
- `duplicate_eid_allowed` if True, allows differents contents with same eid (default : `False`)
- `fill_empty_translated_fields` if True, fills empty translated fields with same value (default : `False`)
- `warn_on_missing_fields` (default : `False`)
- `warn_on_missing_objects` (default : `False`)
- `separator` (default : `'+'`)
- `eid` field name for eid (default : `None`)
- `provider` (default : `None`)
- `fields` (default : `None`)
- `m2m_fields` (default : `{}`)
- `constant_fields` (default : `{}`)
- `m2m_constant_fields` (default : `{}`)
- `m2m_aggregate_fields` (default : `[]`)
- `non_fields` (default : `{}`)
- `natural_keys` (default : `{}`)
- `field_options` (default : `{}`)
- `default_language` use another default language for this parser (default : `None`)

18.2.9 Start import from command line

Just run :

```
sudo geotrek import HebergementParser
```

Change `HebergementParser` to match one of the class names in `var/conf/parsers.py` file. You can add `-v2` parameter to make the command more verbose (show progress). Thank to `cron` utility you can configure automatic imports.

18.2.10 Start import from Geotrek-admin UI

Open the top right menu and clic on `imports`.

18.3 Import data from a remote Geotrek instance

Importing from a Geotrek instance works the same way as from SIT. A usecase for this is to aggregate data from several Geotrek-admin instance.

Danger : Importing data from a remote Geotrek instance does not work with dynamic segmentation, your instance where you import data must have dynamic segmentation disabled.

For example, to import treks from another instance, edit `/opt/geotrek-admin/var/conf/parsers.py` file with the following content :

```
class DemoGeotrekTrekParser(BaseGeotrekTrekParser):
    url = "https://remote-geotrek-admin.net"  # replace url with remote instance url
    delete = False
    field_options = {
        'difficulty': {'create': True, },
        'route': {'create': True, },
        'themes': {'create': True},
        'practice': {'create': True},
        'accessibilities': {'create': True},
        'networks': {'create': True},
        'geom': {'required': True},
        'labels': {'create': True},
    }
```

Then run in command line

```
sudo geotrek import DemoGeotrekTrekParser
```

Treks are now imported into your own instance.

18.4 Import other datas from a file

You can add parsers in your custom *parsers.py* file (`/opt/geotrek-admin/var/conf/parsers.py`) which will allow you to import data from files directly in your admin (superusers only). For example, some parsers are not available by default but you can use them adding some lines in your parsers file :

```
from geotrek.trekking.parsers import TrekParser # only without dynamic segmentation␣
↪(`TREKKING_TOPOLOGY_ENABLED` = False)
from geotrek.trekking.parsers import POIParser
```

You can also use some of Geotrek commands to import data from a vector file handled by GDAL (https://gdal.org/drivers/vector/index.htm) (e.g. : ESRI Shapefile, GeoJSON, GeoPackage etc.)

Possible data are e.g. : POI, infrastructures, signages, cities, districts, restricted areas, dives, paths.

You must use these commands to import spatial data because of the dynamic segmentation, which will not be computed if you enter the data manually.

Here are the Geotrek commands available to import data from file :

- `loaddem`
- `loadpoi`
- `loaddive`
- `loadinfrastructure`
- `loadsignage`
- `loadcities`
- `loaddistricts`
- `loadrestrictedareas`

Usually, these commands come with ability to match file attributes to model fields.

To get help about a command :

```
sudo geotrek help <subcommand>
```

18.4.1 Import DEM (altimetry)

`sudo geotrek help loaddem`

```
usage: manage.py loaddem [-h] [--replace] [--update-altimetry] [--version] [-v {0,1,2,3}
↪] [--settings SETTINGS] [--pythonpath PYTHONPATH] [--traceback] [--no-color] [--force-
↪color]
                         [--skip-checks]
                         dem_path

Load DEM data (projecting and clipping it if necessary). You may need to create a GDAL␣
↪Virtual Raster if your DEM is composed of several files.

positional arguments:
  dem_path

optional arguments:
  -h, --help            show this help message and exit
  --replace             Replace existing DEM if any.
  --update-altimetry    Update altimetry of all 3D geometries, /!\ This option takes lot␣
↪of time to perform
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will be␣
↪used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

18.4.2 Import POIs

sudo geotrek help loadpoi

```
usage: manage.py loadpoi [-h] [--encoding ENCODING] [--name-field NAME_FIELD] [--type-
↪field TYPE_FIELD] [--description-field DESCRIPTION_FIELD]
                         [--name-default NAME_DEFAULT] [--type-default TYPE_DEFAULT] [--
↪version] [-v {0,1,2,3}] [--settings SETTINGS] [--pythonpath PYTHONPATH]
                         [--traceback] [--no-color] [--force-color] [--skip-checks]
                         point_layer

Load a layer with point geometries in a model

positional arguments:
  point_layer

optional arguments:
  -h, --help            show this help message and exit
  --encoding ENCODING, -e ENCODING
                        File encoding, default utf-8
  --name-field NAME_FIELD, -n NAME_FIELD
                        Name of the field that contains the name attribute. Required or␣
↪use --name-default instead.
  --type-field TYPE_FIELD, -t TYPE_FIELD
                        Name of the field that contains the POI Type attribute. Required␣
↪or use --type-default instead.
  --description-field DESCRIPTION_FIELD, -d DESCRIPTION_FIELD
                        Name of the field that contains the description of the POI␣
↪(optional)
  --name-default NAME_DEFAULT
                        Default value for POI name. Use only if --name-field is not set
  --type-default TYPE_DEFAULT
                        Default value for POI Type. Use only if --type-field is not set
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will
                        be used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

18.4.3 Import Infrastructure

sudo geotrek help loadinfrastructure

```
usage: manage.py loadinfrastructure [-h] [--use-structure] [--encoding ENCODING]
    [--name-field NAME_FIELD] [--name-default NAME_DEFAULT]
    [--type-field TYPE_FIELD] [--type-default TYPE_DEFAULT]
    [--category-field CATEGORY_FIELD] [--category-default CATEGORY_DEFAULT]
    [--condition-field CONDITION_FIELD] [--condition-default CONDITION_DEFAULT]
    [--structure-field STRUCTURE_FIELD] [--structure-default STRUCTURE_DEFAULT]
    [--description-field DESCRIPTION_FIELD] [--description-default DESCRIPTION_DEFAULT]
    [--year-field YEAR_FIELD] [--year-default YEAR_DEFAULT]
    [--eid-field EID_FIELD]
    [--version] [-v {0,1,2,3}] [--settings SETTINGS] [--pythonpath PYTHONPATH] [--
↪traceback]
    [--no-color] [--force-color] [--skip-checks]
    point_layer

Load a layer with point geometries and import features as infrastructures objects
(expected formats: shapefile or geojson)

positional arguments:
  point_layer

optional arguments:
  -h, --help            show this help message and exit
  --use-structure
                        If set the given (or default) structure is used to select or␣
↪create
                        conditions and types of infrastructures.
  --encoding ENCODING, -e ENCODING
                        File encoding, default utf-8
  --name-field NAME_FIELD, -n NAME_FIELD
                        The field to be imported as the `name` of the infrastructure
  --name-default NAME_DEFAULT
                        Default name for all infrastructures, fallback for entries␣
↪without a name
  --type-field TYPE_FIELD, -t TYPE_FIELD
                        The field to select or create the type value of the␣
↪infrastructure
                        (field `InfrastructureType.label`)
  --type-default TYPE_DEFAULT
                        Default type for all infrastructures, fallback for entries␣
↪without a type.
  --category-field CATEGORY_FIELD, -i CATEGORY_FIELD
                        The field to select or create the type value of the␣
↪infrastructure
                        (field `InfrastructureType.type`)
  --category-default CATEGORY_DEFAULT
                        Default category for all infrastructures, "B" by default.␣
↪Fallback for entries
                        without a category
  --condition-field CONDITION_FIELD, -c CONDITION_FIELD
```

(suite sur la page suivante)

(suite de la page précédente)

```
                        The field to select or create the condition value of the␣
↪infrastructure
                        (field `InfrastructureCondition.label`)
  --condition-default CONDITION_DEFAULT
                        Default condition for all infrastructures, fallback for entries␣
↪without a category
  --structure-field STRUCTURE_FIELD, -s STRUCTURE_FIELD
                        The field to be imported as the structure of the infrastructure
  --structure-default STRUCTURE_DEFAULT
                        Default Structure for all infrastructures
  --description-field DESCRIPTION_FIELD, -d DESCRIPTION_FIELD
                        The field to be imported as the description of the infrastructure
  --description-default DESCRIPTION_DEFAULT
                        Default description for all infrastructures, fallback for entries
                        without a description
  --year-field YEAR_FIELD, -y YEAR_FIELD
                        The field to be imported as the `implantation_year` of the␣
↪infrastructure
  --year-default YEAR_DEFAULT
                        Default year for all infrastructures, fallback for entries␣
↪without a year
  --eid-field EID_FIELD
                        The field to be imported as the `eid` of the infrastructure␣
↪(external ID)
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output,
                            1=normal output,
                            2=verbose output,
                            3=very verbose output
  --settings SETTINGS
                        The Python path to a settings module, e.g. "myproject.settings.
↪main".
                        If this isn't provided, the DJANGO_SETTINGS_MODULE environment␣
↪variable
                        will be used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

Load a layer with point geometries and import entities as infrastructures objects.

- expected formats for the *point_layer* file are shapefile or geojson (other geodjango supported-formats may work but untested),
- the command updates existing Infrastructure objects based on the *eid* field (external ID),
- if the Infrastructure object does not exist (or if *eid* is not specified) it is created.

Usage example

```
sudo geotrek loadinfrastructure \
```

(suite sur la page suivante)

(suite de la page précédente)

```
    --name-field "shpname" \
    --type-field "shptype" \
    --description-field "shpdesc" \
    --year-field "shpyear" \
    --eid-field "shpid" \
    --condition-default "Badly damaged" \
    --year-default "2023" \
    --category-default "A" \
    ./infrastructures_to_be_imported.shp
```

— The command expects entries from *point_layer* file to have the the following fields : *shpname*, *shptype*, *shpdesc*, *shpyear* and *shpid*.
— A default value is provided for the condition. It will be set for all imported infrastructures.
— A default value is provided for the year in addition to the field mapping. In case the shapefile entry does not have a year attribute the command will take the default value instead.
— The command will select or create InfrastructureType values based on the *type* argument, taking the default value « A » for the category.

Required fields

The following fields are mandatory to create an Infrastructure object : *name*, *type* and *category*. For each of those fields either an import field and/or a default value MUST be provided. If the command is unable to determine values for those fields for a given layer, the layer is skipped with an error message.

Default values

— When a default value is provided without a fieldname to import the default value is set for all Infrastructure objects.
— When a default value is provided in addition to a fieldname to import it is used as a fallback for entries without the specified import field.

Selection and addition of parameterized values

Infrastructure objects have several values from Geotrek's parameterized values sets :
— *type* from InfrastructureType values (and *category* which is implied by the *type* value),
— *condition* from InfrastructureCondition values.

New parameterized values are created and added to Geotrek Admin if necessary. The command checks if the imported *type* value already exists by looking for an InfrastructureType with the right *type* + *category*.

```
sudo geotrek loadinfrastructure  --type-field "type"  --category-field "cat" [...]
```

Selected or added InfrastructureType value :
— label <- value of *type* import field
— type <- value of *cat* import field
— optionnally if *–use-structure* : structure <- the structure value (import field or default)

For InfrastructureCondition the check uses the *condition* argument.

```
sudo geotrek loadinfrastructure  --condition-field "cond" [...]
```

Selected or added InfrastructureCondition value :
— label <- value of *cond* field
— optionnally if *–use-structure* : structure <- the structure value (import field or default)

18.4.4 Import Dive

sudo geotrek help loaddive

```
usage: manage.py loaddive [-h] [--encoding ENCODING] [--name-field NAME_FIELD] [--depth-
↪field DEPTH_FIELD] [--practice-default PRACTICE_DEFAULT]
                          [--structure-default STRUCTURE_DEFAULT] [--eid-field EID_
↪FIELD] [--version] [-v {0,1,2,3}] [--settings SETTINGS] [--pythonpath PYTHONPATH] [--
↪traceback]
                          [--no-color] [--force-color] [--skip-checks]
                          point_layer

Load a layer with point geometries in the Dive model

positional arguments:
  point_layer

optional arguments:
  -h, --help            show this help message and exit
  --encoding ENCODING, -e ENCODING
                        File encoding, default utf-8
  --name-field NAME_FIELD, -n NAME_FIELD
  --depth-field DEPTH_FIELD, -d DEPTH_FIELD
  --practice-default PRACTICE_DEFAULT
  --structure-default STRUCTURE_DEFAULT
  --eid-field EID_FIELD
                        External ID field
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will be␣
↪used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

18.4.5 Import Signage

sudo geotrek help loadsignage

```
usage: manage.py loadsignage [-h] [--use-structure] [--encoding ENCODING] [--name-field␣
↪NAME_FIELD] [--type-field TYPE_FIELD] [--condition-field CONDITION_FIELD]
                             [--structure-field STRUCTURE_FIELD] [--description-field␣
↪DESCRIPTION_FIELD] [--year-field YEAR_FIELD] [--code-field CODE_FIELD]
                             [--type-default TYPE_DEFAULT] [--name-default NAME_DEFAULT]␣
↪[--condition-default CONDITION_DEFAULT] [--structure-default STRUCTURE_DEFAULT]
```

(suite sur la page suivante)

(suite de la page précédente)

```
                            [--description-default DESCRIPTION_DEFAULT] [--eid-field␣
↪EID_FIELD] [--year-default YEAR_DEFAULT] [--code-default CODE_DEFAULT] [--version]
                            [-v {0,1,2,3}] [--settings SETTINGS] [--pythonpath␣
↪PYTHONPATH] [--traceback] [--no-color] [--force-color] [--skip-checks]
                            point_layer

Load a layer with point geometries in te structure model

positional arguments:
  point_layer

optional arguments:
  -h, --help            show this help message and exit
  --use-structure       Allow to use structure for condition and type of infrastructures
  --encoding ENCODING, -e ENCODING
                        File encoding, default utf-8
  --name-field NAME_FIELD, -n NAME_FIELD
                        Name of the field that will be mapped to the Name field in␣
↪Geotrek
  --type-field TYPE_FIELD, -t TYPE_FIELD
                        Name of the field that will be mapped to the Type field in␣
↪Geotrek
  --condition-field CONDITION_FIELD, -c CONDITION_FIELD
                        Name of the field that will be mapped to the Condition field in␣
↪Geotrek
  --structure-field STRUCTURE_FIELD, -s STRUCTURE_FIELD
                        Name of the field that will be mapped to the Structure field in␣
↪Geotrek
  --description-field DESCRIPTION_FIELD, -d DESCRIPTION_FIELD
                        Name of the field that will be mapped to the Description field␣
↪in Geotrek
  --year-field YEAR_FIELD, -y YEAR_FIELD
                        Name of the field that will be mapped to the Year field in␣
↪Geotrek
  --code-field CODE_FIELD
                        Name of the field that will be mapped to the Code field in␣
↪Geotrek
  --type-default TYPE_DEFAULT
                        Default value for Type field
  --name-default NAME_DEFAULT
                        Default value for Name field
  --condition-default CONDITION_DEFAULT
                        Default value for Condition field
  --structure-default STRUCTURE_DEFAULT
                        Default value for Structure field
  --description-default DESCRIPTION_DEFAULT
                        Default value for Description field
  --eid-field EID_FIELD
                        External ID field
  --year-default YEAR_DEFAULT
                        Default value for Year field
  --code-default CODE_DEFAULT
```

(suite sur la page suivante)

(suite de la page précédente)

```
                        Default value for Code field
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will be␣
↪used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

18.4.6 Import Cities

sudo geotrek help loadcities

```
usage: manage.py loadcities [-h] [--code-attribute CODE] [--name-attribute NAME] [--
↪encoding ENCODING] [--srid SRID] [--intersect] [--version] [-v {0,1,2,3}] [--settings␣
↪SETTINGS]
                        [--pythonpath PYTHONPATH] [--traceback] [--no-color] [--force-
↪color] [--skip-checks]
                        file_path

Load Cities from a file within the spatial extent

positional arguments:
  file_path             File's path of the cities

optional arguments:
  -h, --help            show this help message and exit
  --code-attribute CODE, -c CODE
                        Name of the code's attribute inside the file
  --name-attribute NAME, -n NAME
                        Name of the name's attribute inside the file
  --encoding ENCODING, -e ENCODING
                        File encoding, default utf-8
  --srid SRID, -s SRID  File's SRID
  --intersect, -i       Check features intersect spatial extent and not only within
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will be␣
↪used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
```

(suite sur la page suivante)

(suite de la page précédente)

```
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

18.4.7 Import Districts

sudo geotrek help loaddistricts

```
usage: manage.py loaddistricts [-h] [--name-attribute NAME] [--encoding ENCODING] [--
↪srid SRID] [--intersect] [--version] [-v {0,1,2,3}] [--settings SETTINGS]
                               [--pythonpath PYTHONPATH] [--traceback] [--no-color] [--
↪force-color] [--skip-checks]
                               file_path

Load Districts from a file within the spatial extent

positional arguments:
  file_path             File's path of the districts

optional arguments:
  -h, --help            show this help message and exit
  --name-attribute NAME, -n NAME
                        Name of the name's attribute inside the file
  --encoding ENCODING, -e ENCODING
                        File encoding, default utf-8
  --srid SRID, -s SRID  File's SRID
  --intersect, -i       Check features intersect spatial extent and not only within
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will be␣
↪used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

18.4.8 Import Restricted areas

`sudo geotrek help loadrestrictedareas`

```
usage: manage.py loadrestrictedareas [-h] [--name-attribute NAME] [--encoding ENCODING]␣
↪[--srid SRID] [--intersect] [--version] [-v {0,1,2,3}] [--settings SETTINGS]
                                     [--pythonpath PYTHONPATH] [--traceback] [--no-
↪color] [--force-color] [--skip-checks]
                                     file_path area_type

Load Restricted Area from a file within the spatial extent

positional arguments:
  file_path             File's path of the restricted area
  area_type             Type of restricted areas in the file

optional arguments:
  -h, --help            show this help message and exit
  --name-attribute NAME, -n NAME
                        Name of the name's attribute inside the file
  --encoding ENCODING, -e ENCODING
                        File encoding, default utf-8
  --srid SRID, -s SRID  File's SRID
  --intersect, -i       Check features intersect spatial extent and not only within
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will be␣
↪used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

18.5 Exploitation commands

18.5.1 Delete attachment from disk

When an attachment (eg. pictures) is removed, its file is not automatically removed from disk. You have to run `sudo geotrek clean_attachments` manually or in a cron to remove old files. After that, you should run `sudo geotrek thumbnail_cleanup` to remove old thumbnails.

18.5.2 Remove duplicate paths

Duplicate paths can appear while adding paths with commands or directly in the application. Duplicate paths can cause some problems of routing for topologies, it can generate corrupted topologies (that become MultiLineStrings instead of LineStrings).

You have to run `sudo geotrek remove_duplicate_paths`

During the process of the command, every topology on a duplicate path will be set on the original path, and the duplicate path will be deleted.

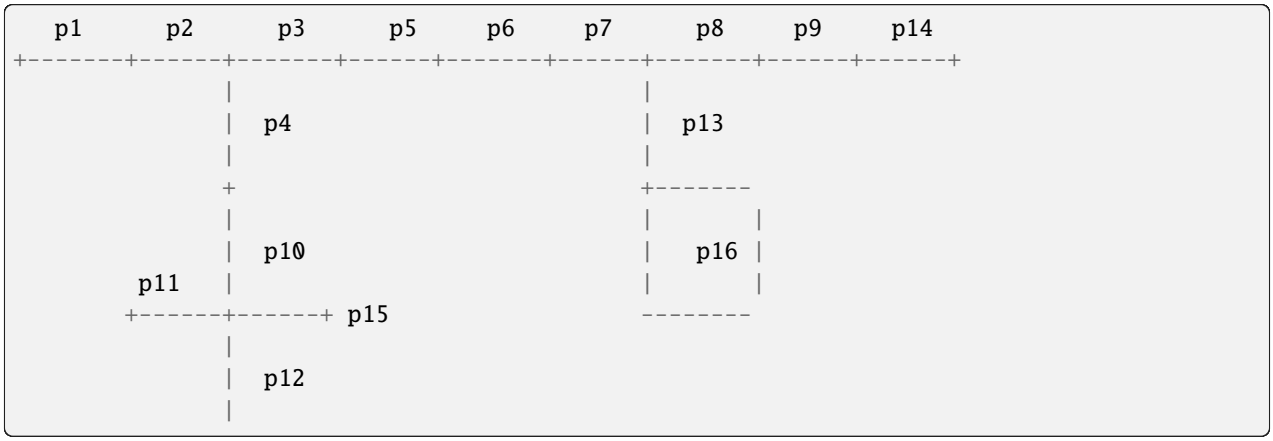
18.5.3 Merge segmented paths

A path network is most optimized when there is only one path between intersections. If the path database includes many fragmented paths, they could be merged to improve performances.

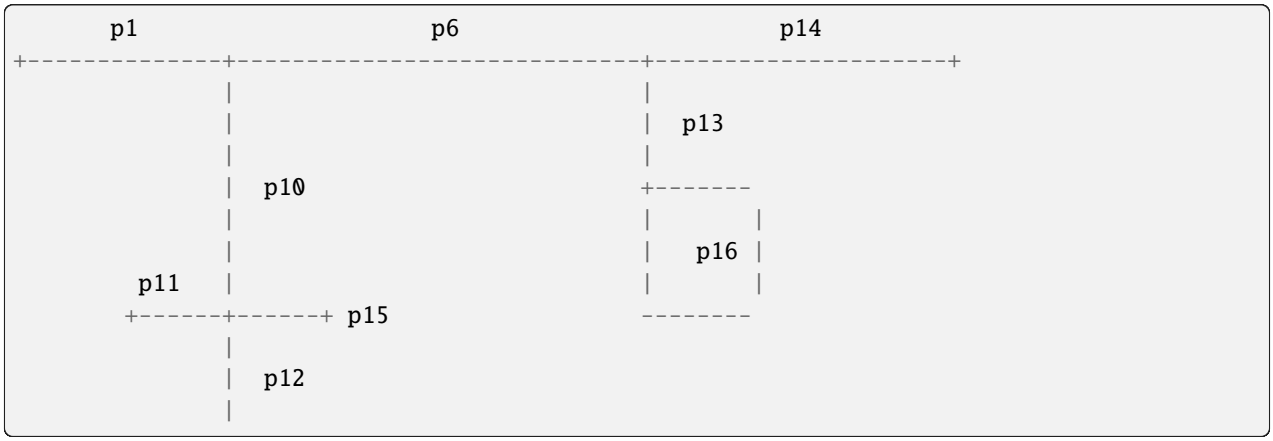
You can run `sudo geotrek merge_segmented_paths`.

Danger : This command can take several hours to run. During the process, every topology on a path will be set on the path it is merged with, but it would still be more efficient (and safer) to run it before creating topologies.

Before :



After :



18.5.4 Unset structure on categories

Use this command if you wish to undo linking categories to structures for some models.

You have to run `sudo geotrek unset_structure`

```
usage: manage.py unset_structure [-h] [--all] [--list] [--version] [-v {0,1,2,3}] [--
↪settings SETTINGS] [--pythonpath PYTHONPATH] [--traceback] [--no-color] [--force-color]
                                 [--skip-checks]
                                 [model [model ...]]

Unset structure in lists of choices and group choices with the same name.

positional arguments:
  model                 List of choices to manage

optional arguments:
  -h, --help            show this help message and exit
  --all                 Manage all models
  --list                Show available models to manage
  --version             show program's version number and exit
  -v {0,1,2,3}, --verbosity {0,1,2,3}
                        Verbosity level; 0=minimal output, 1=normal output, 2=verbose␣
↪output, 3=very verbose output
  --settings SETTINGS   The Python path to a settings module, e.g. "myproject.settings.
↪main". If this isn't provided, the DJANGO_SETTINGS_MODULE environment variable will be␣
↪used.
  --pythonpath PYTHONPATH
                        A directory to add to the Python path, e.g. "/home/
↪djangoprojects/myproject".
  --traceback           Raise on CommandError exceptions
  --no-color            Don't colorize the command output.
  --force-color         Force colorization of the command output.
  --skip-checks         Skip system checks.
```

Danger : You can't chose for each choice which set of category you want to unset structures, it will happen for all categories

Firstly, if a categroy is linked to a structure, it creates the same category but with no structure associated. Secondly, every element with this old category gets assigned to this new category. Finally all old categories are removed.

18.5.5 Reorder topologies

All topologies have information about which path they go through on and in which order. Actually, when a path is split in 2 by another path, a new path is added to the database. We need to add information for all topologies that need to go through this new path. This is badly managed at the moment, especially for the order of passage of the paths. `sudo geotrek reorder_topologies`

It removes a lot of useless information which can accelerate the process of editing topologies afterward.

During the process of this command, it tries to find a good order of passage on the paths which creates only one Linestring from start to end. It stays as close as possible to the corrupted order. This command uses the same algorithm to generate one Linestring when the order is not well managed during topologies" display.

Danger : It can happens that this algorithm can't find any solution and will genereate a MultiLineString. This will be displayed at the end of the reorder

18.5.6 Automatic commands

You can set up automatic commands by creating a *cron* file under `/etc/cron.d/geotrek_command` that contains :

```
0 3 * * * root /usr/sbin/geotrek <command> <options>
```

example :

```
0 4 * * * root /usr/sbin/geotrek reorder_topologies
```

This example will automatically reorder topologies at 4 am every day.

CHAPITRE 19

Contributing

— *Conventions*
— *Definition of done*
— *Pull requests*
— *Release*
— *Other ways to contribute*

19.1 Conventions

— Before contributing, open an issue and discuss about it with community (is it a bug or a feature ? What is the best way to achieve my goal ?)
— Use flake8
— KISS & DRY as much as possible
— Elegant and generic is good, simple is better
— Separate bug fixes and new features in several pull requests.
— Open a new Pull Request in « Draft » status until tests passed. Use at least “bug”, “improvement” or “feature” label.
— Commits messages are explicit and mention issue number (`(ref #12)` or `(fixes #23)`), they should contains corresponding tag (see below)
— Features are developed in a branch and merged from Github pull-requests.

19.2 Definition of done

— `docs/changelog.rst` is up-to-date
— An explicit unit-test covers the bugfix or the new feature.
— A frontend test (:path :jstests/nav-*.js) covers the navigation bug fix or feature
— A JS *Mocha* test (:path :jstests/tests.*.js) covers the JavaScript bug fix or feature
— Unit-tests total coverage is above or at least equal with previous commits. Patch coverage is 100% on new lines.
— Settings have default value in `settings/base.py`
— Installation instructions and documentation are up-to-date

Check TODO in the source tree :

```
find geotrek | xargs egrep -n -i '(TODO|XXX|temporary|FIXME)'
```

19.3 Pull requests

Before creating a pull request, ensure you follow thoses rules :
— Follow the guidelines of this page
— Self-review your code
— Add comments in your code, particularly in hard-to-understand areas
— Make corresponding changes to the documentation
— There is tests that prove my fix is effective or that my feature works.
— All new lines of code are tested
— There is an entry in the changelog file (with the corresponding issue referenced)
It is important to add a label to the pull request corresponding to the perimeter. Release notes are generated according to labels of pull requests. This is the list of available labels for pull requests :

Tag	Branch prefix	Emoji code	Unicode
Improvements	impr_	dizzy	
Bug fixes	bug_	bug	
Features	feat_	sparkles	
Documentation	doc_	memo	
Maintenance	maint_	building_construction	
Refactor	ref_	recycle	
Dependencies	dep_	arrow_up	
CI/CD	cicd_	construction_worker	
Performances	perf_	zap	
UI/UX	uiux_	lipstick	
Security	sec_	lock	
Translations	trans_	globe_with_meridians	
Hotfix	hot_	ambulance	
Breaking change	break_	boom	

Commits in pull requests are following a naming convention in order to easily establish their perimeter. Commit is formated as `emoji [tag] description of commit`.

You can use one of the above prefix for your commits but also one of the following (tags used only for commits, not for pull requests) :

Tag	Emoji code	Unicode
Codestyle	art	
Clean	fire	
Tests	white_check_mark	
Release	bookmark	
Merge	twisted_rightwards_arrows	

19.4 Release

On master branch :

— Update files *VERSION*, *docs/conf.py* and *docs/changelog.rst* to remove `+dev` suffix and increment version (please use semver rules)
— Run `dch -r -D RELEASED`, update version in the same way and save
— Commit with message "Release x.y.z" and push to `master`
— Create new release on Github, with tag X.Y.Z, click on « Generate release notes »
— Wait for release to be published through CI
— Update files *VERSION*, *docs/conf.py* and *docs/changelog.rst* to add `+dev` suffix
— Run `dch -v <version>+dev --no-force-save-on-release` and save
— Commit with message "Back to development" and push to `` `master ``
— When creating a new release "x.y.z" on github, Github actions will generate the .deb package file, and publish it on https://packages.geotrek.fr (see `.github/workflows/test.yml` file for details)

19.5 Other ways to contribute

— Help the users and answers questions on the mailing-list ;
— Open a ticket when you encounter a bug ;
— Open a ticket when you have a suggestion or feature idea ;
— Translate the documentation ;
— Translate the menus, buttons and labels (we use Weblate) ;
— Maintain the installation script for different Linux distributions (*requires some basic Linux skills*) ;
— Fix bugs or improve layout and apparence (*requires Webmaster skills*) ;
— Fix bugs or improve core modules (*requires python/Django skills*).

Join the mailing list ! Send an email to `geotrek-fr+subscribe@googlegroups.com` and you will receive an invitation automatically.

CHAPITRE 20

Development

— *Quickstart*
— *Install git hooks*
— *Adding or upgrade dependencies*
— *Model modification*
  — *Definition of Done for new model fields*
— *Check quality*
— *Run tests*
— *Setup to run rando synchronization locally*
— *Setup to use screamshotter-related features locally*
— *Database reset*
— *Restore existing Database*
— *Mapentity development*
— *UML diagrams of data model*

20.1 Quickstart

```
cp .env-dev.dist .env
# Edit .env if need be
cp docker-compose-dev.yml docker-compose.yml
docker-compose build
docker-compose run --rm web update.sh
docker-compose run --rm web load_data.sh
docker-compose run --rm web ./manage.py createsuperuser
docker-compose up -d
```

Go to http://geotrek.localhost:8000

PDF generation might not work unless you use this domain and is correctly set to SERVER_NAME variable in your .env file.

20.2 Install git hooks

— Several git hooks are available to prevent pushing to master branch or launch quality tests before committing. Install them with following commands :

```
ln -s -f ../../.githooks/pre-push .git/hooks/pre-push
ln -s -f ../../.githooks/pre-commit .git/hooks/pre-commit
```

20.3 Adding or upgrade dependencies

Consider using pip-tools to manage dependencies.

— add your dependency in setup.py for general dependency, dev-requirements.in for dev dependency, then run :

```
docker-compose run --rm web pip-compile
docker-compose run --rm web pip-compile dev-requirements.in
```

or

```
make deps
```

20.4 Model modification

```
docker-compose run --rm web ./manage.py makemigrations <appName>
docker-compose run --rm web ./manage.py migrate
```

Note : Add migration file to source control.

20.4.1 Definition of Done for new model fields

When updating or adding a new field `my_field` to a model `MyModel`, please proceed with the following changes to ensure this field is included in existing functionalities.

In `MyModel` class :

— If `my_field` is a `ForeignKey` :

— make sure to override `related_name` with an explicit set name

— if `my_field` causes cascading deletion (argument `on_delete=models.CASCADE`), make sure to log potential deletions (see example `log_cascade_deletion_from_sector_practice` in `geotrek/outdoor/models.py`)

— Make sure to set `verbose_name` on the field and add proper translations in `.po` files

Outside of model class :

— To display `my_field` in detail views, add a row in template `mymodel_detail_attributes.html`

— Look for form class `MyModelForm(CommonForm)` :

— If it exists, and field needs to be included in form, add `my_field` to form attributes (`fields` on the `Meta` class, sometimes `fieldslayout` as well).

— If field is added to the form **and is optional**, please add `my_field` to the documentation for hideable form fields : in `docs/advanced-configuration.rst` look for `HIDDEN_FORM_FIELDS['mymodel']` and add your field to the list.

- Look for list view class `MyModelList(CustomColumnsMixin, MapEntityList)` :
  - If it exists, please add `my_field` to the documentation for custom list view columns : in `docs/advanced-configuration.rst` look for `COLUMNS_LISTS['mymodel_view']` and add your field to the list.
  - If it exists, and if you wish to display a column for `my_field` in the list view for this model by default, simply add `my_field` to `default_extra_colums` on this class.
- Look for exports view class `MyModelFormatList(MapEntityFormat, MyModelList)` :
  - If it exists, please add `my_field` to the documentation for custom list exports columns : in `docs/advanced-configuration.rst` look for `COLUMNS_LISTS['mymodel_export']` and add your field to the list.
  - If it exists, and if you wish to display a column for `my_field` in CSV/SHP exports for this model by default, simply add `my_field` to `default_extra_colums` on this class.
- Follow the documentation you just edited to test that custom columns and hideable fields do work properly with your new field.
- Look for sql file defaults `geotrek/{app_name}/sql/post_90_defaults.sql` :
  - If it exists find your modelname in the list and depending on the default value alter column `my_field` or add `-- my_field`
  - If the modelname doesn't exist, create a new section (even if you don't need to alter column)
- Look for sql view file `geotrek/{app_name}/sql/post_20_views.sql` and update the view for your model with an alias for the new field

In API v2 :

If `MyModel` is served by APIv2, make sure to add a serializer for the new field in `geotrek/api/v2/serializers.py` and if you wish to filter on this field, create a new filter and add it to the right `ViewSet` under `geotrek/api/v2/views`, using attribute `filter_backends`.

When updating a field `my_field` in a model `MyModel` for `new_field`, check if this field is translated in `geotrek/{app}/translation.py`.

If so, you need to add a migration just after the migration generated by django. This migration should rename the old fields generated by modeltranslation `my_field_en` by `new_field_en` (example : `geotrek/trekking/migrations/0014_auto_20200228_2127.py`)

20.5 Check quality

Flake8

run :

```
docker-compose run --rm web flake8 geotrek
```

or

```
make flake8
```

20.6 Run tests

Django tests :

ENV variable must be set to run tests :

```
docker-compose run --rm -e ENV=tests web ./manage.py test
```

Test without dynamic segmentation :

```
docker-compose run --rm -e ENV=tests_nds web ./manage.py test
```

Cypress tests :

Create an empty project with docker :

```
docker-compose down
docker-compose up -d
```

Install elements for the cypress tests

```
make load_data
make load_test_integration
make load_test_integration_workflow
```

Move in cypress folder and install

```
cd cypress
npm ci
```

Launch tests

```
./node_modules/.bin/cypress run
```

Pictures of the problem and videos are generated in cypress/videos and cypress/screenshots

20.7 Setup to run rando synchronization locally

(sync rando is only relevant for an admin paired with geotrek rando v2)

In your django settings you must set the “url” key of SYNC_RANDO_OPTIONS to use the same domain as defined by SERVER_NAME in your .env.

For instance with SERVER_NAME=geotrek.localhost (default value)

```
SYNC_RANDO_OPTIONS = {
    'url': 'http://geotrek.localhost:8000'
}
```

20.8 Setup to use screamshotter-related features locally

Use the domain defined in SERVER_NAME in your .env to reach your local geotrek admin web instance. By default the address is *http ://geotrek.localhost :8000*.

20.9 Database reset

Data only :

```
docker-compose run --rm web ./manage.py flush
```

20.10 Restore existing Database

Assuming a dump of your database is located in your project directory :

```
docker-compose run --rm web pg_restore --clean --no-owner --no-acl -h $POSTGRES_HOST -U
↪$POSTGRES_USER -d $POSTGRES_DB /opt/geotrek-admin/<path_to_backup>.dump
```

Restore your `./var/conf/` project files, and data files into `./var/media`.

Then run a synchronization.

20.11 Mapentity development

See Django-Mapentity documentation

20.12 UML diagrams of data model

UML diagrams of Geotrek-admin data models are available in `docs/data-model` directory. To regenerate them from PostgreSQL, install postgresql-autodoc and graphviz Ubuntu packages and run `make uml`.

CHAPITRE 21

Translating

— *Getting started*
  — *Create an account*
  — *Browse by project*
  — *Browse by language*
  — *Select a coponent to translate*
— *Translation features*
— *Release translations (only for github repository managers)*
  — *Send modifications to Github repository*
  — *Add translations to next release*

Geotrek-admin can be translated online on Makina Corpus Weblate instance

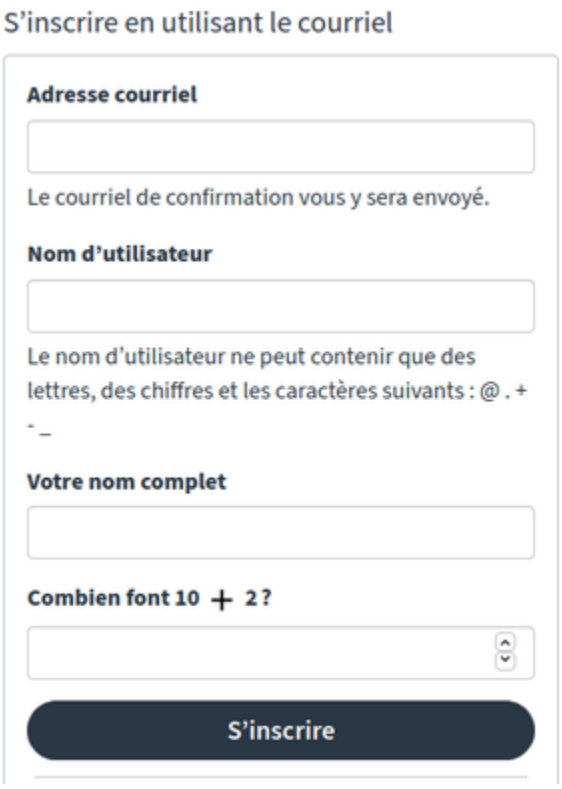
21.1 Getting started

— Create an account
— Browse by project
— Browse by language

21.1.1 Create an account

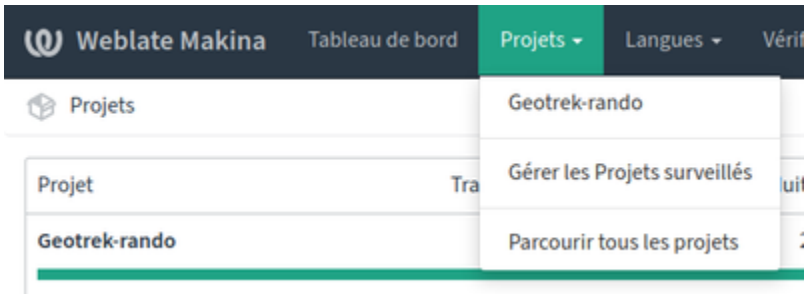
— Click on « Register »
— Fill the register form
— Validate your email
— Fill the password
— Then connect to weblate

Official documentation to create an account and manage your profile.

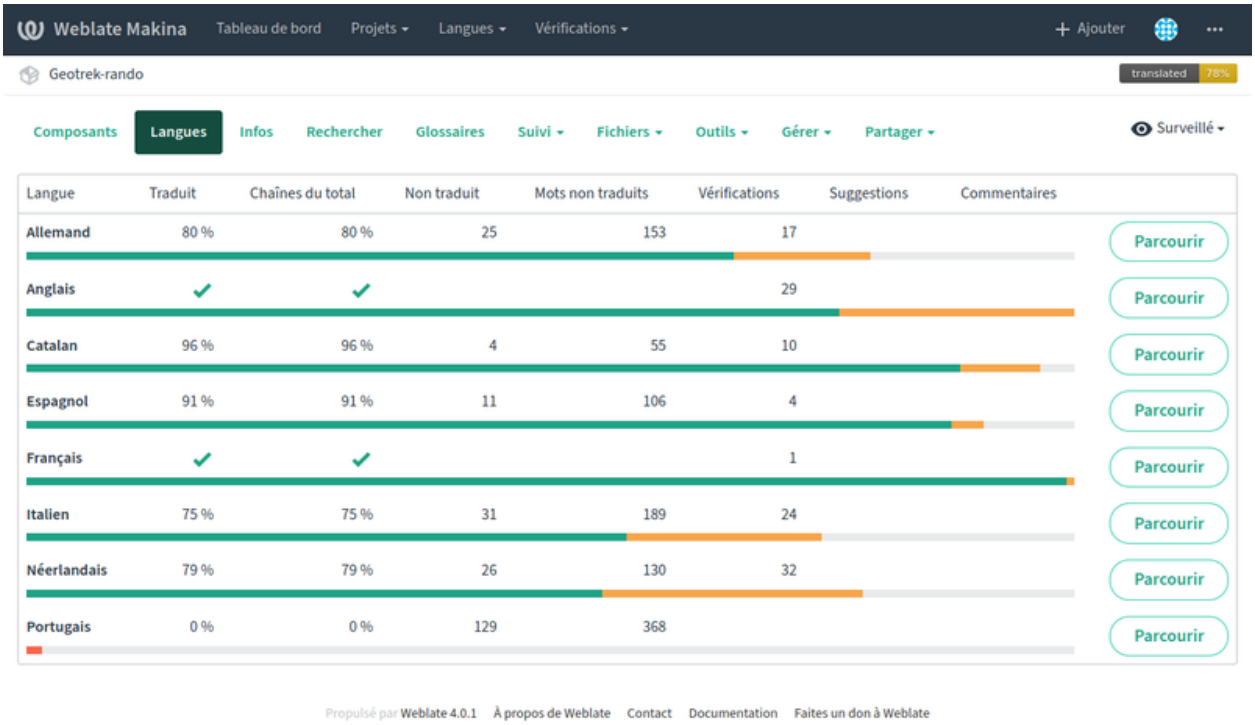


21.1.2 Browse by project

— Go to "Project > Browse all projects"



— Select Geotrek-admin project
— Click on tab "Languages"
— Choose the language to translate



21.1.3 Browse by language

You could also choose language first

- Go to “Languages > Browse all languages”
- Choose the language you want to translate
- Select the project to translate

21.1.4 Select a coponent to translate

- Select a component (a module or a piece of documentation)
- Click on « Translate » to start translating

And let’s go !

Weblate official documentation for translation.

21.2 Translation features

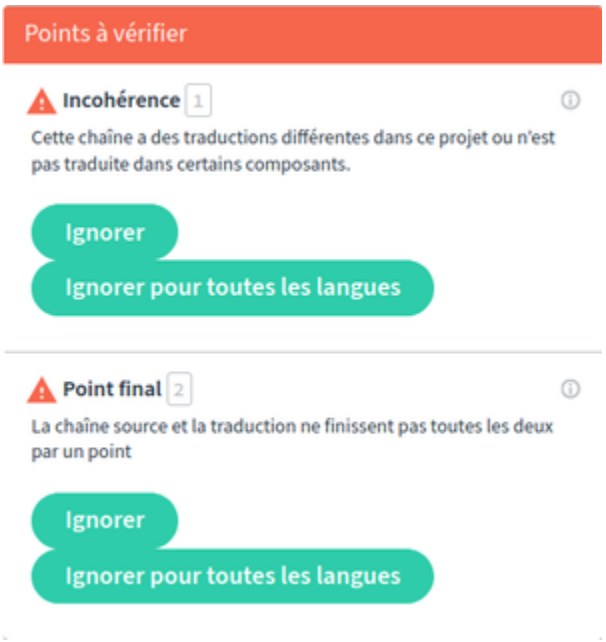
Weblate shows all translation by language and by module. Errors and left to translate can be displayed easily.

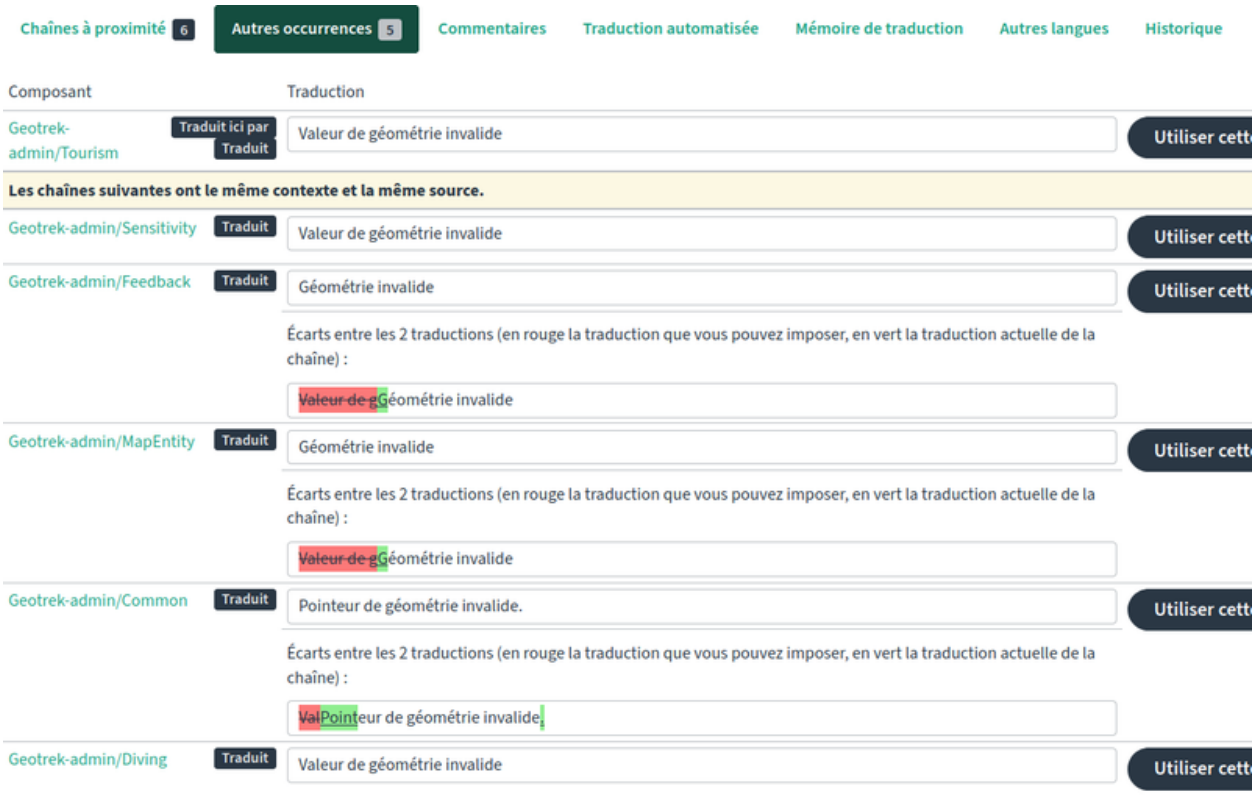
Weblate can identify problematic translations as chains between projects, punctuation inconsistancy.

Other occurrences in all components allows to check consistency.

Each translation generate a permalink (picto).

Weblate has a « Zen mode » showing only chains to translate.


Points à vérifier
Incohérence 1
Cette chaîne a des traductions différentes dans ce projet ou n'est pas traduite dans certains composants.
Ignorer
Ignorer pour toutes les langues
Point final 2
La chaîne source et la traduction ne finissent pas toutes les deux par un point
Ignorer
Ignorer pour toutes les langues


Chaînes à proximité 6
Autres occurrences 5
Commentaires
Traduction automatisée
Mémoire de traduction
Autres langues
Historique
Composant
Traduction
Geotrek-admin/Tourism
Traduit ici par Traduit
Valeur de géométrie invalide
Utiliser cett
Les chaînes suivantes ont le même contexte et la même source.
Geotrek-admin/Sensitivity
Traduit
Valeur de géométrie invalide
Utiliser cett
Geotrek-admin/Feedback
Traduit
Géométrie invalide
Utiliser cett
Écarts entre les 2 traductions (en rouge la traduction que vous pouvez imposer, en vert la traduction actuelle de la chaîne) :
Valeur de gGéométrie invalide
Geotrek-admin/MapEntity
Traduit
Géométrie invalide
Utiliser cett
Écarts entre les 2 traductions (en rouge la traduction que vous pouvez imposer, en vert la traduction actuelle de la chaîne) :
Valeur de gGéométrie invalide
Geotrek-admin/Common
Traduit
Pointeur de géométrie invalide.
Utiliser cett
Écarts entre les 2 traductions (en rouge la traduction que vous pouvez imposer, en vert la traduction actuelle de la chaîne) :
ValPointeur de géométrie invalide.
Geotrek-admin/Diving
Traduit
Valeur de géométrie invalide
Utiliser cett

21.3 Release translations (only for github repository managers)

Weblate send new translations to *translations* branch in Github, dedicated to translations. When new translations chains are validated, manager has to send them manually to Github.

For each release, *translations* branch must be merged into master before building the release.

21.3.1 Send modifications to Github repository

— In component, click « Manage > Repository maintenance »
— Click « Commit » to save translation in local repository
— Click « Push » to send local commits to *translations* branch in Github repository

21.3.2 Add translations to next release

1. In **Github**, merge *translations* branch into *master*, and update changelog.
2. After releasing, in **Weblate**, rebase the branche :
   — In the component, click « Manage > Repository maintenance »
   — Click « Rebase » to rebase *translations* branch onto *master*

CHAPITRE 22

Documentation

— *Translate documentation*

We use sphinx doc and sphinx-rtd-theme. Requirements are included.

A container based on sphinx image is created using docker-compose-dev.yml, documentation is built in watch mode thanks to sphinx-autobuild.

To compile and test documentation on local environment, run :

```
docker-compose up -d sphinx
```

Access to documentation built in html : http://0.0.0.0:8800

22.1 Translate documentation

— Generate .pot if needed

```
docker-compose run --rm sphinx make gettext
```

— Generate .po files

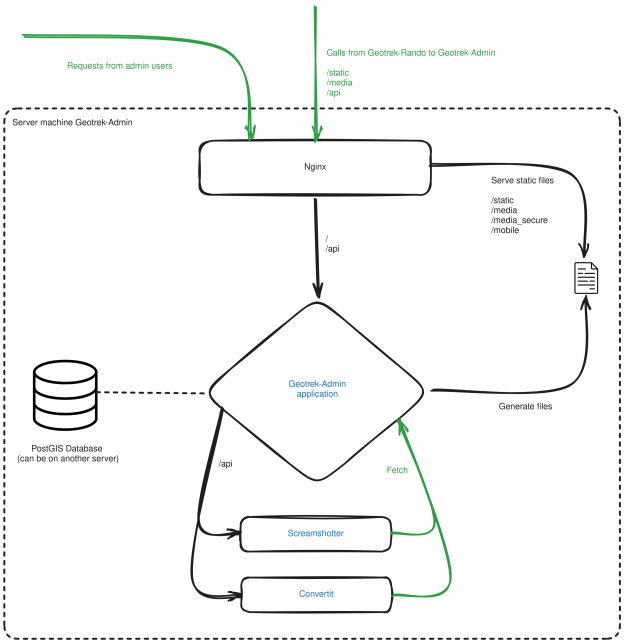
```
docker-compose run --rm sphinx sphinx-intl update -p _build/locale -l fr
```

CHAPITRE 23

Design

- *Architecture*
- *Main components*
- *Django conventions twists*
- *Main roles of PostgreSQL triggers*

23.1 Architecture

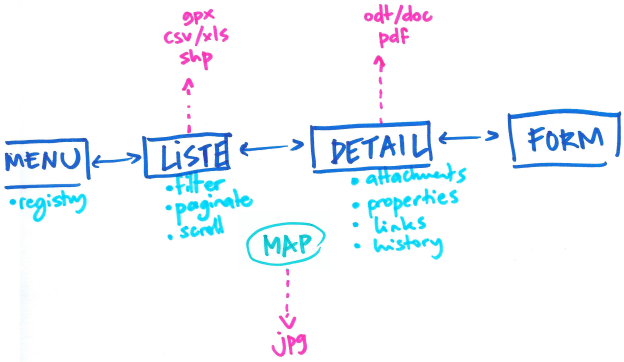


— **Geotrek-admin**, the web application

— **Convertit**, a Web API to convert document and image formats (*.odt to .doc*, *.svg to .png*)
— **Screamshotter**, a Web API to perform Web pages screenshots (*map image exports*).

23.2 Main components

The whole project is built on top of *mapentity*, a generic application in charge of :
— Menus and models registry
— List/Detail/Create/Update/Delete views
— Filtering
— Map images exports
— File attachment
— Document export
— Shapefile/GPX/CSV serializers
For a Django model, we use the registry to activate all views and menus :



Business specific notions are implemented in Geotrek-admin respective applications :
— **common** : shared concepts between all applications (*Organism*, *utils*, …)
— **authent** : groups, user, profile and structure notions. Optional external authent backend.
— **core** : paths, snapping, spatial referencing (topologies)
— **land** : physical types, competence, signage and work management
— **infrastructure** : buildings
— **signage** : signages and blades related content
— **maintenance** : interventions (on paths or on infrastructures) and projects
— **trekking** : POIs and treks
— **outdoor** : outdoor sites and courses
— **zoning** : static cities/districts/restricted areas layers
— **altimetry** : elevation, DEM
— **api** : manage all external communications interfaces (api v2, mobile)
— **cirkwi** : implementation of cirkwi related api
— **feedback** : feedbacks content, implementation of suricate workflow
— **flatpages** : flatpages models, used for publishing content on rando and mobile apps
— **sensitivity** : sensitive areas and species related content, link with Biodiv'Sport
— **tourism** : touristic contents and events, links with SIT (apidae, tourinsoft)

23.3 Django conventions twists

We have a couple of Django conventions infringements :

- SQL triggers everywhere : since Geotrek-admin database is to become the central storage component of all territory organisation data, it has to behave consistently whether data is modified through the Web application or raw access tools (pgadmin, QGIS). (For example, insertion & update timestamps, geometry computation or DEM wrapping.)
- Safe delete : update field `deleted = True` instead of performing actual delete in table. Requires every query-sets to be filtered on deleted. (**TODO** : use dango-safedelete, issue 813)

23.4 Main roles of PostgreSQL triggers

Automatic computation of fields :

- Date insert/update
- Geometry computation of linear referencing (topologies)
- DEM elevation wrapping (3D length, slope, etc.)

Topological path network :

- Split paths at intersection
- Snap paths extremities

CHAPITRE 24

Troubleshooting

— *Frequently encountered problems*
  — *Error 500 with* django.db.utils.IntegrityError … NOT NULL for column « language »
  — *Error 500 with document generation or map capture*
— *Signature check for debian packages*
— *Show main packages versions*

Geotrek-admin logs are stored in `/opt/geotrek-admin/var/log/geotrek.log` file.

But if Geotrek-admin does not start, take a look at systemd logs for each of the 3 Geotrek-admin services (user web interface, API and asynchronous tasks) :

```
sudo journalctl -eu geotrek-ui
sudo journalctl -eu geotrek-api
sudo journalctl -eu geotrek-celery
```

The output is paginated. With -e option you are at the end of the logs but you can go up an down with arrows. Type Q to quit. If you want to copy the log to a file, run :

```
sudo journalctl -u geotrek-ui > systemd-geotrek-ui.log
```

24.1 Frequently encountered problems

24.1.1 Error 500 with *django.db.utils.IntegrityError … NOT NULL for column « language »*

django.db.utils.IntegrityError : ERREUR : une valeur NULL viole la contrainte NOT NULL de la colonne « language »

This means specific migrations for translated fields have not been executed on database during update. You have to run them manually, classical migrations included :

```
geotrek migrate
geotrek sync_translation_fields
geotrek update_translation_fields
geotrek update_geotrek_permissions
geotrek update_post_migration_languages
```

24.1.2 Error 500 with document generation or map capture

Les captures de carte (utiles entre autre à la génération des documents et aux PDF) utilise un logiciel nommé screamshotter. Ce logiciel, pilote un navigateur web (chromium, via puppeteer) qui va appeler la page web de geotrek-admin de l'objet dont il doit réaliser une capture de carte.

ex : https://mon-geotrek-admin.fr/trek/1/

Il est donc nécessaire que cette URL lui soit accessible.

Paquet Debian :

— l'URL utilisée depuis le navigateur (https://mon-geotrek-admin.fr/trek/1/) doit être accessible depuis l'hôte de l'application geotrek-admin.

Image docker :

— l'URL utilisée depuis le navigateur (https://mon-geotrek-admin.fr/trek/1/) doit être accessible depuis le container de l'application screamshotter (et convertit).

Avertissement : Faites attention aux pare-feux, proxy et domaines privés. L'hôte (ou le container docker) doit pouvoir correctement résoudre l'adresse IP du domaine utilisé.

Sur certaines infrastructures, en particulier en entreprise ou derrière certains proxy, il se peut que la configuration de base empêche le bon fonctionnement.

Exemple :

— l'IP derriere le domaine demo-admin.geotrek.fr depuis mon poste de travail est 88.77.66.55, il s'affiche bien, mais depuis le serveur c'est 172.16.0.10, les captures de carte ne fonctionnent pas. Il faut jouer avec la configuration de la machine, les serveurs DNS, les paramètres network/alias ou extra_hosts du docker-compose.yml afin de le rendre accessible.

— le Proxy qui gère les certificats SSL ne transmet pas les bons en-têtes à geotrek-admin. De ce fait, geotrek-admin pense tourner en http et génère des url en http:// à screamshotter (ex : prend une capture de http://mon-geotrek-admin.fr/trek/1/ au lieu de https://mon-geotrek-admin.fr/trek/1/). Il faut penser à bien transmettre l'en-tête **X-Forwarded-Proto https** .

Comment débloquer le bon fonctionnement ?

Tout d'abord, vérifiez que vous utilisez bien les dernières versions des outils screamshotter et convertit.

```
sudo apt update
sudo apt install screamshotter convertit
```

Pour docker

```
docker compose pull screamshotter
docker compose pull convertit
```

puis relancer toute l'application (down / up)

Vérifiez que l'URL de votre geotrek-admin est accessible depuis le serveur ou le container.

```
wget https://mon-geotrek-admin.fr/trek/1/
```

Depuis docker :

```
docker compose run --user root --rm screamshotter bash
wget https://mon-geotrek-admin.fr/trek/1/
```

la réponse devrait ressembler à une page HTML de connexion.

Si ce n'est pas le cas, vérifiez l'IP du domaine

```
ping mon-geotrek-admin.fr
```

La réponse doit être une IP publique, idéalement la même que depuis votre poste de travail.

Testez la capture de carte depuis geotrek-admin, sur une carte, le bouton avec un appareil photo.

Si ça ne fonctionne pas, vérifiez le message d'erreur :

Request on http://screamshotter:8000/?url=http%3A//mon-geotrek-admin.fr/trek/1/xxxx failed (status=500)

On peut voir que l'URL est **http** et non **https**, c'est un problème d'en-tête non transmis. Il faut régler ça au niveau du proxy.

24.2 Signature check for debian packages

When you try to upgrade your Geotrek-admin, you can have problems with signature check :

```
An error occurred while checking the signature.
The repository is not updated and previous index files will be used.
GPG error: https://packages.geotrek.fr/ubuntu bionic InRelease: The following signatures␣
↪are invalid
```

You have to update the signature key to get the last update :

```
wget -O- "https://packages.geotrek.fr/geotrek.gpg.key" | sudo apt-key add -
```

24.3 Show main packages versions

To debug or to report a bug, you can check or provide in issue the main versions of Geotrek-admin and its dependencies :

```
geotrek check_versions
```

From docker :

```
docker compose run --rm web ./manage.py check_versions
```

Some examples :

```
geotrek check_versions --full
geotrek check_versions --postgis
```

Check command help for more options :

```
geotrek check_versions --help
```

CHAPITRE 25

Authors

— *Makina Corpus*
— *Parc national des Ecrins*
— *Others*
— *Credits*

25.1 Makina Corpus

— Gilles Bassière
— Sylvain Beorchia
— Mathieu Leplatre
— Anaïs Peyrucq
— Satya Azemar
— Simon Thépot
— Chloé Morisset
— Frédéric Bonifas
— Benjamin Marguin
— Eric Brehault
— Célian Garcia
— Gaël Utard
— Jean-Etienne Castagnede
— Timothée de Montety
— Bastien Potiron
— Emmanuelle Helly
— Célia Prat
— Paul Florence
— Marine Faucher
— Marc-Antoine Dupré
— Hugo Hasson
— Joaquim Nallar

25.2 Parc national des Ecrins

— Camille Monchicourt
— Théo Lechémia
— Élie Bouttier

25.3 Others

— Michaël Viadere (OpenIG)
— Guillaume Boitel
— Jean Lenormand
— Claude Paroz
— Amandine Sahl (Parc National des Cévennes)
— Frédéric Cloitre (LPO AuRA)
— Idrissa Djepa Creutz (OepnIG / Parc National des Cévennes)
— Yannick LE DUC
— Guillaume Boitel (Parc National de Port Cros)
— Pierre Loicq (Parc National de la Vanoise)
— Nathanaël (@roipoussiere)

25.4 Credits

— *Information desk* by Alex Berkowitz from The Noun Project
— *Calendar* by Laurent Patain from The Noun Project
— *Restaurant* designed by Roberto Pinto from the Noun Project
— *Kayak* designed by johanna from the Noun Project
— *Luggage* designed by Luis Prado from the Noun Project
— *Location* designed by Jacob Lowe from the Noun Project
— *Food* designed by Rémy Médard from the Noun Project
— *Climbing* designed by Eric Milet from the Noun Project
— *Box Office* designed by Alessio Damiano from the Noun Project
— *Steep Descent* designed by Sergey Patutin from the Noun Project
— *Landscape* by Adrien Coquet from the Noun Project
— *Ski course* by Sahua D from the Noun Project

CHAPITRE 26

CHANGELOG

26.1 2.103.2 (2024-03-22)

Bug fixes
- — Fix bug deleted blades still displayed on detail view of signages (fix #4003)
- — Fix interferences on *practice* mapping in Aggregator by changing calls order in GeotrekOutdoorParser (refs #3569)

Maintenance
- — Bump mapentity from 8.7.1 to 8.7.2

Development
- — Add git to Dockerfile build staging

26.2 2.103.1 (2024-03-15)

Maintenance
- — Bump mapentity from 8.7.0 to 8.7.1

Hot fix
- — Fix fonts in public PDF (docker image only)

26.3 2.103.0 (2024-03-14)

Bug fixes
- — Fix bug in « portals » filter on outdoor_rating endpoint in API V2 (fix #3997)

26.4 2.102.2 (2024-03-13)

WARNING ! Do not use - Causes bug in Geotrek-Rando-v3 exposing Outdoor data

New features
- Add *include_externals* filter to Cirkwi trek exports, to allow excluding treks with an external id (eid) (#3947)
- Tourism : add price to TouristicEvent model - ref #3587
- Add *check_versions* command to check Geotrek, Python, Django, PostgreSQL and PostGIS versions.
- Add `GeotrekCourseParser` and `GeotrekSiteParser` in Aggregator to retrieve Outdoor models (refs #3569)
- Add trail on leaflet overlay

Bug fixes
- Fix cache key for zoning cities
- Change signage group on leaflet overlay
- Add some translation on leaflet overlay layer

Improvements
- Add popup button to add organizer in touristic event form
- Change the *organizer* field of *TouristicEvent* model to a many to many field named *organizers* (#3587)
- Update favicon with current Geotrek logo
- Add intervention geometries union in projects database view (`v_projects`) (#3892)

Documentation
- Reorganize major sections in documentation, and add content

Minor fixes
- Add missing translation in intervention form (refs #3825)

Maintenance
- Upgrade django-mapentity to 8.7.1

26.5 2.102.1 (2024-02-20)

Hot fix
- Remove workforce cost (#3824)

26.6 2.102.0 (2024-02-19)

Minor changes
- Update all translation files

Bug fixes
- Signage & Blade conditions translations + admin BladeCondition entry (#3847)
- Add intervention date filter and add intervention years filter (#3825)

Documentation
- Improve performance in spatial intersection (zoning district and zoning city) for sql views (#3600)

New features
- Add UUIDs of parent and children `Courses` and `Sites` in APIv2 (#3569)

Improvements
- Add missing translations for fields on `Courses` and `Sites` in APIv2 (#3569)
- Allow Apidae Trek parser to handle traces not in utf-8
- Add workforce cost into intervention model (#3824)
- Add contractor to intervention model (#3820)

26.7 2.101.5 (2024-01-11)

New features
- Land : Add `CirculationEdge` model to manage circulation types and authorization types in the land module (#3578)
- Generalize``AccessMean`` model and add field `access` to `Intervention` (#3819)

Improvements
- Add rules fixture on sensitive area (#3470)
- Change condition on signage & blade to select many of them (#3847)
- Allow to set headers in requests from Parsers (#3861)
- Sort bladeType alphabetically (#3821)
- Update `Intervention` model to have begin & end date (#3825)

Documentation
- Improve documentation for database restore
- Improve documentation aboute suricate's workflow (#3070)
- Update layer section for maps since IGN updates

Maintenance
- Update `check_ign_keys` script to match new IGN urls
- Update `base.py` configuration for layers
- Add `merge_segmented_paths` command to find and merge paths (#3607)

Bug fixes
- Extract all geometry types in views *v_outdoor_sites* and *v_outdoor_courses* (#3603)
- Display only related interventions on project detail map (#3878)
- Maintenance appears several times on some zoning filters (#3881)
- Fix sub-language usage (en-US, zh-hant, …) (#3801)

26.8 2.101.4 (2023-11-15)

Bug fixes
- Fix : filters choices can raise exception in lists and not updated until application restart (#3812)
- Fix missing geometries for HD *view_points* in APIv2's */poi/* and */site/* routes (#3701)
- Fix cannot click on objects after customizing map styles (#3800)
- Fix profile elevation PNG generation by using cairosvg instead of convertit (#3833)

Documentation
- Improve help and doc for the *loadinfrastructure* command

26.9 2.101.3 (2023-10-26)

Bug fixes
- Fix *sync_rando* admin command failure if Trek has SVG attachment (#3803)
- Fix provider choices in list filter forms

26.10 2.101.2 (2023-10-17)

Bug fixes
— Fix Aggregator fails when updating Tour steps order (#3793)
— Fix services list display error (refs ##3795)

26.11 2.101.1 (2023-10-06)

Bug fixes
— POI cirkwi XML endpoint is fixed (2.101.0 regression) (#3783)

26.12 2.101.0 (2023-10-05)

New features
— Sensitivity : Add `openair` export format for aerial sensitive areas (#2372)
Bug fixes
— Fix missing update rights for Infrastructure Condition and Infrastructure Type with no structure in Admin Site (#3747)
— Allow to load a signage with the year set to None, raise error if set to NaN (#3611)
— Fix filters on Intervention exports (resolve #3749)
— Fix cities display on string (refs #3585)
Improvements
— Add organizer_id on TouristicEvent endpoint (#3587)
— Sort API V2 by begin date by default on touristic events (#3597)
Maintenance
— Upgrade *django-mapentity* to 8.6.1. New authentication system for screamshotter and convertit by token instead of IP detection.
— Refactor code for accessibility attachments
Documentation
— Add new contributors to the authors list in documentation
— Reorganize settings section (related to PR #3669)
— Update WYSIWYG link to help user when creating labels

26.13 2.100.2 (2023-09-12)

Improvements
— Remove “review” field on ServiceType (#1669)
Documentation
— Update loading_data section to add information about MNT values that needs to be integer (#1891)
— Add details and template for the pull requests process
— Update documentation fr translation files
— Uniformize documentation section
— Move user management section to user manual (#3709)
CI
— Reorganize generated release notes
Bug fixes
— Fix missing geometries for HD *view_points* in APIv2’s */trek/* route (#3701)
— Increase length size of label on TouristicEventOrganizer model to fix migrations problems (#3719)

26.14 2.100.1 (2023-09-05)

Documentation
— Replace broken link
Improvements
— Add rules data on `v_sensitivearea` view (#3613)
Clean
— Remove unused folder “bulkimport” from project (#3673)

26.15 2.100.0 (2023-09-05)

DO NOT USE

Bug fixes
— Fix : unable to search within a list of services (#3521)
— Fix : Unpublish trek in all languages when path is deleted (#1321)
— Fix : duplication on sites now does not duplicate children sites (#3665)
New features
— Filter trek and outdoor site labels according to whether they are published or not (#3529)
— Respond 404 JSON if page not found in API v2
Improvements
— Filter by multiple structures on Blades list (#3646)
— Add a multiselect to filter the Blades by more than one manager
— Filter by end date by default on touristic events in APIv2 (#3597)
— Add model LinePictogram for each line (#3327)
— Create Organizer model for touristic events, configurable in admin site (#3625)
— Improve CSS of the altitude profile of altimetry (#3657)
— Remove elliptic annotations from HD Views (they cannot be displayed on Leaflet)
— Serve GeoJS script locally
— To delete parent outdoor sites you must first delete their children (#3151)
Documentation
— Add configuration file for readthedocs
— Update architecture schema
Maintenance
— Upgrade *django-mapentity*

26.16 2.99.0 (2023-07-18)

New features
— Add field `access` to Signage and Infrastructure models (#3605)
— Enable filtering lists by objects IDs on APIv2 (#3458)
— Add information desks link on Treks with AggregatorParsers
— Add filter by manager to Blades module
— Add filter « Published » to outdoor course and outdoor site (#2810)
— Add a « district » attribute to views containing the « cities » attribute in API V2 (#3632)
— Make signage blade lines text optional (#3326)
— Add path information on API V2 about departure, arrival, comfort, source, networks, usages and stake (#3262)
Improvements
— Published by language depending on each portals and languages.
— Use default value with parsers when no value is found

— Improve filter popover (#2968)
— Add a scroll bar into filter form and module list (#2849)
— In projects, start year must be before end year (#3567)

Maintenance

— Upgrade *django-mapentity*

26.17 2.98.1 (2023-05-30)

Bug fixes

— Fix : Remove user group creation in Outdoor fixture (#3524)
— Fix : Configure nginx to invalidate mobile cache on language change
— Fix : service pictograms" URLs are made absolute in the API output of Trek descriptions (#3321)
— Fix : APIDAE Events parser now handles integer values for capacity (#3573)
— Fix : Configure *large_image* to use *libvips* even for PNG images (fixes HD Views for PNGs)
— Fix : Deleting signages must also delete their blades

Maintenance

— Upgrade *django-large-image* and *pip-tools*

Improvements

— Improve cascading deletions logic, and log them to LogEntry model to maintain history of deletions

26.18 2.98.0 (2023-03-27)

Bug fixes

— Fix : trekparser allowed to create trek with other geometry than linestrings
— Fix : do not prevent activity mappings overriding in subclasses of APIDAE Trek parser
— Fix permissions bypass structure was always needed on accessibility attachments (#3396)
— Fix default pictogram for mountainbike practice (it was blurry on mobile apps)
— Fix : *delete=True* mode now works for APIDAE Trek parser
— Fix missing insert and update date in fixtures for Sensitivity and Outdoor modules
— Fix target should not be ordonnable for interventions
— Fix : filter geometries on right geometry types in synchro mobile
— Fix : trek deletion was not possible without removing report link to this trek
— Fix : duplication attachments

Improvements

— Add arguments loadsignage : sealing / manager (#3377)
— Various minor improvements for APIDAE Trek parser
— The « near_xxx » API filters now use the topological link regarding topological objects. This will provide better performances for those endpoints when topologies are enabled. See Issues #3472 and #3505.
— Enable using Suricate workflow without moderation steps

Minor changes

— The « trek » API filter on POI and SensitiveArea list views now provide the same treatment as « near_trek » and is marked as deprecated.
— */api/v2/sensitive_area/?trek=123* now returns an empty list when trek does not exist instead of 404 - Not Found.
— */api/v2/sensitive_area/* results are no longer sorted by ID when the « trek » filter is used.
— */api/v2/sensitive_area/?trek=123* now uses the configured intersection margin for sensitive areas (previously returned intersections w/o margin).
— The « near_trek » API filter now removes from results the trek's excluded POIs.

Performances

— Improve performance zoning filter interventions

Documentation

— Update UML diagrams in documentation

26.19 2.97.4 (2023-03-09)

Performances
— Fix interventions list loading

26.20 2.97.3 (2023-02-28)

Bug fixes
— Fix : nearby sensitive areas now appears in outdoor details pages (and the other way too) (Issue #3494)
— Fix Interventions list datatable is empty
Improvements
— Set max zoom on HD Views depending on tiles depth

26.21 2.97.2 (2023-02-22)

Bug fixes
— Fix link between attachment and file is lost when updating old attachment without title and suffix

26.22 2.97.1 (2023-02-17)

Bug fixes
— Fix link between attachment and file is lost when updating old attachment without suffix

26.23 2.97.0 (2023-02-17)

New feature
— Add rules (with pictograms, descriptions and url) on regulatory sensitive areas (#3386)
Bug fixes
— Fix intervention filter when outdoor or signage is not installed
— Fix intervention's geojson
— Fix pictogram's for interventions on lands
Documentation
— Update Suricate documentation
— Add HD Views documentation
Security
— Add safety checks on uploaded files
Warning
— Attachment filenames are now suffixed with a random string. This might cause duplication of old attachment files that previously did not have a suffix. Make sure to run *clean_attachments* command regularly to save disk space.

26.24 2.96.1 (2022-02-02)

Bug fixes
— Fix APIv2 filters deteriorated performances
Improvements
— Sensitivity : Add missing attachments list to sensitive areas API

26.25 2.96.0 (2023-02-01)

DO NOT USE IT !

Warning
— APIv2 filters performances are deteriorated - Skip to 2.96.1 instead
New feature
— Handle very high resolution images (HD Views) that will automatically be tiled, for `Trek`, `POI` and `Site` (#3378)
— Handle annotations on HD Views (points, lines, polygons and text)
Improvements
— APIDAE Trek Parser output now shows APIDAE IDs of entities triggering warnings during import
— Update maximum request size in Nginx from 10M to 200M to allow uploading HD pictures (#3378)
Bug fixes
— Fix intervention datatable list if one intervention has no target
— Fix intervention datatable list with interventions on lands
— Fix signage's blade detail
— APIDAE Trek parser now raises an import error on geometry with not continuous segments
Development
— New contributing guide (docs/CONTRIBUTING.rst).
— Development dependencies are now split in dedicated file.
— pip-tools and flake8 are now available in developer environment.
— Dependency graph is now checked in CI (see docs/contribute/development to how add a new dependency).
— New git pre-commit hook to check all is alright before commit (see docs/contribute/development).
Warning
— The default Nginx configuration template has been improved (https://github.com/GeotrekCE/Geotrek-admin/pull/3298/commits/f9c72d95c1fd7eee2dee26dc73a5927966a812bf) to allow uploading big images. It is highly recommanded to apply changes to your Nginx configuration template (in /opt/geotrek-admin/var/conf/nginx.conf.in).

26.26 2.95.0 (2023-01-24)

New features
— Add possibility to duplicate objects with geometries
Minor improvements
— Add blade type on signage detail view (#3325)
Warning

Bionic (Ubuntu 18.04) instances need to install deadsnakes PPA to handle python3.8 updates :

```
apt-get install software-properties-common
```

```
add-apt-repository --yes ppa:deadsnakes/ppa
```

```
apt-get install python3.8
```

Maintenance

In preparation for HD Views developments (PR #3298)
- Bump Python to 3.8
- Bump MapEntity to 8.4.0
- Bump Pillow to 9.3.0
- Bump Celery to 5.2.1
- Bump django-celery-results to 2.4.0
- Bump django-clearcache to 1.2.1
- Add libvips to dependencies

Improvements
- Apidae trek parser supports geometry import from kml or kmz attachment
- More checks on Apidae trek parser in order not to import trek without a geometry

Bug fixes
- Fix loaddem command update other types of geometry
- Recreate cache folders if missing. (#3384)
- Modify site's geometry before saving to avoid edition and export of shapefiles (#3399)
- Fix API V2 cache key with X-Forwarded-Proto header (#3404)
- Check pictogram exist on categories during generation of pdfs
- Prevent « Internal Error » on API v2 when wrong url parameter is provided on courses and sites filter for pois
- Fix ApidaeParsers does not update every time
- Add fixtures licenses initial install
- Fix default conf nginx for mobile
- Replace image's relative URLs with absolute URLs in API v2 trek descriptions (#3321)
- Disable scroll propagation on layers list to avoid zoom changes on map (#2687)

26.27 2.94.0 (2022-12-12)

New feature
- New `LEIParser` to import touristic content and event from LEI touristic data system
- New `XMLParser` to import content from XML
- ApidaeTrekParser : import trek's contact info into description
- New `Parser` subclass to import POIs from the APIDAE touristic data system.
- New `POIParser` to import POIs from files (with and without dynamic segmentation)
- Change default color of imported filelayer (#306)

Bug fixes
- Fix shp zipfile import
- ApidaeTrekParser : round computed duration
- ApidaeTrekParser : fix attached pictures import

26.28 2.93.0 (2022-12-06)

New feature
- New `Parser` subclass to import treks from the APIDAE touristic data system.

Improvements
- Use MapEntity widget for geometries even without setting `TREKKING_TOPOLOGY_ENABLED` (to always display file layer leaflet plugin)

26.29 2.92.3 (2022-12-02)

Improvements

— API v2 : - revert `?trek filter` by direct intersecting geometry on sensitive area endpoint. - improve `?near_xxx` filters by direct intersecting buffered geometry on sensitive area endpoint.

26.30 2.92.2 (2022-12-01)

Bug fixes

— Fix cache management in API v2

26.31 2.92.1 (2022-12-01)

Improvements

— Show direction on lines with setting `DIRECTION_ON_LINES_ENABLED` in signage detail
— Add mobile nginx configuration directly on Geotrek-admin

Bug fixes

— Fix display lines on signage with setting `DIRECTION_ON_LINES_ENABLED`
— Show required's style for lines in blade form
— Fix cache management in API v2

26.32 2.92.0 (2022-11-29)

Warning

!!!! Clear cache after update. You can do this by going to admin panel, « clearcache » section, then delete default / fat and api_v2 !!!!

Improvements

— Cache API v2 Detail endpoints and themes list endpoint
— Sensitive areas are now computed with buffered geometries with settings SENSITIVE_AREA_INTERSECTION_MARGIN. Use ST_INTERSECTS on it is faster.
— Zoning informations are now cached until instance or zoning is updated.
— Show more decimal for coordinates in signage sql view

New feature

— Separate application and API v2 cache, ability to purge them with command or via admin

Bug fixes

— Check geom is valid before save
— Fix old migration script of Topology.geom (actually causes Django to falsely detect model changes not yet with a migration in NDS mode)
— Check that the Spatial Reference Identifier (SRID) unit is in meters before launching application (was during migration)
— Fix filter_type1 and filter_type2 for EspritParcParser when val is a list
— Fix « "NoneType" object is not iterable » when responseData is null for EspritParcParser

Documentation

— Fix parameter name `MAIL_MANAGERS` in documentation

26.33 2.91.1 (2022-11-18)

Bug fixes
— Fix flatpages can't be saved

26.34 2.91.0 (2022-11-17)

Minor improvements
— Add paths in overlays for elements which are not topologies
Bug fixes
— Add missing file field in Imports form layout
— Add missing help texts and validators on `TouristicEvent` `intervention_duration` and `preparation_duration`
— Fix flatpages can't be saved

26.35 2.90.1 (2022-11-04)

Bug fixes
— Prevent providers from APIv2 from overriding local providers when using `GeotrekParser`
— Add missing sources parsing to `GeotrekParser` (for `Trek`, `Touristic Content`, `Touristic Event`)

26.36 2.90.0 (2022-11-03)

New features
— Add new command to reorder pathaggregations of topologies
Bug fixes
— Fix APIv2 does not return sources related to published sites

26.37 2.89.1 (2022-10-20)

Bug fixes
— Prevent migration `0033_auto_20220929_0840` from failing by escaping Touristic Events `participant_number`
— Fix signage details page with DIRECTION_ON_LINES enabled (hide « Direction » column header)

26.38 2.89.0 (2022-10-20)

DO NOT USE IT !

Warning
— Migrations for Touristic Events can fail depending on data for `participant_number` - Skip to 2.89.1 instead
New features
— Add fields `preparation_duration`, `intervention_duration` to TouristicEvents
— Add new setting `DIRECTION_ON_LINES_ENABLED` to have the `direction` field on lines instead of blades

— Partially handle translated fields : when setting `fill_empty_translated_fields` to True, all empty translation fields for all languages will be set with the parsed value

Bug fixes

— Blade list view now takes into account custom columns from `COLUMNS_LISTS` setting
— Fix Suricate Workflow : do not unlock reports when resolving them
— Fix Suricate Workflow : display clickable links in report related emails

26.39 2.88.0 (2022-10-11)

DO NOT USE IT !

Warning

— Migrations for Touristic Events can fail depending on data for `participant_number` - Skip to 2.89.1 instead

New features

— Add optional places to TouristicEvents, using place selector to locate TouristicEvent on form map (#3266)
— Add fields `end_time`, `cancelled`, `cancellation_reason`, `bookable` and `place` to TouristicEvents (#3237)
— `cancellation_reason` selector is displayed in Event form if `bookable` is checked (#3237)
— `booking` text box is displayed in Event form if `bookable` is checked (#3237)
— Create `Assessment` tab in Event form to input retrospective information such as number of attendees per category (#3237)
— Create `TouristicEventParticipantCategory` model to define types of attendees for Events (#3237)

Breaking changes

— Rename `meeting_time` to `start_time` for TouristicEvent. APIv2 serialisation for TouristicEvent now exposes `start_time` instead of `meeting_time` (#3237)
— Rename `participant_number` to `capacity` for TouristicEvent. APIv2 serialisation for TouristicEvent now exposes `capacity` instead of `participant_number` (#3237)
— These fields are still available in API v2 for retrocompatibility but should not be used by default (#3237)
— If you have specific parsers importing into TouristicEvents, you should rename `meeting_time` to `start_time` and `participant_number` to `capacity` (#3237)

Bug fixes

— Fix TouristicEvent with no end dates are not returned in APIv2 (#3127)

Minor improvements

— Check `begin_date` is before `end_date` in TouristicEvent forms (#3237)
— Set `begin_date` not null for TouristicEvents (#3237)
— Change order of attributes in Event forms and detail view (#3237)
— Update Event SQL view `v_touristicevents` according to above changes (#3237)

Suricate Workflow (#2366)

— Show sentinel email addresses only to workflow manager

New features

— Add new setting *DIRECTION_ON_LINES_ENABLED* to have the *direction* field on lines instead of blades

26.40 2.87.2 (2022-09-23)

New features

— Add *default_language* attribute to Parsers to specify which language to update

Minor improvements

— Ensure attachments from parsers have generated thumbnails

Bug fixes

— Fix *provider* is not used properly when parsing TouristicContents
— Improve Aggregator translation management
— Fix PermissionError during sync-rando on fresh install from .deb package

26.41 2.87.1 (2022-09-20)

Bug fixes
— Fix acces rights on files after synchronization

26.42 2.87.0 (2022-09-20)

New features
— Add *provider* field to Trek, POI, Service, Signage, Infrastructure, TouristicContent, TouristicEvent, InformationDesk, Path, Trail, Course, Site, SensitiveArea (#3189)
— Add parser using api v2 (InformationDesk, TouristicContent, TouristicEvent, POI, Trek, Service, Signage, Infrastructure)
— Add aggregator parser with a conductor using json file
Minor improvements
— Disable debug log in debian package post installation script.
— Improve and fix error logging, now errors and warnings are logged to var/geotrek.log and console.
— Allow configuring email alerts for late reports (generalized from Suricate Workflow #2366)
Bug fixes
— Fix filtering on Services List does not filter
— Fix Site creation form is initialized with parent Site
— Fix memory leak and optimize SQL queries on zoning intersections
— Fix error message should not be displayed on attachments from the same structure as user
Maintenance
— Upgrade dependencies. The detail for the main dependencies :
  — django to 3.2.15
  — celery[redis] to 5.1.2
Suricate Workflow (#2366)
— Do not unlock reports when resolving them
— Improve Suricate workflow alert emails

26.43 2.86.0 (2022-09-05)

New features
— Add sync_rando / sync_mobile option *empty_tmp_folder* which will force deletion of all directories / files in tmp directory
— Add information desk uuid (#3189)
— Add setting `ALERT_DRAFT` which send mail whenever a path has been changed to draft (#2904)
— Add file type to attachments in API v2 (#3189)
— Add possibility to use different type of file with import form
— Add setting MAX_CHARACTERS for rich text fields with Mapentity 8.2.1 (#2901)
— Set map resizable with Mapentity 8.2.1 (#3162)
— Add Category, certification label and status fields on trails (#2900 & #3152)
Minor improvements
— Remove problems of tmp_sync_rando / tmp_sync_mobile which are not removed before new sync_rando / sync_mobile
— Change translation for Tag in Feedback module
— Change concatenation of null value for multiples values from "*" to "_" on sql views
— Prevent « Internal Error » on API v2 when wrong url parameter is provided
— Add "source", "portal", "labels" and "structure" to Cirkwi trek exports (#3220, #3164)
New ci

— New common interface github actions

Bug fixes

— Set relevant max zoom level for OpenTopoMap in the default config
— Fix fields filter for infrastructure

Maintenance

— Upgrade mapentity to 8.2.1

! Regression !

— System permissions on files output by *sync_rando* and *sync_mobile* commands were inadvertently changed to more restricted with no reading allowed by group or other. This may cause trouble if your deployment relies on those permissions. The original broader permissions have been restored with *v2.87.1*.

26.44 2.85.0 (2022-07-26)

New features

— Fix downgrade user permissions (is_staff, is_superuser) for external authent (#3156)
— Use permission bypass_structure on attachments and accessibility attachments (#2899)
— Add boolean field "display_in_legend" to Report Status model
— Add setting `ALERT_REVIEW` which send mail whenever an object has been changed to review (#2903)
— Add setting `PAPERCLIP_MAX_BYTES_SIZE_IMAGE` unallow usage of huge image (#2902)
— Add setting `PAPERCLIP_MIN_IMAGE_UPLOAD_WIDTH` unallow usage of images with small width (#2902)
— Add setting `PAPERCLIP_MIN_IMAGE_UPLOAD_HEIGHT` unallow usage of images with small height (#2902)
These settings will influence the attachments downloaded in parsers

Documentation

— Add tutorial to visualize sql views in Qgis
— Add sql views for Qgis

Bug fixes

— Fix api v2 services are published by type
— Fix form outdoor/trekking when rating scale is used with modification of practice
— Fix initial value of rating was not shown in trekking form (#3121)

Suricate Workflow (#2366)

— Add form field to enter messages for administrators in Report Workflow Mode
— Improve Suricate workflow alert emails

26.45 2.84.1 (2022-06-21)

Bug fixes

— Fix length_2d or land's app for exports and lists

26.46 2.84.0 (2022-06-20)

New features

— Add filter valid geometries on topologies (#2515)[3.1]
— Add setting `ALLOW_PATH_DELETION_TOPOLOGY` which protect or not against deletion of path with topologies linked to it (#2515)[3.3.1]
— Add eid on InformationDesk
— Add parser InformationDesk for Apidae
— Add accessibility on Infrastructure in api v2

Minor improvements

— Add length 2d for land's app for exports and lists (#2976)

- Add option to recalculate altimetry with `loaddem` command

Bug fixes

- Log entry menu is now only displayed if user has permission (#3130)
- Admin menu is now only displayed if user has permission (#3130)
- Object "All history" button is now only displayed if user has permission (#3130)
- Error 404 default template now display a visible message
- Error 500 default template doesn't make recursive exceptions anymore
- Log entry permissions are now managed by « mapentity - xxx log entries » instead of « admin - xxx log entries »
- Fix information desk filter when outdoor module is not available (#3135)
- Fix APIv2 does not return labels and themes on published outdoor sites

Breaking Changes

- This release requires PostGIS 2.5 or later.
  - Ubuntu bionic 18.04 users, take care, PostGIS default is 2.4. You need to upgrade your PostGIS version.
    - See documentation https://geotrek.readthedocs.io/en/latest/install/installation.html#ubuntu-bionic-postgis-2.5-upgrade)

Warning

- From now, Geotrek-admin is not installable on Ubuntu 18.04 bionic anymore. But upgrade are still available.
- The default Nginx configuration template has been improved to increase data loading performances. It is highly recommanded to apply changes to your Nginx configuration template (in `/opt/geotrek-admin/var/conf/nginx.conf.in`).

Improvments

- New GeoJSON generation system, using Django Rest Framework and PostGIS functions (#2967)
- Enable GZIP compression on JSON / GeoJSON by Nginx

Maintenance

- Upgrade mapentity to 8.1.2

26.47 2.83.0 (2022-06-01)

New features

- Display link to attachment in admin site for attachments
- Add license field on attachments (#3089) [thanks to Paul Florence]
- If `COMPLETENESS_FIELDS` is set for a model an object is published, display completeness fields if missing on page detail (#2898)
- Avoid publication or review if `COMPLETENESS_FIELDS` is set for a model, and `COMPLETENESS_LEVEL` is one of "error_on_publication" and "error_on_review" (#2898)

Bug fixes

- Fix APIv2 does not return information desks on published outdoor sites(#3095)
- Fix trail detail link in list view
- Fix infrastructure detail link in list view
- Fix dive detail link in list view
- Fix signage and infrastructure attachment access if published

Documentation

- Improve import from file section

Minor improvements

- Add image widget to tinymce editors by default
- Delete filenames in captions of attachments when importing from Apidae (#2698)
- Add copyright when importing from Apidae on attachments (#2698)
- Improve basic fixture for Feedback app allowing to initialize Report form in one go

Maintenance

- Add a git hook to prevent pushing to master.
- Update to paperclip 2.5.0

26.48 2.82.2 (2022-04-28)

Bug fixes
- Prevent exceptions on malformed images when launching `sync_suricate` command
- Fix alert on Project list view

26.49 2.82.1 (2022-04-28)

WARNING ! Do not use, list view for Projects raises Datatable alert

Bug fixes
- Fix display objects with wrong colors when `ENABLE_REPORT_COLORS_PER_STATUS` is True

26.50 2.82.0 (2022-04-27)

WARNING !

Do not use, or set `ENABLE_REPORT_COLORS_PER_STATUS` to False, else objects will not be displayed properly on map - Release 2.82.1 should be used instead

New features
- Server-side list pagination. Better performance for large lists (#2967)
- Add overlays for objects from Trekking, Maintenance, Infrastructure and Feedback modules (#1300)

Minor improvements
- Refer to Reports by a label instead of email addresses
- Increase default cache expiration from 8hours to 30days (#2967)
- Use distance from setting `SENSITIVE_AREA_INTERSECTION_MARGIN` in sensitive area filter *trek* in api v2

Bug fixes
- Fix filter `trek` in api v2 for information desks
- Fix filter `trek` in api v2 for pois with setting `TREKKING_TOPOLOGY_ENABLED` (#3054)

Maintenance
- Update to mapentity 8.0.1

Suricate Workflow (#2366)
- Add `assigned_user` field to Report model
- Add `color` field to Report Status model
- Add TimerEvent class, used to alert Report supervisors when timer expires, with `check_timers` command
- Force workflow when `SURICATE_WORKFLOW_ENABLED` setting is enabled
- Add setting `ENABLE_REPORT_COLORS_PER_STATUS` to display different colors in status list view
- Add editable predefined emails
- Display only some reports depending on which user is logged in
- Add City and District information to Report detail page
- Alert user about synchronization problems in Suricate Workflow mode

26.51 2.81.0 (2022-04-11)

New features
— Add SQL default values directly on most tables of the database (#3008)
Minor improvements
— Rename French field names of attachment and accessibility attachment tables (author, legend, title)
— Improve pdf for sites, courses
Maintenance
— Update to paperclip 2.4.3

26.52 2.80.0 (2022-04-05)

Minor improvements
— Improve pdf for sites, courses
— Add a new parameter in parsers, allowing to add multiple values to fields from multiple parsers (#2091)
— Add locale altimetry filters
— Change order list actions and add new signage in signage module (#2852)
Bug fixes
— Fix templates map and image
— Fix trekking's template elevation was not on the right
— Show accessibility block only with datas in accessibility
— Compile messages of every apps
— Fix required language in form is ignored from configuration
— Fix link initial mode is now File (#3001)
— Fix line topologies drawing sometimes fails on some paths
— Fix poi's csv generation of elements from other modules (#2286)
— Fix pdfs booklet outdoor
— Fix api v2 schema targets (GTRV3#607)
— Fix api v2 translation schema targets (values should not be in french)
Maintenance
— Fix required language in form is ignored from configuration
— Allow configuring scheme forwarding though proxy
— Update to paperclip 2.4.2
WARNING !

If an error occurred while checking the signature for debian packaging check troubleshooting section for additional informations

26.53 2.79.0 (2022-03-25)

New features
— Add public booklet pdf for courses, sites, events, contents, dives
— Improve treks pdf templates and add new accessibility fields (#2838)
WARNING !

Check your custom trekking's templates, blocks order changed. There is a huge new block accessibility. Disabled infrastructure's block have been removed

Bug fixes
— Fix maps height when height is bigger than width in treks pdf (#2746)

26.54 2.78.0 (2022-03-22)

New features
— Ability to customize public PDF by portal (#2691)
Minor improvements
— Add block logo in public PDF templates
Bug fixes
— Fix pdf booklet use the right template

26.55 2.77.3 (2022-03-18)

Minor improvements
— Add *only_filters* filter api v2 for labels (#3002)
— Add filter labels_exclude for api v2 allowing to exclude particular label on treks, sites
Bug fixes
— Fix parser biodiv didn't collect all sensitive areas (#2966)
— Fix attachments external links (#3001)
Maintenance
— Update to paperclip 2.4.1

26.56 2.77.2 (2022-03-15)

Bug fixes
— Fix migration 2.77.1 publication
Minor improvements
— Add publication informations by lang on infrastructure
— Remove table Infrastructure on infrastructure
— Fix Intervention detail page breaks when target is a Report
— Add translation signage

26.57 2.77.1 (2022-03-11)

Minor improvements
— Show all infrastructures and signages on interventions (#2851)
Bug fixes
— Show trail and path on intervention (#2851)
— Remove duplicate id POI export (#2893)
— Fix migration 2.77.0 publication

26.58 2.77.0 (2022-03-09)

DO NOT USE IT !

New features
- Add filter label sites outdoor api v2
- Add accessibility field on Infrastructure

Minor improvements
- Add ratings_description field in export (#2755)

Bug fixes
- Remove width and height in SVG generating problems in Geotrek-rando V3 by Camille Monchicourt
- Fix labels filter api v2 (#2764)
- Fix linebreaks template detail

Maintenance
- Update to mapentity 7.1.3

26.59 2.76.4 (2022-03-07)

Minor improvements
- Move fields in forms and details (#2755)
- Add information rating scale in csv for treks (#2755)

26.60 2.76.3 (2022-02-09)

Documentation
- Fix documentation trek with gear and not equipments

Bug fixes
- Fix css caption detail
- Fix ACCESSIBILITY_ATTACHMENTS_ENABLED setting work as intended
- Fix attachment translations
- Facilitate the comprehension of the difference between fields label_accessibility and approved in touristic content detail
- Fix migration translations equipment and disabled_infrastructure

26.61 2.76.2 (2022-02-08)

DO NOT USE IT !

Bug fixes
- Remove multiple choice ratings by rating scale for treks
- Fix translations equipment and disabled_infrastructure are recovered for gear and accessibility_infrastructure

26.62 2.76.1 (2022-02-07)

New features
- Add ACCESSIBILITY_ATTACHMENTS_ENABLED setting allowing to disable/enable menu attachments for accessibility
- Add accessibility field on sites (#2838)
- Change field disabled_infrastructure for accessibility_infrastructure (#2838)

Minor improvements
- Text pasted in rich text fields (TinyMCE) are now cleaned up.
- Facilitate the comprehension of the difference between fields label_accessibility and approved in tourism (#2838)
- Move trek DEM serialization to APIv2 (for 3D view)
- Move trek altimetry profile serialization to APIv2
- Change fixture rating trekking
- Move gear field form and detail (#2838)

Performances
- Fix DEM cache does not invalidate on trek update

Bug fixes
- Fix translation equipment api v2 courses

26.63 2.76.0 (2022-02-02)

New features
- Add ratings, rating scales fields on trekking (#2755)
- Add equipments field on trekking (#2845)
- Add filters altimetry on all apps
- Add accessibility attachments on trekking (#2838)
- Add accessibility field on courses (#2838)
- Add accessibility field on touristic content (#2838)
- Add accessibility field on information desks (#2838)
- Add label accessibility field on touristic content and informations desks (#2838)
- Add information desk type api v2

Minor improvements
- Add translations NL, ES, DE, IT, EN for all apps
- Change admin translations fields, add tab (#2892)
- All rich text fields are updated according new TinyMCE theme.
- Improve API v2 POI serializer to include type labels and pictograms

Maintenance
- Update to mapentity 7.1.0
- Update to django-tinymce 3.4.0 and TinyMCE 5.10.1

Bug fixes
- Fix missing trademark (#2921)
- Fix bootstrap theme in warning and error messages or alerts (#2872)
- Fix search in infrastructure admin panels (#2924)
- Fix APIv2 nearby content filter throws exceptions when queried for missing data (#2926)
- Prevent exceptions when parsers receive integers instead of strings

Performances
- Add missing indexes on geometry fields (WARNING, if you add indexes manually you should delete them before applying migrations) (#2933)

26.64 2.75.0 (2022-01-07)

Tools
— Update check_ign_key tool
New features
— Add new group external authent *EDITOR_TREKKING_MANAGEMENT* (#2842)
Bug fixes
— Fix bootstrap theme in warning and error messages or alerts
— Fix Services external IDs were not displayed in detail pages
— Fix interventions filtering on zonings (#2766)
— Fix interventions shapefiles with *ENABLE_JOBS_COSTS_DETAILED_EXPORT* setting (#1798)
— Fix projects on interventions with GeometryCollection's geometry
— Fix parser when DatabaseError occurs
— Add customization columns *COLUMNS_LISTS* on every models listed in documentation (#2688)
Minor improvements
— Add filtering portals sync_mobile for touristic contents and events (#1941)
Maintenance
— Update to mapentity 7.0.6

26.65 2.74.1 (2021-12-21)

Bug fixes
— Fix blank line due to mapentity template error
Maintenance
— Update to mapentity 7.0.5

26.66 2.74.0 (2021-12-17)

Minor improvements
— Show every paths in intervention csv (#2711)
— Hide signage/blade dropdown-toggle with settings `BLADE_ENABLED=False` (#2852)
— Remove urls blade with settings `BLADE_ENABLED=False` (#2852)
Bug fixes
— Fix multiple forms in formsets deletion (#2693)
— Fix access to pictures generated with watermark (#2840)
— Fix intervention creation and update is now scrollable after merging tabs (#2712)
— Fix restricted area and restricted area type filters on intervention (#2766)
New features
— Allow to filter Cirkwi `circuits.xml` and `pois.xml` API with portals and structures (#2822)
— Add restricted area and restricted area type filters on projects (#2766)
— Add `reservation_id` in `/trek` API v2 (#2817)

26.67 2.73.0 (2021-12-13)

Bug fixes
— Fix formset item deletion raises error in forms (#2693)
Refactoring
— MapEntity is now a separate dependency (https://github.com/makinacorpus/django-mapentity)
New features
— Optimize Path caching in edition views (#2847)
— Filter list views by Restricted Area as well as by Restricted Area Type (#2766)
— Add `BLADE_ENABLED` setting to hide Blade in Signage forms and in Signage detail page (#2852)
— Add `LINE_ENABLED` setting to hide Line in Blade forms and in Blade detail page (#2852)
— Add `PAPERCLIP_RESIZE_ATTACHMENTS_ON_UPLOAD` setting to enable resize attachments on upload (#2835)
— Add `PAPERCLIP_MAX_ATTACHMENT_WIDTH` and `PAPERCLIP_MAX_ATTACHMENT_HEIGHT` to configure attachment resizing (defaults 1280px) (#2835)
— Use `MAPENTITY_CONFIG` setting to configure map style on list and detail views (#2554)
User interface
— Clarify Land Edge module browsing (#1404)
— Renamed « Tronçons physique » to « Types de voie », « Tronçons de compétence » to « Compétence sentiers », « Tronçons de gestion de travaux » to « Gestionnaire travaux », « Tronçons de gestion signalétique » to « Gestionnaire signalétique » (#1301)
— Renamed « zonage réglementaire » to « zonage » (#2766)
Minor improvements
— Merge tabs in Intervention forms (#2712)
— Make targets display more specific in Interventions exports (#2711)
— Improve support for Tourinsoft v3 with new medias management
Bug fix
— Fix TopologyException when filtering objects by several RestrictedAreaTypes

26.68 2.72.0 (2021-11-16)

New features
— APIv2 : Add `attachment` field to Touristic Event serialization
Minor improvements
— Add possibility to fill `code` field in Signage model when using `loadsignage` command. Two parameters added : `code_field` and `code_default`
Bug fixes
— Prevent Signages and Infrastructures from being displayed on PDFs when unpublished
— Database : fix SQL cleanup that delete foreign key on `core_pathaggregation.path_id -> core_path.id` (#2819)
— Fix generation altimetry profile (`dem.json`)

26.69 2.71.0 (2021-11-03)

New features
— APIv2 : Add filter by portal on outdoor practices and ratings
Bug fixes
— APIv2 : Fix exceptions on filter by portals or themes in Outdoor Course route

26.70 2.70.0 (2021-11-02)

New features
— Add UUIDS to the following objects, and to APIv2 serialization for those included : Path, TouristicContent, TouristicEvent, Outdoor Site, Outdoor Course, Attachment, and Topology (inherited by POI, Trek, Service, Trail, Signage, Infrastructure, PhysicalEdge, CompetenceEdge, LandEdge)
— APIv2 : Add pictograms to outdoor practice routes
— APIv2 : Add cities to outdoor sites and outdoor courses routes
— APIv2 : Add filter by themes, cities, districts, types, and structures to outdoor sites and outdoor courses routes
— APIv2 : Change Web Links serialization on outdoor sites routes, to detailed instead of just an id
Breaking changes
— Geotrek-admin now needs PostgreSQL extension "pgrypto".
WARNING !

Before upgrading to this version make sure to run `CREATE EXTENSION IF NOT EXISTS "pgcrypto";` from `postgres` user in database.

```
su postgres -c "psql -q -d $POSTGRES_DB -c 'CREATE EXTENSION pgcrypto;'"
```

26.71 2.69.0 (2021-10-22)

New features
— Add public PDFs to Outdoor Course and Outdoor Site, with templates

26.72 2.68.1 (2021-10-21)

Bug fixes
— Fix error 404 on CSS from 2.68.0

26.73 2.68.0 (2021-10-20)

DO NOT USE IT !

New features
— Link an Outdoor Course to multiple parent Sites instead of one
— Added notion of points of reference for Outdoor Courses. (Can be disabled with `OUTDOOR_COURSE_POINTS_OF_REFERENCE_ENABLED = False`)
Breaking change
— APIv2 serialisation for Courses now exposes `sites` instead of `site`
Bug fixes
— Fix translations for Site and Course filters in Interventions list view

— Fix bug that auto-confirms the modal when launching a synchronization (bug introduced with bootstrap migration)

User Interface

— Display children Sites above parent Sites in Outdoor Sites list view

26.74 2.67.0 (2021-10-12)

New features

— APIv2 : Add "children" and "parent" fields to Outdoor Site serialization
— APIv2 : Add filter by pratices on outdoor courses
— Filter interventions by Outdoor model targets in Intervention module's list view

User Interface

— Distinguish Sites from Courses in Outdoor tree display thanks to bullets
— Display full Sites hierarchy in Outdoor detail views

Bug fixes

— Fix nearby Courses and nearby Sites display in Outdoor detail pages
— Fix Outdoor migrations fail on empty database
— Fix sync_mobile does not check for published or unpublished treks

26.75 2.66.0 (2021-09-27)

New features

— APIv2 : Add filter by ratings on outdoor courses and sites
— APIv2 : Add filter by pratices in hierarchy on outdoor courses and sites
— APIv2 : Add filter by ratings in hierarchy on outdoor courses and sites
— Display children sites" ratings in site page
— APIv2 : Add "sector" and "attachment" fields to Outdoor Site serialization
— Add DISPLAY_COORDS_AS_DECIMALS setting to format coordinates as decimal degrees instead of degrees minutes seconds
— Enable translations on "equipment" field on Outdoor Course

Bug fixes

— Fix dynamic forms on outdoor cotations display all cotations when selector empty
— Hide excluded POIs on Outdoor Site and Course detail pages

User Interface

— Sort sites by alphabetical order in outdoor course forms

26.76 2.65.0 (2021-09-21)

New features

— APIv2 : Add filter on Outdoor Site route to only retrieve root sites from hierarchy
— Add fields "duration", "type", "gear", "ratings_description" to Outdoor Course
— Add fields on APIv2 for Course model : "min_elevation", "max_elevation", "children", "parents", "attachments"
— Add excluded_pois on Course and Site models.
— Add filter on APIVv2 POI endpoint to retrieve pois related to Course or Site
— Replace Outdoor Site "ratings_min" and "ratings_max" fields with "ratings"
— Make Outdoor Site and Course "ratings" form fields dynamically change on practice selection
— APIv2 : Add children courses to sites" serialization
— Add Course Type management to admin site

26.77 2.64.0 (2021-09-14)

New features
- Add endpoints for infrastructure and related types in APIv2
- Add endpoints for signage and related types in APIv2
- Filter TouristicContentTypes according to published content in APIv2

Bug fixes
- Fix missing translations for infrastructure difficulty levels in admin
- Fix impossible import of uninstalled module "sensitivity" in "dive"

26.78 2.63.0 (2021-09-03)

New features
- Add difficulty level fields (usage and maintenance) to infrastructure
- Add "active" field to job model, and hide inactive jobs in forms
- Add detailed jobs costs to interventions exports, with a new column for each job
- Add SURICATE_MANAGEMENT_ENABLED setting
- Add SURICATE_MANAGEMENT_SETTINGS setting to configure second Suricate API
- Add helper to make requests to Suricate
- Add parser to retrieve statuses, activities, and reports (in bounding box) from Suricate
- Add sync_suricate command to retrieve Suricate data
- Change Report model to use one of 3 modes : No Suricate, Suricate Report or Suricate Management (SURICATE_REPORT_ENABLED and SURICATE_MANAGEMENT_ENABLED settings)
- Generalize existing filters in APIv2
- Add "near_outdoorsite" and "near_outdoorcourse" filters in APIv2
- Add "created_before", "updated_before", "created_after" and "updated_after" filters in APIv2
- Add route to APIv2 to retrieve Geotrek version
- Add API_V2_DESCRIPTION setting to change description text in API v2 Swagger page
- Add endpoints for services in APIv2 : service, service type
- Add link between reports and interventions

Bug fixes
- Fix length_kilometer attribute computation in treks
- Fix date update format in lists
- Add CORS header to access medias
- Change geographic intersection calculation from annoted queries to optimized build-in method

26.79 2.62.0 (2021-07-06)

New features
- Add custom columns configuration to list views
- Add custom columns configuration to list CSV exports
- Add custom form fields configuration to creation views

Bug fixes
- Fix filter difficulty in API v2

26.80 2.61.1 (2021-06-28)

Bug fixes
— Fix filter in_bbox in API v2

26.81 2.61.0 (2021-06-25)

New features
— Add Web Links to Trek endpoints in APIv2
— Add endpoints for Web Links categories in APIv2
— Ensure APIv2 returns outdoor sites list and outdoor courses list as ordered by localized name

26.82 2.60.0 (2021-06-25)

New features
— Add endpoints for user feedback in APIv2 : report category, report activity, report problem magnitude, and report status
— Ensure APIv2 returns treks list and touristic contents list as ordered by localized name
Bug fixes
— Fix confirm delete attachment modal not visible
— Fix required "*geom" position
— Fix scroll unwanted when list is full
— Fix responsive on dataTables
— Remove excluded POIs from results in POI endpoint on api v2 when filtering by trek id
— Sort attachments listed in api v2 endpoints for Trek, TouristicContent, POI
— Ensure content is displayed only when a related object is published on api v2
— Exclude deleted content of portal filters in api v2
Maintenance
— Update to paperclip 2.3.2

26.83 2.59.0 (2021-06-07)

Breaking Change
— Template nginx.conf.in was changed to work with multiple rando portals (#2670).
First, if you changed file */opt/geotrek-admin/var/conf/nginx.conf.in*, back it up somewhere. 1 - While installing, choose "Y" to get the new version. 2 - Copy your changes from the backed up file to the new version. 3 - Execute : *dpkg-reconfigure geotrek-admin* to reapply your customization.

User Interface
— Important visual changes due to CSS framework upgrade
— Improve responsive
Maintenance
— Upgrade Bootstrap to 4.6
— Upgrade JQuery to 1.9.1
— Upgrade DataTables to 1.10.23
— Upgrade Chosen to 1.2.0
— Move to vendor folder updated JS Libraries used by Mapentity
— Update HTML markup in many templates, and update tests too
— Expired sessions stored in database are now deleted at each update

Bug fixes
— Fix gpx/kml are not generated on all languages (The first object was working).

26.84 2.58.0 (2021-05-20)

Documentation
— Add documentation ssl
New features
— Mobile API returns multiple pictures for objects like Treks and POIs. Can be configurated with MOBILE_NUMBER_PICTURES_SYNC setting.
— Add filter bad topologies and geoms
Bug fixes
— Fix DistanceToPointFilter usage in API v2
— Fix pdf/gpx/kml are not generated on all languages

26.85 2.57.0 (2021-04-28)

New features
— Add managers field to outdoor sites
Bug fixes
— Fix projection of departure_geom in API v2

26.86 2.56.0 (2021-04-27)

Bug fixes
— Fix API v2 crash when trek geom is a point
New features
— Add outdoor course endpoint to API v2
— Add all fields to outdoor site/course exports (csv/gpx/shp)
— Link outdoor sites and courses to other objects, especially POIs, infrastructures and interventions
Documentation
— Update database ULM schemas (with outdoor)
— Update faq.rst
— Proofreading
Maintenance
— Update parser for Esprit Parc National data streams
— Upgrade Weasyprint to 52.5
— Use screamshotter >= 2.0.9 by default
Security
— Bump django-debug-toolbar from 3.1.1 to 3.2.1

26.87 2.55.1 (2021-04-15)

Documentation
— Add outdoor section to user manual
Bug fixes
— Fix themes not including published touristic contents/events in API v2
— Fix duplicate Access-Control-Allow-Origin header in sensitive areas API endpoint
— Fix orientation/wind labels in outdoor course filter
— Hide « Add a brother site » link if no parent site
— Filter outdoor site/course orientations with a OR instead of a AND
— Reverse wind arrows
Maintenance
— Use upstream appy dependency

26.88 2.55.0 (2021-04-09)

New Feature
— Add /sensitivearea_species endpoint on api v2

26.89 2.54.0 (2021-04-09)

New Feature
— Add “trek” filter on endpoint /sensitivearea in api v2

26.90 2.53.1 (2021-04-07)

Bug fixes
— Fix geojson display in API V2 /trek/ endpoint
— Add publication filter by language on /trek/ detail view endpoint
— Fixed the fact that the detail view of /trek/ endpoint crash when a trek has more than one parent
— Do not display elements linked to content not published or not used at all in multiple endpoints on API V2

26.91 2.53.0 (2021-04-01)

New Feature
— Add departure_city attribute to treks and touristiccontents in API v2
— Allow to filter nomenclatures by portal in API v2
— Allow to retrieve a single unpublished trek if its parent is published in API v2
Maintenance
— Simplify code thanks to Python 3 (thanks to Claude Paroz)
— Add new sentry-sdk dependency
Bug fixes
— Avoid a db connection when requesting time from database (thanks to Claude Paroz)
Security
— Bump lxml from 4.6.2 to 4.6.3

26.92 2.52.0 (2021-03-25)

Bug fixes
- Allow to add an outdoor sub-site or a course in a site owned by another structure
- Fix outdoor site orientation/wind filtering
- Add missing outdoor module translations

New Feature
- Outdoor course itinerancy
- Add altimetry informations to outdoor sites and courses
- Add outdoor course fields height and equipment
- Add course layer to layers control
- Allow VAR_DIR setting from environment (thanks to Claude Paroz)
- Allow easier customization of loadpaths command (thanks to Claude Paroz)

Security
- Bump pillow from 7.1.2 to 8.1.1
- Bump jinja2 from 2.11.1 to 2.11.3

26.93 2.51.2 (2021-03-16)

Bug fixes
- Translate all text fields in API v2 trek endpoint
- Serve attachments for flatpages
- Fix bbox filtering of interventions

Performances
- Add prefetch to Path exports (CSV/Shapefile/GPX)

26.94 2.51.1 (2021-03-05)

Bug fixes
- Fix departure_geom attribute in API v2 (WGS84 projection, without Z)

26.95 2.51.0 (2021-03-02)

New features
- Add filtering by restricted area types
- Add outdoor course module
- Add a site/course tree view in outdoor site and course detail pages

Bug fixes
- Fix a backward compatibility to keep MAP_STYLES[“xxx”] config working in custom.py. However, we recommend to use new MAPENTITY_CONFIG[“MAP_STYLES”] for this.
- Use 2D lengths instead of 3D length for Geotrek-rando (to be consistent with Geotrek-mobile)
- Translate touristiccontent_category endpoint in API v2
- Fix crash of trek endpoing in API v2 when a geometry is a multilinestring (the previous fix was not working)

26.96 2.50.0 (2021-02-19)

BREAKING CHANGES
- Change URL of some API v2 enpoints. See Swagger online doc.
- API v2 thumbnails are now 400px large
- Split PDF urls by language in API v2

Bug fixes
- Fix API v2 crash when trek geom is a multilinestring
- Fix touristic content filtering in API v2 when both type1 and type2 are specified
- Synchronize pictogram for service types in mobile app

New features
- Cover image for static pages

Performances
- Automatically remove temporary topologies created before version 2.48.0.

Security
- Upgrade cryptography from 3.2 to 3.3.2

26.97 2.49.0 (2021-02-09)

BREAKING CHANGES
- MAP_STYLES setting should be now set in MAPENTITY_CONFIG["MAP_STYLES"]. A fallback exists to keep configuration from MAP_STYLES.
- The name of several filters in APIv2 are now in plural form. See swagger doc.

Bug fixes
- Fix cities filter in API v2 when id begins with a zero
- Fix cities and districts filter in API v2 when given id is nonexistant
- Allow to pass more than one id in most API v2 filters (see swagger doc)
- Allow to filter on several items in most list page filter

New features
- Add flatpage endpoint to API v2
- Add sector filter to outdoor site liste page
- Compute aggregated fields only from children, not parents for outdoor sites
- Practice, sector, wind and orientation filters on outdoor sites now take children into account

26.98 2.48.1 (2021-02-05)

Bug fixes
- Fix missing geometry in API v2 touristiccontent endpoint when using near_trek filter

26.99 2.48.0 (2021-02-03)

Performances
- Do not save temporary topologies in database. Sometimes they are not removed and accumulate

Refactoring
- Allow to use zoning app independently of others apps

Minor improvements
- Add id attribute to source and informationdesk APIv2 endpoints
- Add structure attribute to touristic contents/events in API v1

— Add publication, hierarchical level, practices and modification time columns to outdoor site list

26.100 2.47.2 (2021-01-28)

Bug fixes
— Fix crash in API v2 for touristics contents with alphanumeric external id

26.101 2.47.1 (2021-01-27)

Bug fixes
— Remove thumbnail and pictures attribute from API v2
— Replace them by the attachments attribute on Trek, POI and Touristic content
— The pdf attribute now returns an absolute URL
Performances
— Do not recreate geometry columns indexes at each upgrade

26.102 2.47.0 (2021-01-26)

New Feature
— Add cities and departure_geom fields to API v2 trek endpoint
— Add practice filter to API v2 trek endpoint
— Add touristiccontentcategory endpoint to API v2 (with types)
— Add many fields and filters to touristiccontent API v2 endpoint
Performances
— Optimize generation of the list of cities in list pages

26.103 2.46.0 (2021-01-25)

Database change
— « mnt » DEM table is now managed by django. It was renamed with altimetry_dem label. Data coming from mnt will be copied to new table.
Security fixes
— Enable XFrameOptionsMiddleware
— Hide nginx version
— Disable swagger (API v2 documentation) by default. To enable it, see swagger item in advanced documentation page.
— Fix XSS in filter popover
Bug fixes
— Fix impossibility to add paths on Ubuntu 20.04 (PostGIS 3)
— Fix doc that explains how to load fixtures
New Feature
— Allow to select API v2 fields for all endpoints
— Optimize development environment
— Add an order field on rating scales
— Allow multiple cardinal points for wind and orientation
— Add sectors for outdoor sites
— Add pictograms to outdoor practices and ratings

— Compute outdoor site sector, practice, orientation and wind from childs and parents

26.104 2.45.0 (2021-01-10)

HAPPY NEW YEAR !

Security fixes
— Upgrade cairosvg and lxml libraries
Bug fixes
— Fix migrations if some outdoor sites were created before
— Fix missing placeholders for orientations in filter
— Fix outdoor fixtures
— Fix doc to enable outdoor
— Fix path edition with PostGIS 3 (on Ubuntu 20.04)
— Allow site type to be blank
New Feature
— Add min/max ratings for outdoor sites
— Reorder outdoor site fields

26.105 2.44.0 (2020-12-18)

New Feature
— Add new fields to outdoor sites
— Allow geometrycollection for site geometry

26.106 2.43.1 (2020-12-10)

Bug fixes
— Remove a SQL debug

26.107 2.43.0 (2020-12-10)

BREAKING CHANGES
— Old attachments are now deleted by default in parser. Add *delete_attachment = False* attribute to your parsers if you want to keep old behaviour (unlikely).
Bug fixes
— Fix creation of interventions with their own topology
— Fix height of map on detail/create/update pages

26.108 2.42.0 (2020-12-04)

New Feature
— Minimal outdoor module (see documentation to enable)
Bug fixes
— Fix API v2 swagger

26.109 2.41.2 (2020-11-27)

Bug fixes
— Do not create point edges on zone borders (fix some crash when adding paths)
— Enable postgis_raster extension when creating a new DB in Ubuntu 20.04 package
WARNING !
— Geotrek Ubuntu repository changed to managed two versions (18.04 and 20.04) in parallel. If you already installed Geotrek Ubuntu package before you should run once "sudo apt-get update –allow-releaseinfo-change" to accept these changes.

26.110 2.41.1 (2020-11-25)

Bug fixes
— Fix publish ubuntu 20.04/18.04

26.111 2.41.0 (2020-11-25)

New Feature
— Allow to install geotrek on ubuntu 20.04 and 18.04
Maintenance
— Upgrade from Django 2.2 to Django 3.1
Minor Changes
— Names of file in shapefiles changed
Bug fixes
— Truncate attachment legend too long in AttachmentParserMixin

26.112 2.40.1 (2020-11-23)

Bug fixes
— Fix dive pictogram (fix PDF crash)
Minor Changes
— Remove language from user profile. Now you can switch language from menu.
— More API v2 improvements (trek endpoint, new API_IS_PUBLIC setting)
Doc improvements
— Update translation

26.113 2.40.0 (2020-11-18)

New Features
— Handle different file formats in loadpoi command (all formats supported by gdal)
— Improve API V2 filters and endpoints
Bug fixes
— Fix tooltip hidden on module bar (change layout mode to display flex)
Doc improvements
— Reorganize index
— Add sphinx container for dev mode
— Improve custom dist to give right templates of values in parameters

26.114 2.39.1 (2020-10-28)

Bug fixes
— Fix delete draft permission should allow use delete button

26.115 2.39.0 (2020-10-27)

New Features
— Modification of API V2 routes
— Add some filtering on Treks in API V2
Doc improvements
— Fix doc development command line
— Improving docs : advanced configuration / synchronisation

26.116 2.38.6 (2020-10-20)

Bug fixes
— Fix middleware interfaces without ipv4
Minor Changes
— Pictogram for trek's label is optional

26.117 2.38.5 (2020-10-20)

New Features
— Create new label for trekking, move inside_park to this label

26.118 2.38.4 (2020-10-16)

New Features
- Add relation between a Report and a Trek
- Change Report mail template to link the related Report in admin

Minor Changes
- Handle Z coordinates on GPX files
- Force size pictograms in admin

Doc improvements
- Add info about what's new in 2.33
- Change commands and so according to 2.33 [camillemonchicourt]
- Fix doc about spatial extent setting

26.119 2.38.3 (2020-10-05)

Bug fixes
- Fix diving levels display on lists
- Fix scrollable leaflet right control layer
- Fix lists in csv (#2286)

Doc improvements
- Add doc for translating
- Update synchronization with sync_rando options (Thanks JeanLenormand)

New Feature
- Show booklet pdf version on detail view

26.120 2.38.2 (2020-09-24)

Bug fixes
- Fix APIDAE parser when there is no element
- Fix booklet generation with pdfimpose

26.121 2.38.1 (2020-09-22)

Bug fixes
- Fix USE_BOOKLET_PDF setting

26.122 2.38.0 (2020-09-21)

New Feature
- Add facebook informations on target Portals
- Add description and title on target Portals
- Synchronize multiple meta informations with target portals.
- Add booklet pdfs with setting USE_BOOKLET_PDF

Bug fixes
- Fix stake deletion list
- Fix generation of stake automatically created with factories

Minor changes

— Fix use of screamshotter and convertit for development
— Use official postgis docker image
— Change of legend size on pdfs

Doc fixes

— Update suricate configuration doc
— Update anonymize report documentation

26.123 2.37.0 (2020-09-16)

Bug fixes

— Fix script install

New Feature

— Add second external id api v2 for treks

26.124 2.36.1 (2020-09-04)

Bug fixes

— Fix crash in json DEM generation if the topology is a point

26.125 2.36.0 (2020-09-01)

New Feature

— Allow to (un)publish some cities/district/areas on Geotrek-rando/mobile

26.126 2.35.1 (2020-08-24)

Bug fixes

— Really add an id field to each SQL view

26.127 2.35.0 (2020-08-21)

New Feature

— Allow for custom SQL to be run at install/upgrade

Bug fixes

— Add an id field to each SQL view to allow QGIS to open them

26.128 2.34.0 (2020-07-10)

New Feature
- Add reservation system/id fields to treks to allow itinerancy online booking
- Add category code (used in Geotrek-rando) to categories list in admin

Minor changes
- Add install scripts for Ubuntu packages

Bug fixes
- Fix icons display in categories list in admin

26.129 2.33.13 (2020-07-01)

New Feature
- Add fields to reports for Suricate support
- Add helper to send report to Suricate API on save, if setting *SURICATE_REPORT_ENABLED* is *True*

26.130 2.33.12 (2020-06-23)

Bug fixes
- Change doc flatpages-flatpages.jpg to png
- Fix line topologies create path
- Fix svg's fixtures wich cannot be tranform as png with cairosvg
- Fix duration's filter mobile
- Fix report email OSM coords

New Feature
- Synchro mobile get only used practice, themes, networks ...

26.131 2.33.11 (2020-06-05)

Bug fixes
- Fix long attachments name synchro

26.132 2.33.10 (2020-06-02)

Bug fixes
- Fix migration is_image 0011_attachment_add_is_image

New Feature
- Allow to clean attachments not used anymore (clean_attachments)

26.133 2.33.9 (2020-06-02)

Bug fixes
— Fix small treks profile

26.134 2.33.8 (2020-05-22)

Bug fixes
— Fix package install if geotrek user already exists
— Attachment download error breaks global import

26.135 2.33.7 (2020-05-18)

Bug fixes
— Show blades without line in signage detail page
— Fix information desks editing
— Fix trek and POI filtering

26.136 2.33.6 (2020-05-14)

Bug fixes
— Don't overwrite initial data in existing database on first install

26.137 2.33.5 (2020-05-13)

Bug fixes
— Add a scrollbar to signage and blade forms
— Fix city affectation for looping paths
— Fix attachment download with redirection
— Fix logout next page
— Fix blade/line creation crash
— Fix lines layout in blade detail page

Upgrade notes

If you installed version 2.33.3 before (no matter if you upgrade directly or from 2.33.4), you should get errors like « django.db.utils.ProgrammingError : column « deleted » of relation signage_blade ». To fix them, run `sudo geotrek migrate --fake signage 0016;`.

26.138 2.33.4 (2020-05-04)

Minor changes
— Improve blade CSV export
Bug fixes
— Fix ordering of blades
— Fix empty attachment link in admin list
— Fix some french translations
— Fix redirections when downloading attachments in parsers
— Fix migrations when DB contains a deleted blade
— Fix stdout flush in sync commands
Maintenance
— Upgrade from Django 2.0 to Django 2.2
— Fix deprecation warnings

26.139 2.33.3 (2020-04-28)

No changes. Just force a new build in CI

26.140 2.33.2 (2020-04-28)

No changes. Just force a new build in CI

26.141 2.33.1 (2020-04-28)

No changes. Just force a new build in CI

26.142 2.33.0 (2020-04-28)

BREAKING CHANGES
— New installation method (Ubuntu packaging)
— Alternative installation method (Docker, for experts only)
— Remove name field from feedback report, to be GDPR compliant
— Rename functions, triggers and sequences in database
Bug fixes
— Fix timeout when saving long treks (increase computation performances)
— Fix mecanism to put tables in postgresql schemas
— Better download errors handling in parsers
— Make sure signage and related blade have the same related structure
Maintenance
— Upgrade from Django 1.11 to Django 2.0
New features
— Allow to attach interventions to blades, paths, trails, treks, POIs and services in addition to infrastructures and signages
— Allow to merge dropdown list items in admin. Check them in list view and choose « Action : Merge »
— Add a django command to erase email from feedback reports after 365 days
Upgrade notes

— The installation method has been totally rewritten with an Ubuntu packaging (`apt install geotrek-admin`), only available for Ubuntu 18.04 actually.
— If you upgrade from Geotrek-admin <= 2.32, then apply the dedicated migration script. See https://geotrek.readthedocs.io/en/master/installation.html#upgrade-from-geotrek-admin-2-32.
— Geotrek-admin is now automatically installed in `/opt/geotrek-admin/` directory and the advanced configuration file moved to `/opt/geotrek-admin/var/conf/custom.py`. See advanced configuration documentation for details.
— The automatic NGINX configuration can be overriden in `/opt/geotrek-admin/var/conf/nginx.conf.in` file. See NGINX configuration documentation for details.

26.143 2.32.11 (2020-03-17)

Minor changes
— Add UML digrams of data model to documentation
— Remove URL in weblinks dropdown
— Move ambiance after description teaser
Bug fixes
— Fix a WeasyPrint warning
— Fix zoning filters on path

26.144 2.32.10 (2020-03-11)

Bug fixes
— Fix POI, touristic contents and touristic events sort in mobile v3 API
— Change Lambert93 signage coordinates format
— Fix TourInSoftparser with # inside <MoyenDeCom> values
— Show File and URL fields as required in attachement form
— Do not show Function field as required in Intervention form
— Do not show Amount and Organism fields as required in Project form

26.145 2.32.9 (2020-03-06)

Bug fixes
— Fix « upper bound of FOR loop cannot be null » crash in SQL triggers

26.146 2.32.8 (2020-03-05)

Minor changes
— Allow to choose Touristic content ordering in API
— Add external ID to projects and interventions
Bug fixes
— Fix the modification of the published field without the « Can publish… » permission

26.147 2.32.7 (2020-03-02)

BREAKING CHANGES
— Rename tables and fields in database
Minor changes
— Retry on HTTP 503 errors in parsers
Bug fixes
— Fix install on Xenial (again)
— Fix video embed url https
— Fix « Only LINESTRING and MULTILINESTRING are supported » crash in SQL triggers

26.148 2.32.6 (2020-02-28)

DO NOT USE IT !

26.149 2.32.5 (2020-02-18)

Bug fixes
— Fix filters sort in mobile v3 API

26.150 2.32.4 (2020-02-12)

Bug fixes
— Fix install on Xenial

26.151 2.32.3 (2020-01-27)

Bug fixes
— Fix review, publish do not display after resave a published or without permission to publish
— Fix attachment asterisks and crispy form
— Display only one time the same path when on trail detail

26.152 2.32.2 (2020-01-09)

Bug fixes
— Upgrade WeasyPrint

26.153 2.32.1 (2019-12-20)

Bug fixes
- Fix a crash in stake computation when adding an intervention
- Fix a crash in project list when one of them has no end year
- Fix drapping with no-data DEM values
- Fix nav pills to choose language in forms

26.154 2.32.0 (2019-12-13)

New features
- Add DISPLAY_SRID into settings to allow user to choose it's own format for GPS coordinates
- Make some fields optional (class Trail, Intervention, Project, OrdererdTrekChild, POI)
- Sort dropdown lists
- Document settings

26.155 2.31.0 (2019-12-06)

New features
- Sync mobile data from web UI
- The SHOW_LABELS setting allows to hide status labels on map

26.156 2.30.0 (2019-11-26)

Breaking changes
- Remove support of Ubuntu 14.04 Trusty

Maintenance
- Move from Python 2 to Python 3

Bug fixes
- Fix PDF generation for not published treks

26.157 2.29.15 (2019-11-12)

Bug fixes
- Fix install (use a version of venusian that is compatible with Python 2)

26.158 2.29.14 (2019-11-04)

Bug fixes
- Do not check structure for excluded POIs

26.159 2.29.13 (2019-10-30)

Minor changes
- Do not set structure by default when creating elements in dropdown lists.
- Trek duration is now optional
- Automatically disable empty filters in API for mobile v3
- Add support for Tourinsoft v3 in addition to v2
- Add more links form/to sensitive areas
- Add more unit tests

Bug fixes
- Fix SEO for static page titles
- Fix TouristicContentParser deletion having type1/2 with same values
- Fix serialization of MultiPolygon sensitive areas

26.160 2.29.12 (2019-10-23)

Minor changes
- Show completeness on dive detail page
- Add practice field to trek and dive completeness

Bug fixes
- Fix multiple sensitive areas on treks with settings SENSITIVE_AREA_INTERSECTION_MARGIN = 0
- Fix multiple sensitive areas on dives

26.161 2.29.11 (2019-10-17)

Bug fixes
- Fix filter still available after come back to list
- Add settings allowing to change permission on voluminous datas. Voluminous datas are not stocked at the same place

26.162 2.29.10 (2019-10-08)

Minor changes
- Do not set username as attachment author by default

Bug fixes
- Don't crash sync_rando with PIL.Image.DecompressionBombError
- Fix mode selection when adding/editing an attachment
- Fix authenticated parsers

26.163 2.29.9 (2019-10-02)

Bug fixes
— Fix sync_rando : sensitive area with multi polygons

26.164 2.29.8 (2019-09-26)

Minor Changes
— Increase path name field length
Bug fixes
— Fix csv_display signage with not ascii character

26.165 2.29.7 (2019-09-25)

Minor Changes
— Add pois services tourism on sync_rando
— Add endpoints api for diving
Bug fixes
— Fix is_public() call checking if the object is ppublic or not.
— Remove duplicate description detail diving

26.166 2.29.6 (2019-09-19)

Bug fixes
— Fix sync_rando command with diving

26.167 2.29.5 (2019-09-13)

Bug fixes
— Sync POIs related to dives
— Fix sync of manual PDF (again)

26.168 2.29.4 (2019-09-09)

Minor Changes
— Add reviews in dives module
Bug fixes
— Fix length should be length_2d in pdfs

26.169 2.29.3 (2019-08-28)

Minor Changes
— Allow to override nginx port in etc/settings.ini
Bug fixes
— Fix sync of manual PDF

26.170 2.29.2 (2019-08-28)

Minor Changes
— Add a command to import dives
Bug Fixes
— Fix crash when a dive is not a point

26.171 2.29.1 (2019-08-26)

Minor Changes
— Show treks related to dives
Bug fixes
— Fix retrieval of content-length of attachments with HTTPS
— Fix detection of hardcoded SRID in migrations
— Fix Est/West swap in diving module
— Fix version of more-itertools
— Fix missing difficulty and technical levels in dive detail page and PDF

26.172 2.29.0 (2019-08-20)

New features
— Diving module (optional, see manual to enable)
Minor Changes
— Improve mobile sync
— Do not automatically zoom over level 16
Bug fixes
— Fix black map screenshots (after a manual cache deletion)
— Fix related POI order with dynamic segmentation disabled

26.173 2.28.0 (2019-08-09)

New features
— Geotrek without dynamic segmentation is back
Minor Changes
— Add a settings allowing to remove certain items from the left menu
— Serve attachment with “Topoguide” type as public PDF
Bug fixes
— Fix missing pictograms for mobile app
— Translate feedback acknoledgment email
— Fix sync_mobile command for itinerancy

26.174 2.27.12 (2019-07-22)

Minor Changes
— Add itinerancy mobile

26.175 2.27.11 (2019-07-17)

Minor Changes
— Change condition's on_delete for SET_NULL
— Add the possibility to add Multipoint with one Point on commands loadinfrastructure/loadsignage
Bug fixes
— Fix choices fields, should only take in account existing (not deleted) elements
— Fix delete Organism
— Fix sensitivity parser with MultiPolygon
— Fix profile and languages

26.176 2.27.10 (2019-07-10)

Minor Changes
— Set OpenTopoMap as default map background
— Resize information desk type pictograms in mobile API
Bug fixes
— Fix delete intervention type

26.177 2.27.9 (2019-07-01)

Minor Changes
— Add ambiance field to trek detail endpoint in mobile API

26.178 2.27.8 (2019-06-28)

Minor Changes
— Add primary color setting for PDF
— Allow to override practices pictogram color in custom trek PDF template

26.179 2.27.7 (2019-06-26)

Bug fixes
— Fix public PDF overflow
— Resize category and POI pictograms for mobile app
— Convert pictograms from SVG to PNG for mobile app
— Fix structure (or not) related scrolldowns validation
— Remove unvisible paths in remove_duplicate_paths command
— Fix list of additional layers in layer selector
— Don't reset excluded POIs when saving treks

Minor Changes
— Allow to merge multiple comment columns when importing paths
— Add color field to touristic contents categories (for mobile app only)
— Handle invalid geometries when importing districts

26.180 2.27.6 (2019-06-04)

Bug fixes
— Fix mobile API

26.181 2.27.5 (2019-05-29)

Bug fixes
— Fix regulatory sensitive area parser
— Fix handling of parser errors

26.182 2.27.4 (2019-05-27)

Bug fixes
— Fix crash with –srid option of loadpaths command
— Add option portal in sync_mobile for the treks
— Fix encoding error on watermarks
— Fix bad references to sync_mobile in sync_rando command

26.183 2.27.3 (2019-05-23)

New features
— Allow to set order of filters in mobile API
— Add ascent and district filters to mobile API
Minor Changes
— Replace text by an id in url of pictures with watermarks
— Change default settings watermark
Documentation
— Add PDF overriding section

26.184 2.27.2 (2019-05-14)

Minor Changes
— Add points_reference by treks in api mobile
Bug fixes
— Remove public pdf poi
— Fix filter cities without paths

26.185 2.27.1 (2019-05-06)

Bug fixes
— Fix api mobile with only sensitivity app

26.186 2.27.0 (2019-05-02)

New features
— Add watermark on pictures
— Allow to change structure of an object with permission by_pass_structure
Bug fixes
— Fix a floating point computation problem in SQL trigger
— Fix trails in detail of intervention and opposite
— Fix color on restricted area

26.187 2.26.5 (2019-04-19)

Bug fixes
— Add slug to mobile API
— Fix crash with empty images

26.188 2.26.4 (2019-04-18)

Bug fixes
— Fix migration tourism 0004

26.189 2.26.3 (2019-04-12)

Bug fixes
— Fix parsers delete datas
Minor Changes
— Add command loaddistrict, loadcities, loadpaths

26.190 2.26.2 (2019-04-10)

Bug fixes
— Fix sync_rando command (BadZipfile exception)
— Fix nginx and Django conf when SSL is enabled
— Fix restricted area layers

26.191 2.26.1 (2019-04-03)

Bug fixes
— Fix blade form
— Fix sync_mobile, sync_rando with url https and http

26.192 2.26.0 (2019-04-01)

New features
— New API for mobile app v3
Bug fixes
— Fix signage type pictograms
— Some cosmetics on tourism detail pages (clickable links)
— Fix Tourinsoft opening period parsing (multiple periods)
— Fix Bad Status Line exception

26.193 2.25.3 (2019-03-26)

Bug fixes
— Fix Tourinsoft parsers one time again (practical info for events)

26.194 2.25.2 (2019-03-26)

Bug fixes
— Fix Tourinsoft parsers again (postal address)

26.195 2.25.1 (2019-03-25)

Bug fixes
— Fix Tourinsoft parsers

26.196 2.25.0 (2019-03-25)

New features / Performances
— Add the possibility to load layers (do not load them automatically)
Minor changes
— Add Touristic Content TourInSoft Parser
— Add tool testing ign keys without ggp3
Documentation
— How to update IGN urls

26.197 2.24.8 (2019-03-15)

Bug fixes
— Fix bug parsers filetype not related with structure

26.198 2.24.7 (2019-03-13)

Minor changes
— Add elevation on sensible areas
Bug fixes
— Fix retry sync_rando tiles when tiles does not exist (landez 2.4.1)

26.199 2.24.6 (2019-03-07)

Bug fixes
— When updating interventions, stake field is no more required
— Fix duplicates in year filters in intervention module
— Configurable blade code
— Allow letters in blade number
— Improve signage templates
— Add « On signage/infrastructure » filter on intervention list

26.200 2.24.5 (2019-03-06)

Performances
— Add index to date_update columns

26.201 2.24.4 (2019-03-01)

Bug fixes
— Fix get attachments with crop

26.202 2.24.3 (2019-02-28)

Bug fixes
— Fix get attachments using generic foreign and not url
— Fix merge path

26.203 2.24.2 (2019-02-26)

Bug fixes
— Fix attachments and history linked with signage and infrastructure

26.204 2.24.1 (2019-02-12)

Bug fixes
— Fix install.sh (pin cairocffi version)
— Fix routing on paths with sharp angles
— Fix loadrestrictedareas command
— Fix altimetry on straight portions of paths
— Various signage fixes
Performances
— Allow client side caching with systematic revalidation for Layer, JsonList and graph views
— Remove validation of history bar
— Don't bringToFront() every single feature on map
— Do not show bullets at path extremities anymore by default. Set SHOW_EXTREMITIES setting to True in custom.py enable them.
— Remove networks and trails columns in path list

26.205 2.24.0 (2019-01-28)

New features
— Bulk path deletion

26.206 2.23.0 (2019-01-24)

New features
— Signage blades management
— Add gpx and kml export for path detail view
— Allow to load local GPX/Geojson file in list views
Bug fixes
— Fix sensitive areas API v2
— Fix migrations if infrastructure app not is not installed

26.207 2.22.10 (2019-01-09)

Bug fixes
— Fix duplicated results in API v2 with sensitive area filters

26.208 2.22.9 (2019-01-09)

Minor changes
— Separate Infrastructure and Signage models
— Create parser touristic event for apidae
— Refactor ApidaeParser
Bug fixes
— Add italian translations that are visible on Geotrek-rando
— Fix permissions attachments paperclip
Performances
— Improve map's performances

26.209 2.22.8 (2019-01-03)

Minor changes
— Now, empty portal field means « all portals » instead of « no portal »

26.210 2.22.7 (2019-01-03)

Bug fixes
— Fix command loadinfrastructure

26.211 2.22.6 (2019-01-02)

Minor changes
— Index path draft field
— Add eid field to load_infrastructure command
— Add loadrestrictedarea command
— Install postgis package

26.212 2.22.5 (2018-12-19)

Bug fixes
— Fix DB migration

26.213 2.22.4 (2018-12-19)

Bug fixes
— Replace u2028 and u2029 by n in synced (geo)json files (fix Geotrek-mobile crash)
Minor changes
— Add EID field to all models and increase its length

26.214 2.22.3 (2018-12-14)

Bug fixes
— Don't publish deleted infrastructures/signages
— Add default pictograms to published infrastructures/signages

26.215 2.22.2 (2018-12-10)

Bug fixes
— Fix bugs with HTTPS access
— Fix for some modules to edit attributes and not the geometry
Minor changes
— add options to sync signages and infrastructures
— sync global signages and infrastructures

26.216 2.22.1 (2018-11-27)

Bug fixes
— Fix trekking form with pois_excluded
— Give the possibility to get type of infrastructures and signages without structure

26.217 2.22.0 (2018-11-27)

New features
— Allow to publish signage and infrastructure
— Allow to exclude POIs from a trek
— New access rights to edit draft path
— New access rights to edit attributes and not the geometry
— Allow to remove duplicate path in database : command remove_duplicate_paths
Bug fixes
— Fix snap on crossing point : take all paths easier
— Fix a clash between current url and SYNC_RANDO_OPTIONS
— Fix screamshotter when SSL is enabled

26.218 2.21.1 (2018-09-18)

Breaking changes
— Remove type1 from trek API when SPLIT_TREKS_CATEGORIES_BY_PRACTICE is not set
— Rename Trek category label to Hike in API. You can override this in geotrek/locale/<language code>/LC_MESSAGES/django.po files
Bug fixes
— Fix crash in log entries view
Minor changes
— Save column sort by module
— Rename SITRA to APIDAE

26.219 2.21.0 (2018-09-04)

New features
- Path deletion warning page now shows linked topologies
- Allow to add Dailymotion video attachments
- Add a command to unset structure in lists of choices and group choices with the same name

Bug fixes
- Fix Sync_rando View
- Fix loaddem
- Fix creation of Youtube/Soundcloud attachments
- Fix cancellation when editing geometries
- Show which structure choices are related to
- Add confort and stake filters to path list
- Fix sync of touristic contents for mobile app

26.220 2.20.1 (2018-07-16)

Bug fixes
- Fix Completed Filter in Touristic Event

26.221 2.20.0 (2018-06-27)

New features
- Allow to share glossaries between structure (just keep structure field empty)
- Allow to import infrastructures, not only signages
- Allow to split treks by itinerancy
- Path deletion does not delete the related point topologies anymore. Instead point topologies are linked to the nearest remaining path.
- Keep inode and mtime when synced file don't change

Minor changes
- Optional img-responsive class on flatpage images

Bug fixes
- Fix virtualenv install
- Upgrade celery to 4.1.1
- Fix the bug which remove a path when we merge 2 paths with a 3rd path on the point of merge. (ref #1747)

26.222 2.19.1 (2018-05-28)

Bug fixes
- Update APIDAE API URL
- Fix filename encoding errors in import

26.223 2.19.0 (2018-05-22)

Minor changes
— Allow to specify encoding when importing data

26.224 2.18.5 (2018-05-07)

Bug fixes
— Fix related structure when creating new objects
Minor changes
— Show related structure on all detail pages

26.225 2.18.4 (2018-05-02)

Bug fixes
— Fix sync of deleted sensitive areas
— Fix touristiccontents.geojson crash when reservation_system is None
Minor changes
— Add Ubuntu 18.04 Bionic Beaver support

26.226 2.18.3 (2018-04-27)

Bug fixes
— Fix imports when enabling only sensitivity app

26.227 2.18.2 (2018-04-27)

Preventive maintainance
— Upgrade to Django 1.11
— Upgrade several python dependencies
Minor changes
— Allow user with can_bypass_structure permission to set or update the related structure on sensitive areas
Bug fixes
— Put back filter widgets in two columns (#1663)
— Do not include (possibly forged) structure field in post requests
— Fix geojson format in sensitive areas API
— Fix filtering by practices in sensitive areas API
— Fix sync_rando when sensitivity app is not enabled
— Adapt BiodivParser to API modifications
— Order sensitive areas by decreasing area in API to be able to clic on each of them
— Set ownership in parsers depending on logged user
— Pagination requires ordering in v2 API

26.228 2.18.1 (2018-03-26)

Bug fixes
— Fix distribution of tables in schemas
— Allow to reset type1/type2 in TouristicContentSitraParser
Minor changes
— Do not truncate install.log

26.229 2.18.0 (2018-03-22)

Breaking changes
— Drop support of Ubuntu Precise 12.04, please upgrade to Trusty 14.04 or Xenial 16.04 before upgrading Geotrek-admin
Preventive maintainance
— Upgrade to Django 1.9
— Upgrade several python dependencies
Bug fixes
— Fix path duplication in path split trigger
Minor changes
— Show total path length in path list

26.230 2.17.3 (2018-03-23)

** Bug fixes **
— Fix install

26.231 2.17.2 (2018-02-07)

Minor changes
— Use id rather than french name for practices and structure in sensitive areas API
— Add permission to import sensitive areas

26.232 2.17.1 (2018-02-02)

Minor changes
— Implantation year on signages and infrastructures is now a filter with current existing values
— Trek form field “practice” is moved to first panel
— Add sensitive areas to public trek PDF
— Do not show paths on PDF
— Add SENSITIVE_AREA_INTERSECTION_MARGIN setting
Bug fixes
— Fix snapping
— Fix import of sensitive areas when languages lists do not match
— Fix trail links in project detail view
— Add “approved” field in touristic content and event exports
— Fix service type with specials character in trek detail
— Fix bbox filtering in sensitive areas API

— Add CORS header to sensitive areas API
— Filter on spatial extent when importing from Biodiv'sport
— Fix celery task runner version

26.233 2.17.0 (2018-01-15)

New features
— Sensitive Areas

26.234 2.16.1 (2018-01-10)

Bug fixes
— Fix encoding errors while generating static versions of rando pages
— SEO for static versions of rando pages
— Disable template caching (fix ODT generation)

26.235 2.16.0 (2017-12-21)

New features
— Create static versions of rando pages with opengraph data (Facebook)
— Add external id field to POI form
Bug fixes
— Fix download of python packages for pypi.python.org (SSL is now required)
— API v2 : Fix full URL pictures in nested serializers
— Fix network SVG (add viewbox) to make them visible in Geotrek-rando
— Hide file parsers form if no one is available

26.236 2.15.2 (2017-09-28)

Bug fixes
— Fix existing path split in particular cases where postgis doesn't see real intersections.
— Fix project and intervention detail template.
— Fix synchronization of POI media other than images
— Include pois, parking and reference points to compute PDF map zoom

26.237 2.15.1 (2017-08-23)

New features
— Add es translation for PDF
— Add mailssl setting
Bug fixes
— Fix APIDAE import illustrations
— Fix double import parsers
— Fix cirkwi export
— Select only published POIs in GPX and KML files
— Remove deprecated experimental setting

— Fix HTML tags & entities in feedback email

26.238 2.15.0 (2017-07-13)

New features
— API v2 Beta 1. Optimized multilingual filtered endpoints for paths, treks, tours and pois.
— See HTML doc and examples in /api/v2/. Authentication with Basic HTTP (https://en.wikipedia.org/wiki/Basic_access_authentication).
— Don't use it in production without HTTPS
Bug fixes
— Fix pdf default public templates (weasyprint)
— Fix screamshotter standalone install (map screenshots)

26.239 2.14.3 (2017-07-03)

Bug fixes
— Cirkwi export fixes and improvements

26.240 2.14.2 (2017-06-21)

Bug fixes
— Fix attachments edition

26.241 2.14.1 (2017-06-22)

Bug fixes
— Refactor signals pre / post migrate according Django 1.8
— Update translations
— Fix path splitting
— Fix AutoLogin Middleware with mapentity 3.1.4

26.242 2.14.0

WARNING !
— Upgrade to version 2.14.0 is only possible from version 2.13.0
New features
— Upgrade to Django 1.8. This is a big step, migrations are reset, please backup before upgrade.
— Ability to skip attachment download in parsers and use external links.
Minor changes
— Possibility to exclude pois in cirkwi xml export by adding ?withoutpois=1 to url (http://XXXXX/api/cirkwi/circuits.xml?withoutpois=1
— Add MOBILE_TILES_EXTENSION setting (for compatibility with old mobile apps, set it to "png")
— API optimization
— Disable auto size for service icon in trek description.
Bug fixes
— Fix topologies and cities intersections

26.243 2.13.0 (2017-03-02)

Minor changes
— MOBILE_TILES_URL settings is now a list which can be used to merge different layers in mobile application

26.244 2.12.0 (2017-02-16)

New features
— add loadsignage command
Minor changes
— add field implantation_year to model BaseInfrastructure
— add field owner to model LandEdge
— add field agreement to model LandEdge

26.245 2.11.5 (2017-02-06)

Bug fixes
— Fix topologies and cities intersections

26.246 2.11.4 (2017-02-01)

Bug fixes
— Remove deprecated datasource (replaced by import parsers)
— Stop install.sh if make update or wget fails
— Create database with right owner if user exists but database does not
— Make sure supervisor service is started after install
— Fix HTML entities in feedback email
— Fix cirkwi export for treks with multilinestring geom
Minor changes
— Add filter usages on paths
— Add filters name and description on infrastructures and signages
— Add picture to PDF for feedback reports (only in Weasyprint mode)

26.247 2.11.3 (2016-11-15)

Bug fixes
— Upgrade mapentity (fix map centering in PDF exports)
— Fix cirkwi export when trek geom is not a linestring

26.248 2.11.2 (2016-09-15)

Bug fixes
- Do not synchronize not-published treks with published but deleted parents
- Allow to specify portal in touristic content parsers
- Fix import of type1 in HebergementsSitraParser
- Fix source and portal missing in shapefile exports

Performances
- Improve performances of DEM computation for huge treks

26.249 2.11.1 (2016-08-17)

Minor changes
- Fix slug URL for « oe » ligature
- Improve zoom of map captures in PDF

26.250 2.11.0 (2016-08-02)

Bug fixes
- Fix weasyprint install
- Fix label displayed twice with Sitra Parser

Minor changes
- Update translations
- Update import documentation
- Record source is no nore structure related

New features
- ability to filter synchronized content with different portals

26.251 2.10.4 (2016-05-19)

Breaking changes
- Deprecate MAPENTITY_WEASYPRINT setting. Now public PDF use Weasyprint HTML templates and private PDF use legacy ODT template.

Minor changes
- Improve altitude profile computation (increase smoothing)
- Improve HTML templates for public exports
- Improve SITRA parser
- Allow to use source variable in PDF templates

Bug fixes
- Fix comparison of zip files to keep mtime when nothing changed
- Upgrade simplekml lib (should fix KML exports)

26.252 2.10.3 (2016-05-11)

Minor changes
— Update default pictograms for touristic content categories
— Update default pictograms for themes
Bug fixes
— Workaround a bun in supervisor init script
— Fix multilinestring instead of linestring in trek shapefile parser

26.253 2.10.2 (2016-04-12)

Minor changes
— Add source filter to touristic contents/events
— Allow installation as root (not recommended, use with caution)
Bug fixes
— Restore contents deleted and then created again in EspritParcParser
— Add a warning if type1/type2 is not created in EspritParcParser
— Replace input by textarea in flatpage form

26.254 2.10.1 (2016-03-17)

Bug fixes
— Allow access to information desks in API (and so to map capture and PDF) for unpublished treks
Minor changes
— Parsers improvements

26.255 2.10.0 (2016-03-03)

New features
— Add support for Ubuntu 15.04 Vivid
Breaking changes
— Remove TileCache service (you should set up tiles source with LEAFLET_CONFIG variable in *geotrek/settings/custom.py* now)
— Run supervisor as root (you should now run *sudo supervisorctl* instead of *./bin/supervisor*)
— Move nginx and supervisor logs to system dir */var/log/*
Minor changes
— Update default pictograms for difficulty levels
Bug fixes
— Fix sync_rando after deleting a trek with children

26.256 2.9.3 (2016-02-25)

Bug fixes
— Fix line break at start of contact in EspritParcParser
Minor changes
— Add parameters.json and themes.json files to API

26.257 2.9.2 (2016-02-17)

Minor changes
— Increase web link size
Bug fixes
— Fix path split
— Fix attachment parsing with same document type for several structures

26.258 2.9.1 (2016-02-10)

Bug fixes
— Don't forget to sync touristic contents/events media when skipping PDF
— Don't delete attachments of other objects when importing
— Don't delete other objects when constant fields are set in parsers

26.259 2.9.0 (2016-02-04)

New features
— Add parser for brand « Esprit Parc National »
Bug fixes
— Set user structure as related structure for all new objects

26.260 2.8.1 (2016-01-29)

Bug fixes
— Synchronize information desk thumbnails

26.261 2.8.0 (2016-01-28)

New features
— Use POI pictures in PDF if the trek has no picture itself
— Use a placeholder in PDF if there is no picture
— Parser to import touristic contents from SITRA
— Add list of all information desks to API
Bug fixes
— Allow NULL values for id_externe fields in database
— Fix missing elements (eg. POI enumeration) on trek map capture
— Prevent overlaping controls at bottom of list view

— Translation of column names in shapefiles export
— UTF-8 and truncated alerts in shapefile export

26.262 2.7.2 (2016-01-26)

Bug fixes
— Synchronize touristic events with no end date
— Fix PDF synchronization (eg. missing list of POI)

26.263 2.7.1 (2016-01-18)

Bug fixes
— Fix random z-index on forced layer polygon
— Fix pretty duration

26.264 2.7.0 (2016-01-14)

New features
— New button to add Youtube videos in flat pages
Bug fixes
— Fix iframe inclusion in flatpages.
— Fix double column buttons in gridmanager.
— Fix validation on flatpages for combo external_url + content.
— Fix responsive layout for provided templates in flatpages.
— Fix event link to closest visible path only
— Do not log anymore an error when submitting a form with an empty geometry

26.265 2.6.0 (2015-12-30)

New features
— Customization of practices ordering
Bug fixes
— Synchronize record source pictograms
— Add buttons to attachment update form
— Fix timestamps in database when connection with timezone other than UTC

26.266 2.5.2 (2015-12-29)

Bug fixes
— Fix hyphenation language in public PDF templates
— Add parents to trek public PDF template
— Fix numbering style in trek public PDF template
— Show points of reference over other features on trek detail map

26.267 2.5.1 (2015-12-18)

Bug fixes
— Trek public PDF fixes (size of service pictos, style of numbered lists, stages)

26.268 2.5.0 (2015-12-08)

New features
— Order has been added to flatpages which is reflected in the export for geotrek-rando frontend.
— Added 2 templates buttons for flatpages creating two layouts
— Option to add pois pictures to trek ones in Geotrek-Rando
Bug fixes
— Generate tiles zip files for all children of published treks
— Fix URL of video/audio media in API
— Fix default filtering of past touristic events in UI

26.269 2.4.4 (2015-12-02)

Bug fixes
— Show pending import/sync tasks

26.270 2.4.3 (2015-11-27)

Bug fixes
— Fix filtering by source in sync_rando for flatpages and tiles too

26.271 2.4.2 (2015-11-26)

Bug fixes
— Fix permissions of sync rando output directory
— Fix filtering by source in sync_rando

26.272 2.4.1 (2015-11-25)

Bug fixes
— Condition field of infrastructures is no more required
— Fix zipfile detection at import.
— Fix error handling at import (raise exception to browser).

26.273 2.4.0 (2015-11-18)

New features
- Paths can be merged
- Add trek parents to API
- Allow to sync public web site from web interface
- Add begin and end dates to touristic events list
- Filter conmpleted touristic events by default

Bug fixes
- Prevent concurrent imports and/or synchronization
- Fix rendering of HTML markup in weasyprint templates
- Fix missing publication field in some cases

26.274 2.3.0 (2015-11-09)

New features
- Sync rando now synchronizes touristic contents and events.
- Sync rando now exports only future events based on current date.
- Sync rando now synchronizes touristic content categories.

Bug fixes
- Added a custom validation to accept url only contribution in flatpages without content.
- Sync rando now handles crashes when it calls django views.

26.275 2.2.0 (2015-10-09)

New features
- Added normalisation for altimetry's json export
- Clarify 2D/3D lengths (fixes #1400)

Bug fixes
- Change plural on accessibility label for admin filter

26.276 2.1.0 (2015-09-29)

Breaking changes
- Instead of storing the parent of a trek, Geotrek now stores the children of a trek. This allows to use the same trek in several parents and to order them. WARNING ! Existing parent/child relation are lost. You will have to set them again after upgrade. Fixes #1479

New features
- Add trek infos (aka services for now)
- Add email sent to reporting user after submit
- Handle multiple reservation systems (fixes #1488)
- Add an option to sync_rando to filter by source (fixes #1480)
- Add add condition field to infrastructure table (fixes #1494)
- New Geotrek logo

Bug fixes
- Reload supervisor configuration after Geotrek upgrade
- Fix projection of waypoints in GPX exports

— Prevent unnecessary save for geom fields if they are not updated. This prevents triggering geom recalculation in postgres.
— Fix crash in case of missing or invalid picture
— Fix feedback API
— Unzip eggs to fix templates not found error
— Various parsers (import system) fixes and improvements

Documentation

— Document server migration

26.277 2.0.0 (2015-07-20)

Breaking changes

— Rework API URL schemas

New features

— Static API to disconnect Geotrek-rando from Geotrek-Admin (fixes #1428)
— Build zip files for mobile application
— Trek / Touristic content association distance depending on trek practice
— Option to hide published treks nearby topologies
— Add previous/next treks and category slugs to geojson API
— Add external id in trekking/tourism detail pages and exports
— Zip touristic contents as POI for mobile app v1
— Add external id field on Path
— Order intersections in Geotrek light mode
— Add reservation id field for touristic contents
— Integration of WeasyPrint to generate PDF from HTML/CSS instead of ODT

Bug fixes

— Remove HTTP calls to SoundCloud API at serialization
— Allow DEM to partially cover spatial extent

26.278 0.35.1 (2015-07-17)

Bug fixes

— Fix installation on ubuntu 12.04 with recent updates

26.279 0.35.0 (2015-07-10)

New features

— Add an import framework

Bug fixes

— Fix a crash in appy pod (PDF generation)
— Fix login with restricted access to some contents
— Fix buildout bootstrap arguments

26.280 0.34.0 (2015-05-20)

New features
- Itinerancy (parent/children treks)
- Allow to choose ordering of categories for Geotrek-Rando
- Bootstrap grid editor for flatpages
- Approved touristic contents and events
- Option to split trek category by practice or accessibility

Bug fixes
- Fix duration notation
- Flatten altimetry profiles

Bug fixes
- Show accessibility in trek detail page (fixes #1399)

26.281 0.33.4 (2015-04-07)

Bug fixes
- Ensure trek duration is a positive number
- Fix cirkwi exports (second try)
- Fix public PDF templates

26.282 0.33.3 (2015-04-01)

Bug fixes
- Fix systematic crash in PDF conversions

26.283 0.33.2 (2015-04-01)

Bug fixes
- Remove italian from fixtures
- Fix crash when generating two PDF in parallel

26.284 0.33.1 (2015-03-25)

Bug fixes
- Fix flat pages crash
- N to N source field (rel #1354)

26.285 0.33.0 (2015-03-25)

Breaking changes
- A new permission « Can publish … » is required to publish treks, pois, touristic contents and touristic events. Grant it to your users and groups if need be
- DB table `l_b_source` is renamed as `l_b_source_troncon`

New features
- Publication workflow (fixes #1018)
- Allow to add links to Youtube or Soundcloud media as attachment
- Make pictograms optional in some places when not required by Geotrek-Rando
- Add source for treks, touristic contents and touristic events (fixes #1354)
- Add external id field for treks, pois, touristic contents and touristic events
- Group cirkwi matchings in admin site (fixes #1402)

Bug fixes
- Fix projection of OSM link in feedback email
- Fix language in cirkwi exports

26.286 0.32.2 (2015-03-06)

Bug fixes
- Home now redirects to treks list in light version (without topologies)
- Fix Cirkwi export in light version
- Fix SRID in database migrations
- Add signage type filter again (fixes #1352)
- Add missing date filters to touristic events list

26.287 0.32.1 (2015-03-04)

Bug fixes
- Fix creation of a loop topology with two paths (fixes #1026)

26.288 0.32.0 (2015-03-04)

New features
- Export to cirkwi/espace loisirs IGN. After upgrade, run `bin/django loaddata cirkwi` to load data cirkwi tags and categories
- Wysiwyg editor for static web pages

Bug fixes
- Hide not published static pages in public REST API

26.289 0.31.0 (2015-03-02)

New features
- Add support of Ubuntu 14.04 to installer
- Public PDF for touristic contents/events (fixes #1206)
- Add treks close to other treks in REST API
- Add pictograms for trek accessibilities, touristic content types and touristic event types

Bug fixes
- Show edit button when having bypass structure permission
- Export missing fields in list exports (fixes #1167)
- Fix formating of float and boolean values in list exports (fixes #1366, #1380)
- Fix french translation
- Allow anonymous access to altimetry API for public objects
- Hide not published and deleted items in public REST API

26.290 0.30.0 (2015-02-19)

Breaking changes
- Trek practice (formerly usage) is no single valued so if a trek has multiple usages only one will be kept after upgrade. Others will be **lost** !
- After upgrade, run `make load_data` to load fixtures for accessibilities or create them by hand. You should clean-up the list of practices by hand.
- Don't forget to set up permissions to administrate practices and accessibilities.

New features
- Split trek usage field into practice and accessibility
- Treks and POIs are now structure related
- Allow anonymous access to media related to published items
- Check model read permission to give access to media
- Add a settings to set up CORS (cross-origin resource sharing)
- Allow to get POIs for a specific trek in REST API
- Consistent REST API (type1, type2, category for treks, touristic contents and touristic events)

Bug fixes
- Ensure path snapping is done on the closest point and is idempotent
- Fix language of PNG elevation charts
- Fix logo on login page
- Fix logs rotation
- Fix permissions creation

26.291 0.29.0 (2015-02-04)

New features
- GeoJSON API with all properties for Trek and Tourism

Bug fixes
- Fix permissions required to sync static Web pages
- Fix geom computation on line topologies with offset

26.292 0.28.8 (2014-12-22)

Bug fixes
— Fix altimetry sampling for segment with 0 length (rel #1337)

26.293 0.28.7 (2014-12-22)

Bug fixes
— Fix altimetry trigger when TREKKING_TOPOLOGY_ENABLED is set to False

26.294 0.28.6 (2014-12-18)

Bug fixes
— Fix 3D length shorter than 2D length (run sql command `UPDATE l_t_troncon SET geom=geom;` after upgrade to update altimetry informations of existing geometries)
— Fix translation of « Information desks » in public trek PDF
— Fix prepare_map_images and prepare_elevation_charts commands failing for deleted objects and for objects without geom

26.295 0.28.5 (2014-12-09)

Bug fixes
— Fix DEM optimizations when minimum elevation is zero (fixes #1291)
— Fix regression for translations of tourism (fixes #1315)
— Fix duplicate entries with year filter (fixes #1324)
Documentation
— French user manual first step about general interface
New features
— Set PostgreSQL search_path at user level (fixes #1311)
— Show 3D and 2D length in detail pages (fixes #1101)
— Show length and elevation infos in trail and all statuts detail pages (fixes #1222)
— Show trail length in list and exports (fixes #1282)
— Replace stake by length in path list (fixes #956, fixes #1281)
— Add subcontracting in intervention filter (fixes #1144)
— Add missing fields in project filter (fixes #219, fixes #910)
— Show status in interventions table among detail pages (fixes #1193)
— Add missing field in projects exports (ref #1167)
— Add length column to land module lists
— Number of workers and request timeout can be now configured in `settings.ini`
— Various improvements on trek public template, by Camille Monchicourt

26.296 0.28.4 (2014-11-21)

Bug fixes
- Fix mouse position indicator on `/tools/extents/` page when map tiles have Google projection
- Fix missing filters in trails list (fixes #1297)
- Fix infrastructure main type filter (fixes #1096)
- Fix flatpage creation without external url in adminsite
- Fix path detail page where deleted objects were shown (fixes #1302)
- Fix position of POIs on trek detail maps (fixes #1209)
- Fix TinyMCE not preserving colors (fixes #1170)
- Raise validation error instead of crashing when submitted topology is empty (fixes #1272)

Documentation
- Fix mention of MAP_STYLES (ref #1226)

Changes in experimental features
- Renamed *usage* to *type* in touristic events (fixes #1289)

26.297 0.28.3 (2014-11-12)

Bug fixes
- Fix upload form author/legend format (fixes #1293)
- Fixes history list (ref #1276)
- Prevent email to be sent twice on conversion error. Use info instead.
- Fix paperclip translations missing (fixes #1294)
- Fix filetypes not being filtered by structure (fixes #1292)
- Fix apparence of multiple-choices in forms (fixes #1295)

26.298 0.28.2 (2014-11-05)

Bug fixes
- Fix upgrade of django-leaflet to 0.15.0 (overlay layers)
- Fix apparence of overlay layers for tourism when experimental features are disabled
- Fix plural in tourism translation
- Fix unit tests
- Run this command to set the default information desk type with the original pictogram (or select a pictogram from the adminsite)

```
cd Geotrek-0.28.2/

curl "https://raw.githubusercontent.com/makinacorpus/Geotrek/v0.27.2/geotrek/trekking/
↪static/trekking/information_desk.svg" > var/media/upload/desktype-info.svg
```

26.299 0.28.1 (2014-11-05)

Bug fixes

— Fix deployment when tourism is not enabled
— Fix default duration when invalid value is filled (fixes #1279)
— Fix year filters for intervention, infrastructure and project (fixes #1287)
— Fix list filters not being restored (fixes #1236)

26.300 0.28.0 (2014-11-04)

Breaking changes

— Before running install, run this SQL command to add a column for file attachments :

```
ALTER TABLE fl_t_fichier ADD COLUMN marque boolean DEFAULT false;
```

New features

— Information desks now have a type (*Maison du parc*, *Tourist office*, ...) with the ability to set dedicated pictograms (fixes #1192).
— Ability to control which picture will be used in trek, using clicks on stars in attachments list (fixes #1117)
— Ability to edit attachments from detail pages directly (fixes #177, the 5th oldest issue !)
— Add missing columns in intervention exports (fixes #1167)
— Add ability (for super-admin) to add/change/delete zoning objects in Adminsite (ref #1246)
— Add ability to have paths records in database that will not appear in Geotrek lists and maps. Just set column `visible` to false in `l_t_troncon` table.
— Add ability to add external overlay tile layers (fixes #1203)

Bug fixes

— Fix position of attachment upload form on small screens
— Clearer action message in object history table
— Prevent image ratio warning from disappearing (fixes #1225)
— Touristic contents
— Touristic events

Internal changes

— Upgraded Chosen library for dropdown form fields
— Set `valide` column default value to false on paths table `l_t_troncon` (fixes #1217)
— All information desks are now available in GeoJSON (*will be useful to show them all at once on Geotrek-rando*).
— All tables and functions are now stored in different schemas. It allows to distinguish Geotrek objects from *postgreSQL* and *PostGIS*, and to grant user privileges by schema. It is also easier to browse objects in *pgAdmin* and *QGis*.
Caution : if you created additional database users, you may have to change their `search_path` and/or their `USAGE` privilege.

Experimental features

— We introduced models for touristic contents and events. In order to load example values for categories and types, run the following commands :

```
bin/django loaddata geotrek/tourism/fixtures/basic.json
cp geotrek/tourism/fixtures/upload/* var/media/upload/
```

— We introduced models for static pages, allowing edition of public static Web pages from Geotrek adminsite.

In order to enable those features under construction, add `experimental = True` in `etc/settings.ini`. Note that none of them are used in *Geotrek-rando* yet.

notes

Give related permissions to the managers group in order to allow edition (`add_flatpage`, `change_flatpage`, `delete_flatpage`, `add_touristiccontent` ...).

26.301 0.27.2 (2010-10-14)

Bug fixes

— Fix elevation info not being computed when intervention is created (ref #1221)
— Fix list of values for infrastructure and signage types (fixes #1223)
— Signages can now be lines if setting SIGNAGE_LINE_ENABLED is True (fixes #1141)
— Fix HTML tags in PDF exports (fixes #1235)
— Fix regression with Geotrek light

26.302 0.27.1 (2010-10-13)

Bug fixes

— Fix problems in forms, prevent Javascript errors

26.303 0.27.0 (2010-10-09)

Breaking changes

— Attribute for single information desk was removed (was used in **Geotrek-rando** < 1.29)
— Renamed setting `TREK_PUBLISHED_BY_LANG` to `PUBLISHED_BY_LANG`
— Renamed setting `TREK_EXPORT_MAP_IMAGE_SIZE` to `EXPORT_MAP_IMAGE_SIZE`, `TREK_EXPORT_HEADER_IMAGE_SIZE` to `EXPORT_HEADER_IMAGE_SIZE` and `TREK_COMPLETENESS_FIELDS` to `COMPLETENESS_FIELDS`. They are now a dictionnary by object type (see example)

New features

— POI publication is now controlled like treks
— POI now have a public PDF too
— Introduced `VIEWPORT_MARGIN` setting to control list page viewport margin around spatial extent from `settings.ini` (default : 0.1 degree)

notes

After upgrading, mark all POIs as published in the languages of your choice

```
UPDATE o_t_poi SET public_fr = TRUE;
UPDATE o_t_poi SET date_publication = now();
```

Bug fixes

— Add missing credit for main picture in trek PDF (fixes #1178)
— Paths module is now removed from user interface in *Geotrek-light* mode. (i.e. with `TREKKING_TOPOLOGY_ENABLED = False`)
— Make sure text fields are cleared (fixes #1207)
— Intervention subcontracting was missing in detail pages (fixes #1201)
— Make sure TLS is disabled when `mailtls` is False in settings
— Fix list of POIs in path detail pages (fixes #1213)
— Fix highlight from map for project list page (fixes #1180)

Internal changes

— Extracted the trek publication to a generic and reusable notion

— Complete refactor of Trek JSON API, now taking advantage of Django REST framework instead of custom code
— Added read/write REST API on all entities
— Refactored URLs declaration for altimetry and publishable entities
— Change editable status of topology paths in Django forms, since it was posing problems with Django-rest-framework
— Add elevation profile SVG URL in trek detail JSON (fixes #1205)
— Simplified upgrade commands for `etc/` and `var/`, and mention advanced configuration file

26.304 0.26.3 (2014-09-15)

Bug fixes
— Fix pretty trek duration when duration is between 24 and 48H (fixes #1188)
— Invalidate projet maps captures when interventions change, and treks maps when POIs change (fixes #1181)

26.305 0.26.2 (2014-08-22)

Bug fixes
— Fix search among attached files in Adminsite (fixes #1172)

26.306 0.26.1 (2014-08-21)

Bug fixes
— Upgrade *django-mapentity* for bug fix in ODT export and list of values in detail pages

26.307 0.26.0 (2014-08-21)

New features
— Interventions in project detail page is now shown as a simple table (ref #214)
— A generic system for interaction between objects attributes and details map was developped. It works with project interactions, topologies paths, etc. (ref #214)
— Show enumeration of interventions in project PDF exports (fixes #960)
— Number of POIs in now limited to 14 items in trek export (ref #1120)
— Number of information desks in now limited to 2 items in trek export (ref #1120). See settings `TREK_EXPORT_INFORMATION_DESK_LIST_LIMIT` and `TREK_EXPORT_POI_LIST_LIMIT`
— Justify texts of POIs in trek export, now converted to plain text.
— Trek export geometries are now translucid red by default (see `MAP_STYLES` setting) (ref #1120)
— Paths apparence in trek exports are now controlled by MAP_STYLES setting too.
— Images attachments are now resized to 800x800 for publication (instead of 500x500)
— Clarify intervention cost by function and mandays (fixes #1169)

Bug fixes
— Fix paths layer not being shown in detail pages (fixes #1161)
— Fix position of point topologies when closest path is not perpendicular (fixes #1156)
— Prevent parking to be cropped on map exports (fixes #1006)

Upgrades notes

Since the map export have changed, empty the cache :

```
rm -rf var/media/maps/*
```

26.308 0.25.2 (2014-08-14)

Bug fixes

— Fix translation of Job in intervention form (fixes #1090)
— Fix form error when no geometry is provided (fixes #1082)
— Show attachments in adminsite (fixes #1162)
— Fix JSON formatting of object attachment lists in API

26.309 0.25.1 (2014-08-01)

Bug fixes

— Fix Geotrek CSS not being deployed properly
— Fix trek relationships causing errors for PDF export

26.310 0.25.0 (2014-08-01)

New features

— Added projection file EPSG :32622 (fixes #1150)
— Now log addition and suppression of attachments in history
— Added notion of points of reference for treks (fixes #1105). (Can be disabled with `TREK_POINTS_OF_REFERENCE_ENABLED = False`)
— Edit the parking location directly on the trek map (ref #387)
— Show enumeration of POIs in trek PDF exports (fixes #871)

BUG fixes

— Fix permission check to see attachments (fixes #1147, ref #1146)
— Fix grouping of interventions in detail pages (fixes #1145)
— Fix project total intervention cost (fixes #958)
— Fix history entries not being saved when using formsets (fixes #1139)
— Fix postal code being saved as integer (fixes #1138). Existing records will have a leading zero when shorter than 5 charaters.
— Fix bug when form of intervention on infrastracture is not valid
— Limit height of layer switcher on small screens (fixes #1136)
— Get rid of next parameter when redirecting to login when permission missing (fixes #1142)
— Fix apparence of main menu when permissions are missing to view logbook and admin (ref #1142)

Internal changes

— Rework display of lists in detail pages, better factorization
— Removed links in logbook list for certain models
— Display messages in login page too (useful for redirections)

Support edition of several fields on the same map, via django-leaflet new feature (fixes #53)

26.311 0.24.3 (2014-06-27)

BUG fixes
- Fix cursor not removed when terminating topology (fixes #1134)
- Fix information desk geometry hard-coded SRID

26.312 0.24.2 (2014-06-27)

BUG fixes
- Fix EPSG :32620 projection file
- Fix JS error when path layer is not on map
- Fix start and end markers not shown as snapped on path edition (fixes #1116)
- Fix groups not shown in Adminsite with external authent (fixes #1118)
- Use markers as mouse icons for topology creation, use resize cursors as fallback only (fixes #1100)
- Minor changes in trek print template (ref #1120)

26.313 0.24.1 (2014-06-26)

BUG fixes
- Fix SVG files for difficulty pictograms
- Fix group fixtures for « Rédacteurs » (fixes #1128)
- Fix tab « None » in list view (fixes #1127)
- Fix external datasources icons in Admin (fixes #1132)
- Fix information desk maps in Admin forms (fixes #1130)
- Fix topology edition when two forced passages on same path (fixes #1131)

Minor changes
- Ordered log entries by date descending (ref #1123)
- Renamed « Data sources » by « External data sources » (fixes #1125)
- Renamed « Foncier » to « Statuts » (fixes #1126)

26.314 0.24.0 (2014-06-23)

** Breaking changes **
- POI icons shall now have a solid background, since no background is added in trek detail map anymore.
- Pictograms fields were added to trek difficulty, route, network. You can use the images provided in the `trekking/fixtures/upload/` folder.

notes

Just before upgrading, delete the following folders

```
rm -rf lib/src/django-modeltranslation
```

After upgrading, mark all treks as published in the languages of your choice

```
UPDATE o_t_itineraire SET public_fr = TRUE;
UPDATE o_t_itineraire SET date_publication = now();
```

New features
- Public TREK export - hide block label if value is empty (fixes #873)

— Add POIs on trek GPX (fixes #774)
— Close list filter when click outside (fixes #916)
— Rename recurrent field to subcontracting on intervention (fixes #911)
— Rename comments field to description on intervention (fixes #927)
— Show object type in ODT export (fixes #1000)
— Show paths extremities on map (fixes #355)
— Ability to reuse topology when adding objects from detail pages (fixes #574, fixes #998)
— Command to generate all elevation charts (fixes #799)
— SITRA support in Tourism datasources (fixes #1064)
— Added status field on feedback reports (fixes #1075)
— Show restricted areas by type in layer switcher (fixes #961)
— Publication status is now controlled by language (fixes #1003). Previous behaviour can restored by setting `TREK_PUBLISHED_BY_LANG` to False.
— Added publication date on trek (ref #1003)
— Ability to see a trek in the different published languages (ref #1003)
— A trek can now have several information desks (fixes #1001)
— Information desks are now shown in trek detail map (fixes #1001)
— Information desks now have optional photo and position, as well as some additional fields (fixes #1001)
— Disabled marker cluster in trek detail map
— Remove background and halo effect on POI icons
— Added 3 new settings to control trek detail map icons size (`TREK_ICON_SIZE_POI`, `TREK_ICON_SIZE_PARKING`, `TREK_ICON_SIZE_INFORMATION_DESK`)

Minor features

— Intervention disorders is not mandatory anymore (fixes #661)
— Improved details in trek form, use Chosen for many-to-many widgets
— Documented the configuration of map layers apparence
— Show layers colors in layer switcher
— Detail page : replace « Maintenance » by « Works » (fixes #889)
— Detail page : interventions on paths are now grouped together, and a small icon is shown (fixes #735)
— Detail page : show intervention costs (ref #958, fixes #764)
— Show project intervention total costs (fixes #958)
— Allow to override the Trek public document template (see *advanced configuration* in docs)
— Close calendar after date choice in intervention form (fixes #928)
— Renamed Attachment submit button (fixes #925)
— Added a new setting `PATH_SNAPPING_DISTANCE` to control paths snapping distance in database (default : 1m)
— Allow to disable trails notion (fixes #997) (see *advanced configuration* in docs)
— Show POI name on hover instead of category in trek detail pages (fixes #1004)
— Form tabs are now always visible while scrolling (fixes #926)
— New URL to obtain the attached filelist of an object
— Remove float notation in altimetry altitude labels
— Control altimetry profiles font using `ALTIMETRIC_PROFILE_FONT` setting
— Add pictograms to routes and networks (fixes #1102)

Bug fixes

— Fixed Signage and Infrastructure year filter label (fixes #293)
— Fixed paths layers not always shown below other layers (fixes #912)
— Clarify legend and title for attachments (fixes #888)
— Fixed cannot clear trek fields in database (fixes #1095)
— Fixed missing translation of « Load local file » (fixes #1085)
— POI types are displayed as such in adminsite
— Fix duplicate authors in history list in detail pages

Internal changes

— Added pictogram on difficulty, useful for *Geotrek-mobile* (fixes #1109)
— Added experimental *Geotrek-light* support (ref #1019)

26.315 0.23.5 (2014-06-19)

Bug fixes

— Fix crash when TourInFrance has malformed website or phone
— Fix translations not being installed

26.316 0.23.4 (2014-06-18)

Bug fixes

— Fix massive upgrade bug, where new migrations were ignored. Due to migration operation introduction in 0.22 installation script.

Special thanks to Noël Martinon, Félix Merzeau, Gil Deluermoz and Camille Montchicourt for their patience on this.

26.317 0.23.3 (2014-06-18)

** Bug fixes **

— Fix static files compression when using Google Mercator projection in maps
— Fix intermediary points order in topology de/serialization, and remove useless topology serialization optimizations (fixes #1031)

26.318 0.23.2 (2014-06-13)

** Bug fixes **

— Fixed land records not shown in detail pages
— Fixed JSON DEM area extent for treks
— Fixed targets list for tourism datasources (fixes #1091)
— Cache tourism datasources for one day (setting `CACHE_TIMEOUT_TOURISM_DATASOURCES`)
— Fix crashes with TourInFrance sources
— Add link to OSM in feedback email (fixes #1089, #1093)
— Fix feedback email translation (fixes #1087)
— Fix problem with permission check « read attachment » in detail page (fixes #1092)
— Fix measure control appearing twice in forms (fixes #1078)
— Fix 404 on download buttons from list views
— Fix POI translated fields not tabbed (fixes #1065)
— Fix missing translation of « Add a new POI » (fixes #1086)
— Fix invalid snapping when save path without editing geometry (fixes #1099)
— Add missing properties in feedback report detail page.
— Hide all modules information in report detail page.
— Add missing translations of feedback module.
— Show object type in ODT export (fixes #1000)

** Internal changes **

— Upgraded to Mapentity 1.4.0
— Upgraded to Leaflet 0.7.3

** Installation **

— Fixed content types migration of land to zoning apps (Thanks Noël Martinon)
— UbuntuGIS stable maintainers have *upgraded* (sic) GDAL to 1.10.0. Upgrading GDAL is painful, and PostGIS packages may have to be reinstalled (data shouldn't be lost though). *Remember it was recommended to run PostGIS on a different server.*

notes

On June 2th 2014, the Ubuntu GIS stable repository switched from `libgdal1` to `libgdal1h`. It broke the deployment script of many projects, including *Geotrek*.

It is a good thing, since it paves the way for the last Ubuntu LTS release (14.04). However, it breaks the *Long Term Support* philosophy of the previous one (12.04), supposed to be supported until 2019.

Morality : we cannot trust the *Ubuntu GIS stable* repository anymore.

Regarding *Geotrek*, such upgrades of Ubuntu packages is not supposed to be covered by its installation script. If you face any problems, please refer to the community or commercial support (such as *Makina Corpus* for example).

26.319 0.23.1 (2014-05-22)

** Bug fixes **
- Fixed regression when editing topologies without modification
- Fixed widget for Trails to allow linear topologies only

26.320 0.23 (2014-05-22)

** Breaking changes **

Read all release notes carefully.
- Trails are now managed as topologies (fixes #370). Existing trails geometries are likely to be **LOST** (*see below*)
- Rename `mailadmin` to `mailadmins` in `etc/settings.ini`
- Permission systems has been refactored (*see below*)

** Bug fixes **
- Force browser cache revalidation of geojson data (fixes #843)
- Force browser cache revalidation for path graph (fixes #1029)
- Fix deletion porblems in AdminSite (fixes #1008)
- Trek advised parking and public transport are translatable (fixes #1024)
- Fix missing translation « no filters » and « current criterias » (fixes #884)
- Fix PDF versions of documents not being translated (fixes #1028)

** New features **
- Command to import shapefile with points into POI as topologies (fixes #952)
- Add views to serve DEM on object area as JSON (*Geotrek-Rando 3D*)
- New tourism module : external datasources can be configured from Adminsite (*GeoJSON, TourInFrance, …*) and added to maps (by module, or published on *Geotrek-rando…*)
- Show number of attached files in tab (fixes #743)
- New permission to control download of attachments
- New permission to allow users or groups to bypass structure restrictions
- Add a setting to serve attached files as download (default : True) (fixes #976)
- Track objects creations, changes and deletions (fixes #300)
- Added a reader group (fixes #495)
- Topologies are not recreated if user did not edit field (fixes #833)
- Added static file for projection EPSG :32620
- Show land objects in menu (fixes #942)
- Documented configuration of custom projections (fixes #1037)
- Buttons in the list menu to add new objects easily
- Add fullscreen button on maps (fixes #904)
- Add all controls on detail map (fixes #907)
- Add a button to close filters (fixes #424)
- Added new sections in documention : *FAQ*, *User-manal* and *Advanced configuration*

** Internal changes **

— Enabled database connection pooling in production
— An error is raised if SRID has not unit in meters (fixes #921)
— Zoning and land modules are now splitted (fixes #954)
— Complete refactor of geographical form fields. Now uses *django-mapentity* from its own repository instead of internal orphan branch.
— Complete refactor of maps initialization, without inline preprocessed JavaScript
— Rely on Django permissions to control access to detail, list and exports (fixes #675)
— Core and altimetry modules are now splitted (fixes #996)
— Renamed treks POIs GeoJSON properties

notes

— Before upgrading, backup your trail records and geometries, using pgAdmin

```
CREATE TABLE backup_sentiers AS SELECT * FROM l_v_sentier;
CREATE TABLE backup_troncons_sentiers AS (
  SELECT l_t_troncon.id AS troncon, l_t_sentier.id, l_t_sentier.nom
  FROM l_t_troncon, l_t_sentier
  WHERE l_t_sentier.id = l_t_troncon.sentier
);
```

— Before upgrade, rename `mailadmin` to `mailadmins` and add a new line `mailmanagers` in `etc/settings.ini`. See *Email settings* section in documentation.
— Just before upgrading, delete the following folders

```
rm -rf lib/src/django-modeltranslation lib/src/mapentity
```

notes

— After upgrading, load the default permissions of the previous groups, otherwise users won't have access to their modules

```
bin/django loaddata geotrek/authent/fixtures/minimal.json
bin/django loaddata geotrek/authent/fixtures/basic.json
```

— After upgrading, make sure *Active* is checked for the user *__internal__* otherwise screenshotting won't work.
— After upgrading, load basic data for the new module

```
bin/django loaddata geotrek/feedback/fixtures/basic.json
```

— After upgrading, make sure the user specified in *Geotrek-rando* is in the group *Geotrek-rando*, or has at least the following permissions in the AdminSite :
  — `paperclip | attachment | Can read attachments`
  — `trekking | Trek | Can read Trek`
  — `trekking | Trek | Can export Trek`
  — `trekking | POI | Can read POI`
  — `trekking | POI | Can export POI`
  — `feedback | Report | Can add report`
— After upgrading, compare visually the resulting migrated trails using QGis, by opening both layers `l_v_sentier` and `backup_sentiers`.

26.321 0.22.6 (2014-04-27)

— Remove hard-coded mentions of EPSG :2154 in database initial migrations (fixes #1020)
— Fix version download and unzip in installation script.

Thanks Noël Martinon, from Guadeloupe National Park, for reporting both issues.

26.322 0.22.5 (2014-03-19)

— Fix compilation of translations (ref #970)
— Fix distinction between languages and translated languages (fixes #968)
— Fix history tabs not being shown after upgrade to Django 1.6 (fixes #975)
— Fix regression on land layer label colors (fixes #980)
— Fix attached files not shown after file upload/delete (fixes #933)
— Fix links being removed from trek descriptions (fixes #981)
— Fix missing thumbnail in trek and POI detail pages
— Fix black background on map captures (fixes #979)
— Increased scale text size on map captures (fixes #850)
— Show map attributions on map captures (fixes #852)
— Fix aspect ratios of map in trek public documents (fixes #849)
— Fix objects list not being filtered on map extent (fixes #982)
— Fix coherence of map layer when text search in objects list (fixes #702)
— Fix number of results not refresh on text search (fixes #865)
— Added north arrow in map image exports (fixes #851)
— Removed darker effect on backgrounds for map image exports, and added internal advanced setting `MAPENTITY_CONFIG['MAP_BACKGROUND_FOGGED'] = True`

26.323 0.22.4 (2014-03-06)

— Fix install.sh not compiling locale messages (fixes #965)
— Moved trek completeness fields to setting *TREK_COMPLETENESS_FIELDS*. Duration and difficulty were added, arrival was removed (fixes #967)
— Fix regression about source locale messages (fixes #970)
— Fix regression link *Back to application* lost from adminsite (fixes #971)
— Serve uploaded files as attachments (fixes #972)
— Remove help texts being shown from filter forms (fixes #966)
— Fix form pills for translated languages (fixes #968)

26.324 0.22.3 (2014-02-17)

— Fix install.sh help not being shown
— Fix screenshots being empty if deployed behind reverse proxy with rool url (fixes #687)
— Fix GPX file layer circle marker size (fixes #930)
— Remove JS libraries from login page
— Fix install.log being removed during installation
— Fix execution characters being shown during DB backup prompt
— Fix PhantomJS and CasperJS installation and deployment
— Added more automatic frontend tests
— Default allowed hosts is now *

26.325 0.22.2 (2014-02-14)

— Fix secured media URLs when using a non empty *rooturl* setting
— Fix proxy errors by disabling keep-alive (fixes #906)

26.326 0.22.1 (2014-02-13)

— Prevent install script to delete existing media files from disk in some situations.

26.327 0.22 (2014-02-12)

Before upgrade
— Backup your database.
— If you upgrade in the same application folder, first delete the *geotrek* sub-folder.
— Use *install.sh* to upgrade (*make deploy* won't be enough)
— After upgrade, make sure the following query returns only ~23 results :

```
SELECT COUNT(*) FROM south_migrationhistory;
```

BREAKING changes
— For upgrades, Geotrek 0.21 is required.
— Uploaded files are now restricted to authenticated users (fixes #729)

notes
Geotrek-rando 1.23 or higher is required to synchronize content.

NEW features
— In list view, click on map brings to detail page, mouse over highlights in list.
— Show path icon if intervention is not on infrastructure (fixes #909)
— Add spanish translation
— Add photographie into default attachments filetype
— Map location combobox (Cities, Districts, Areas) are not shown if empty or disabled.
— Several database views have been created (fixes #934)
— Remove dots from path icon (fixes #939)
— Intervention, infrastructure and project filters list of years is now dynamic (fixes #948)
— Application available languages (*english*, *french*, *italian*, *spanish*) are now distinct from translated content languages (*languages* value in `settings.ini`)

Minor changes
— Improved apparence of map controls
— Improved apparence of path intermediary points
— Improved apparence of form validation buttons
— Add auto-generated docs at /admin/doc/
— Nicer installation script output

Installation script
— Scan and ortho attributions can now be set using *scan_attributions* and
— Propose to backup DB before Geotrek upgrade (fixes #804)
— Settings edition prompt only happens at first install *ortho_attributions* in *settings.ini*.

BUG fixes
— Fix convert urls behind reverse proxy with prefix
— Fix deployment problem if `layercolor_others` not overidden in settings.ini
— Fix topology kinds to be "INTERVENTION" for intervention without signage/infrastructure
— Fix restricted areas types display in admin (fixes #943)

— Fix list ordering of trek relationships and web links (fixes #929)
— Fix nginx log files being already empty after logrotate (fixes #932)
— Fix project add button when no permission

notes

List of restricted areas is not shown on map by default anymore. Restore previous behaviour with advanced setting *LAND_BBOX_AREAS_ENABLED* as True.

Internal changes

— Upgrade to Django 1.6 (fixes #938)
— Upgrade to Leaflet 0.7
— Upgrade a great number to python and JavaScript libraries
— An internal user (with login permission) is used to authenticate the Conversion and Capture services.
— Installation script is modular (standalone, geotrek only, …)
— Developement server now listens on all interfaces by default
— Database migrations were resetted, no postgres *FATAL ERROR* message will be emitted on fresh install anymore (fixes #937). See *Troubleshooting* in documentation.

26.328 0.21.2 (2014-02-04)

BUG fixes

— Warn on tiling landscape/portrait spatial extent only if map with local projection
— Safety check on thumbnailing if images are missing from disk (*useful for troubleshooting, when importing existing dumps*).
— Fix overlapping filter if no records present (fixes #931)

26.329 0.21.1 (2013-12-11)

Improvements

— Smooth DEM drapping, improving altimetric information and profiles (fixes #840, ref #776)

BUG fixes

— Signage forms are now restricted by structure (fixes #917)
— Fix geometries computation when path split occurs on return topology (fixes #899)
— Add title on links in list views (fixes #913)
— Prevent horizontal scroll on forms, caused by textareas (fixes #914)
— Fix empty 3d geometry of point topologies with offset (fixes #918)

notes

In order to recompute all paths topologies altimetry information, you can perform the following queries :

```
UPDATE l_t_troncon SET geom = geom; UPDATE e_t_evenement SET decallage =
decallage;
```

Reading information from rasters is costly. Be prepared to wait for a while.

26.330 0.21 (2013-11-28)

Improvements

— Increase height of multiple select (fixes #891)
— Add project field in intervention filter (fixes #896)
— Many minor improvements for infrastructures in adminsite (fixes #886)
— Add category in intervention filter (fixes #887)

BUG fixes

— Fix KML coordinates not being in 3D.
— GPX now has trek description (fixes #775)
— Order overlapping topologies by order of progression (fixes #777)
— Improved TinyMCE configuration, for resize and cleanup (fixes #351, #711)
— Changed trek duration interval for notion of days (fixes #880)
— Show city departure in trek public export (fixes #881)
— Document customization of TinyMCE config (fixes #882)
— Fix 404 error on path delete (fixes #900)
— Fix project constraints not being displayed in details (fixes #893)
— Fix organism translation in project form (fixes #892)
— Fix apparence of forms on small screen (fixes #744, #902)
— Fix modify button being hidden to editors (fixes #901)
— Fix overlap between map controls and label (fixes #883)
— Fix translation of district in list filters (fixes #890)
— Fix integrity error on land intersection on path update (fixes #897)
— Fix form layout problems (fixes #712, #879)

26.331 0.20.9 (2013-10-30)

— Fix altimetric profile if topology geometry is wrong (fixes #875)
— Fix apparence of creation button in intervention list (fixes #877)
— Fix topology geometries that were sampled like paths 3D geometry (fixes #878)
— Fix topology lines geometries join in some situations (ref #722)
— Fix topology not well displayed if start/end on intersection (fixes #874)

26.332 0.20.8 (2013-10-22)

— Public trek export : Fix various layout regressions (ref #848)
— Public trek export : Show POI theme pictogram (fixes #858)
— Public trek export : full width for information desk frame (fixes #856)
— Public trek export : add footer with trek title and page numbers (fixes #861)
— Public trek export : add floating picture in POI detail (fixes #860)
— Public trek export : fix POI thumbnails missing (fixes #869)
— Fix point offset lost on path update (fixes #867)
— Fix reconnect point topologies with offset to closest path (fixes #868)

26.333 0.20.7 (2013-10-16)

— Fix topology geometry 3D being draped twice (fixes #863)
— Altimetric profile : Show max distance and round values (fixes #853)
— Altimetric profile : Add settings for colors (fixes #854)
— Public trek export : POIs list in two columns (fixes #855)
— Public trek export : POIs details without column break (fixes #857)
— Public trek export : Show pictures attributions (fixes #859)
— Public trek export : Use 10pt fonts in every text blocks (fixes #848)

notes
Empty profiles cache rm -rf var/media/profiles/*

26.334 0.20.6 (2013-10-14)

— Remove 3D from JS WKT serializer
— Safety check if path is less than 1m
— Remove mentions of 2154 projection in schema migrations
— Fix performance issues in altimetric JSON (fixes #845)
— Fix filter forms missing from Trek and POI lists (fixes #847)
— Fix empty Nginx log files (fixes #846)

26.335 0.20.5 (2013-10-09)

— Fix migration of draping utility function

26.336 0.20.4 (2013-10-09)

— Fix sort stake by id (level) (fixes #835)
— Rename stake to maintenance stake (fixes #834)
— Add validity to path filter (fixes #836)
— Do not redrape topology geometries, use path 3D geometry (fixes #832)
— Fix document export of Trail objects (fixes #839)
— Fix trail helpers for land layers (fixes #838, ref #842)
— Fix install on fresh folder, missing folder `lib/src` (fixes #844)

26.337 0.20.3 (2013-09-30)

BUG fixes
— Fix typo in french translation of Properties (fixes #815)
— Fix missing description from infrastructure/signage detail page (fixes #816)
— Fix Cities / Districts / Restricted Areas in project detail page (fixes #817)
— Fix only deleted topology can have geom = NULL (fixes #818)
— Fix geometries not editable in QGis by switching path and topologies geometries to 2D (fixes #688)
— Fix altimetric sampling precision setting not taken in account in SQL (ref #776)

26.338 0.20.2 (2013-08-27)

— Fix convert urls behind reverse proxy with prefix
— Fix Trek public print conversion
— Fix display of trek length in public document (one decimal only)
— Fix altimetric graph delaying map display in detail pages

notes
Empty maps captures cache rm -rf var/media/maps/trek-*

26.339 0.20.1 (2013-08-26)

— Add DB index for start and end columns
— Merge gunicorn logs with respective applications logs
— Lower logging level in production (WARNING -> INFO)

BUG fixes
— Fix deployment error with application's TITLE
— Fix deployment errors with mandatory external authent values
— Fix trek export layout : fit map image and altimetric profile in one page.

26.340 0.20 (2013-08-23)

— Edit difficulty id in Admin site, mainly used to order difficulty levels (fixes #771)
— Use explicit list of fields in forms, instead of excluding model fields (fixes #736). Issue #712 was closed too, since most suspected cause was field listings. Please re-open if problem re-appears.
— Fix timeout on POI Shapefile and CSV exports (fixes #672)
— Altimetric profiles are now computed in PostGIS (fixes #778, #779)
— Positive and negative ascents are now computed using more DEM resolution (fixes #776)

notes
Setting `PROFILE_MAXSIZE` was replaced by `ALTIMETRIC_PROFILE_PRECISION` which controls sampling precision in meters (default : 20 meters)

— Altimetric profiles were removed from object map images
— Altimetric profiles are now plotted using SVG
— Altimetric profiles are now inserted into path documents and trek public printouts (ref #626)
— Fix deletion of associated interventions when editing infrastructures (fixes #783)
— Latest record is updated (*touch*) when a DELETE is performed on table (refreshs cache) (fixes #698)
— Reworked settings mechanism to follow Django best practices

notes
Replace all computed values from `etc/settings.ini`. For example, replace « 60 * 60 » by 3600. (You can increase this value to several hours by the way)

— Allow server host to capture pages (fixes #733)
— Adjust map capture according to geometry aspect ratio (fixes #627)
— Always show path layer in detail pages (fixes #781)
— Fix restore of topology on loop paths (fixes #760)
— Fix topology construction when loop is formed by two convergent paths (fixes #768)
— Added small tool page at `/tools/extents/` to visualize configured extents (ref #732)

— Removed setting `spatial_extent_wgs84`, now computed automatically from `spatial_extent`, with a padding of 10%.

notes
Have a look at `conf/settings.ini.sample` to clean-up unnecessary values from your settings file.

— Fix paths offset for portrait spatial extent (fixes #732)
— Rely on Tilecache default max resolution formulae (fixes #732)
— Due to bug in Leaflet/Proj4Leaflet (https://github.com/kartena/Proj4Leaflet/issues/37) landscape spatial extents are not supported. Please adjust spatial_extent to be a portrait or square, or application will raise *ImproperlyConfiguredError*.
— Reload map objects on zoom out too (fixes #435)
— Fix computation of *min_elevation* for point topologies (fixes #808)

notes
In order to recompute all paths topologies altimetry information, you can perform the following query : `UPDATE e_t_evenement SET decallage = decallage;`. Reading information from rasters is costly. Be prepared to wait for a while.

26.341 0.19.1 (2013-07-15)

— Restore `pk` property in Trek GeoJSON layer

26.342 0.19 (2013-07-12)

— Intervention length field (readonly if geometry is line)
— Fix apparence bug if no rights to add treks and pois (fixes #713)
— Fix extremities snapping (fixes #718)
— Show information desk in trek detail page (fixes #719)
— Fix topology adjustments after path split (fixes #720)
— On edition show global line orientation instead of individual paths (fixes #679)
— Fix invalid topology if trek goes twice on same path (fixes #671)
— Overlapping is now more precise (fixes #710)
— Reworked trek print layout
— Fix topology building if paths are taken twice (fixes #722)
— Fix tiling offset with horizontal bboxes
— Fix display of POI layer by default on list (fixes #696)
— Fix translation of not validated paths (fixes #730)
— Fix error if topology is required and empty (fixes #745)
— Fix duplication of N-N relations on path split (fixes #738)
— Fix project map in detail page (fixes #734)
— Fix project listed deleted interventions (fixes #739)
— Fix project listed infrastructures through interventions (fixes #740)
— Fix saving intervention form on infrastructure
— Repair serializing of properties after upgrade of django-geojson (fixes #755)
— Added `public_transport` and `advised_parking` to trek JSON detail API (fixes #758)
— Repair land layers colors after upgrade of django-geojson
— Upgraded to django-geojson 2.0
— Upgraded to Django 1.5

notes
Specify allowed host (server IP) in `etc/settings.ini` (*for example*) : * `host = 45.56.78.90`
Empty object caches : * `sudo /etc/init.d/memcached restart` * `rm -rf ./var/cache/*`

26.343 0.18 (2013-06-06)

— Add pretty trek duration in JSON
— Add information desk field in Trek (fixes #624)

26.344 0.17 (2013-05-17)

— Show trek duration as human readable in minutes, hours and days (fixes #471, #683)
— Fix hover on paths that interfered with clic for topology creation (fixes #680)
— Run API urls on different workers (ref #672)
— Fix redirect to root url after logout (fixes #264)
— Fix redirect to next after login (fixes #381)
— Switch to Memcached instead of local memory in production
— Move secret key to settings.ini
— Relate paperclip FileType to Structure (fixes #256)
— Relate PhysicalTypes to Structure (fixes #255)
— Relate Organisms to Structure (fixes #263)
— Compute max_resolution automatically
— Fix creation and edition of interventions on infrastructures (fixes #678)
— Change default objects color to yellow
— Restored Italian translations
— Fix regex for RAISE NOTICE (fixes #673)
— Initial public version

See project history in *docs/history.rst* (French).

Index

V

variable d'environnement
ACCESSIBILITY_ATTACHMENTS_ENABLED, 107
ALERT_DRAFT, 107
ALERT_REVIEW, 107
ALLOW_PATH_DELETION_TOPOLOGY, 107
API_IS_PUBLIC, 97
BLADE_CODE_FORMAT, 108
BLADE_CODE_TYPE, 108
COLORS_POOL, 101
COMPLETENESS_FIELDS, 145
COMPLETENESS_LEVEL, 145
DIVING_INTERSECTION_MARGIN, 152
ENABLE_JOBS_COSTS_DETAILED_EXPORT, 120
ENABLE_REPORT_COLORS_PER_STATUS, 117
ENABLED_MOBILE_FILTERS, 156
FACEBOOK_APP_ID, 147
FLATPAGES_ENABLED, 106
HIDDEN_FORM_FIELDS, 139
HIDE_PUBLISHED_TREKS_IN_TOPOLOGIES, 153
INSTALLED_APPS for API V2, 97
INSTALLED_APPS for Diving, 109
INSTALLED_APPS for Outdoor, 109
INSTALLED_APPS for Sensitive areas, 110
LANDEDGE_MODEL_ENABLED, 106
LEAFLET_CONFIG, 101
LEAFLET_CONFIG['OVERLAYS'], 103
LEAFLET_CONFIG['TILES'], 99
LINE_CODE_FORMAT, 108
MAP_CAPTURE_SIZE, 105
MAPENTITY_CONFIG for characters, 146
MAPENTITY_CONFIG for custom colors in PDF, 150
MAPENTITY_CONFIG for layers color and style, 101
MAPENTITY_CONFIG for map background, 105
MAPENTITY_CONFIG for medias, 118
MOBILE_ASCENT_INTERVALS, 155
MOBILE_DURATION_INTERVALS, 156
MOBILE_LENGTH_INTERVALS, 155
MOBILE_NUMBER_PICTURES_SYNC, 155
MOBILE_TILES_URL, 155
ONLY_EXTERNAL_PUBLIC_PDF, 154
OUTDOOR_COURSE_POINTS_OF_REFERENCE_ENABLED, 104
PAPERCLIP_ALLOWED_EXTENSIONS, 120
PAPERCLIP_EXTRA_ALLOWED_MIMETYPES, 120
PAPERCLIP_MAX_BYTES_SIZE_IMAGE, 119
PAPERCLIP_RESIZE_ATTACHMENTS_ON_UPLOAD, 119
PATH_MERGE_SNAPPING_DISTANCE, 103
PATH_SNAPPING_DISTANCE, 103
PRIMARY_COLOR, 151
SEND_REPORT_ACK, 113
SENSITIVE_AREA_INTERSECTION_MARGIN, 110
SENSITIVITY_DEFAULT_RADIUS, 110
SNAP_DISTANCE, 103
SPLIT_DIVES_CATEGORIES_BY_PRACTICE, 153
SPLIT_TREKS_CATEGORIES_BY_ACCESSIBILITY, 153
SPLIT_TREKS_CATEGORIES_BY_ITINERANCY, 153
SPLIT_TREKS_CATEGORIES_BY_PRACTICE, 153
SYNC_MOBILE_OPTIONS, 154
SYNC_MOBILE_ROOT, 154
SYNC_RANDO_OPTIONS, 151
SYNC_RANDO_ROOT, 151
THUMBNAIL_COPYRIGHT_FORMAT, 146
THUMBNAIL_COPYRIGHT_SIZE, 146
TINYMCE_DEFAULT_CONFIG, 145
TOPOLOGY_STATIC_OFFSETS, 104
TOURISM_ENABLED, 106
TOURISM_INTERSECTION_MARGIN, 152
TRAIL_MODEL_ENABLED, 106
TREK_EXPORT_INFORMATION_DESK_LIST_LIMIT, 152
TREK_EXPORT_POI_LIST_LIMIT, 152
TREK_POI_INTERSECTION_MARGIN, 109
TREK_POINTS_OF_REFERENCE_ENABLED, 104
TREK_WITH_POIS_PICTURES, 154

USE_BOOKLET_PDF, 150