The REG106_PRIOR Table

Welcome to table **REG106_PRIOR**. As the name suggests, this table contains the data of the claimed priorities. In particular, it includes information about the authority, the priority kind, publication in the EPO bulletin, among others.

In [1]:
```
from epo.tipdata.patstat import PatstatClient
from epo.tipdata.patstat.database.models import REG106_PRIOR
from sqlalchemy import func
import pandas as pd

# Initialise the PATSTAT client
patstat = PatstatClient(env='PROD')

# Access ORM
db = patstat.orm()
```

ID (Primary Key)

Technical identifier for an application, without business meaning. Its values will not change from one PATSTAT edition to the next.

In [3]:
```
i = db.query(
    REG106_PRIOR.id
).limit(1000)

df = patstat.df(i)
df
```

Out[3]:

	id
0	21882091
1	18713428
2	18809269
3	23902595
4	83300392
...	...
995	3781119
996	23721243
997	12822321
998	13824730
999	17759688

1000 rows × 1 columns

CHANGE_DATE

It is the date of when the record was saved in the database.

In [4]:
```
change_date = db.query(
    REG106_PRIOR.change_date,
    REG106_PRIOR.id
).limit(100)

change_date_df = patstat.df(change_date)
change_date_df
```

Out[4]:

	change_date	id
0	2005-01-28	4008804
1	2017-06-23	15725281
2	2005-12-22	5754427
3	2018-06-15	16813463
4	1982-01-08	81106502
...	...	...
95	2023-05-12	23154905
96	2019-07-19	19157595
97	2016-04-22	14795048
98	2024-03-13	21937459
99	2021-11-22	19842665

100 rows × 2 columns

BULLETIN_YEAR

For actions that have been published in the EPO Bulletin, it is the year of the publication in the bulletin. The default value is 0, used for applications that are not published or for which the year is not known. The format is YYYY otherwise.

In [5]:
```
years = db.query(
    REG106_PRIOR.bulletin_year,
    REG106_PRIOR.id
).limit(1000)

years_df = patstat.df(years)
years_df
```

Out[5]:

	bulletin_year	id
0	0	21882091
1	2020	18713428
2	2022	18809269
3	0	23902595
4	1983	83300392
...	...	...
995	2005	3781119
996	0	23721243
997	0	12822321
998	0	13824730
999	2018	17759688

1000 rows × 2 columns

BULLETIN_NR

This is the issue number of the EPO Bulletin for actions that have been published in it. The Bulletin number indicates the calendar week the Bulletin has been published. The default value 0 is used when the attribute `bulletin_year` is 0.

In [6]:
```
bulletin_nr = db.query(
    REG106_PRIOR.id,
    REG106_PRIOR.bulletin_nr,
    REG106_PRIOR.bulletin_year
).limit(100)

bulletin_nr_df = patstat.df(bulletin_nr)
bulletin_nr_df
```

Out[6]:

	id	bulletin_nr	bulletin_year
0	21882091	0	0
1	18713428	1	2020
2	18809269	33	2022
3	23902595	0	0
4	83300392	39	1983
...	...	...	...
95	16861015	36	2018
96	94900217	0	0
97	91111265	12	1992
98	1951794	0	0
99	21828851	0	0

100 rows × 3 columns

PRIOR_SEQ_NR

The ordinal number of the priority in the application where it is claimed. If one or more priorities are given, then they are numbered starting from 1. In case the priority rights have been lost, then there will be one entry with PRIOR_SEQ_NR = 0.

In [14]:
```
prior_seq = db.query(
    REG106_PRIOR.id,
    REG106_PRIOR.prior_seq_nr
).order_by(
    REG106_PRIOR.prior_seq_nr
).limit(50000)

prior_seq_df = patstat.df(prior_seq)
prior_seq_df
```

Out[14]:

	id	prior_seq_nr
0	95107047	0
1	11846468	0
2	12793369	0
3	17161870	0
4	93400550	0
...	...	...
49995	22945500	1
49996	6809043	1
49997	21911193	1
49998	15736673	1
49999	22873985	1

50000 rows × 2 columns

We can count how many applications correspond to a specific priority sequence number.

In [16]:
```
tot_seq = db.query(
    REG106_PRIOR.prior_seq_nr,
    func.count(REG106_PRIOR.id).label('number_of_applications')
).group_by(
    REG106_PRIOR.prior_seq_nr
).order_by(
    REG106_PRIOR.prior_seq_nr
)

tot_seq_df = patstat.df(tot_seq)
tot_seq_df
```

Out[16]:

	prior_seq_nr	number_of_applications
0	0	7047
1	1	6142218
2	2	996150
3	3	259323
4	4	114406
...	...	...
415	415	1
416	416	1
417	417	1
418	418	1
419	419	1

420 rows × 2 columns

PRIOR_KIND

Priority kind. It can be "national", "regional" or "international".

In [13]:
```
kind = db.query(
    REG106_PRIOR.prior_kind,
    func.count(REG106_PRIOR.id).label('number_of_applications')
).group_by(
    REG106_PRIOR.prior_kind
).order_by(
    func.count(REG106_PRIOR.id)
)

kind_df = patstat.df(kind)
kind_df
```

Out[13]:

	prior_kind	number_of_applications
0	international	56894
1	regional	324297
2	national	7362753

PRIOR_AUTH

Authority of the priority document.

Let's count how many priorities for each patent authority, as of now, there are in PATSTAT.

In [9]:
```
auth = db.query(
    REG106_PRIOR.prior_auth,
    func.count(REG106_PRIOR.id).label('number_of_applications')
).group_by(
    REG106_PRIOR.prior_auth
).order_by(
    func.count(REG106_PRIOR.id).label('number_of_applications').d
esc()
)

auth_df = patstat.df(auth)
auth_df
```

Out[9]:

	prior_auth	number_of_applications
0	US	2880425
1	JP	1385716
2	DE	813624
3	CN	559428
4	KR	371437
...	...	...
152	TZ	1
153	LS	1
154	TG	1
155	BW	1
156	YE	1

157 rows × 2 columns

PRIOR_NR

Number of the priority document or indicator that the priority rights have been lost. The domain consists of up to 15 characters: number of priority application or the string ‘deleted’.

In [10]:
```
publn_nr = db.query(
    REG106_PRIOR.prior_nr,
    REG106_PRIOR.prior_auth
).limit(1000)

publn_nr_df = patstat.df(publn_nr)
publn_nr_df
```

Out[10]:

	prior_nr	prior_auth
0	201762469311P	US
1	20100104201	JP
2	201562183448P	US
3	20050654060P	US
4	202016823869	US
...	...	...
995	20040606036P	US
996	20170226158	JP
997	201762579610P	US
998	19980021209	JP
999	20150905155	AU

1000 rows × 2 columns

PRIOR_DATE

Filing date of the priority document.

In [11]:
```
publn_date = db.query(
    REG106_PRIOR.prior_nr,
    REG106_PRIOR.prior_date
).limit(1000)

publn_date_df = patstat.df(publn_date)
publn_date_df
```

Out[11]:

	prior_nr	prior_date
0	202022370436U	2020-10-22
1	201762464276P	2017-02-27
2	201862619752P	2018-01-20
3	202211589773	2022-12-12
4	19820349794	1982-02-18
...	...	...
995	20020523378	2002-12-24
996	202263401858P	2022-08-29
997	20110078261	2011-08-05
998	20120171121	2012-08-01
999	20160038328	2016-02-29

1000 rows × 2 columns

In [ ]: