

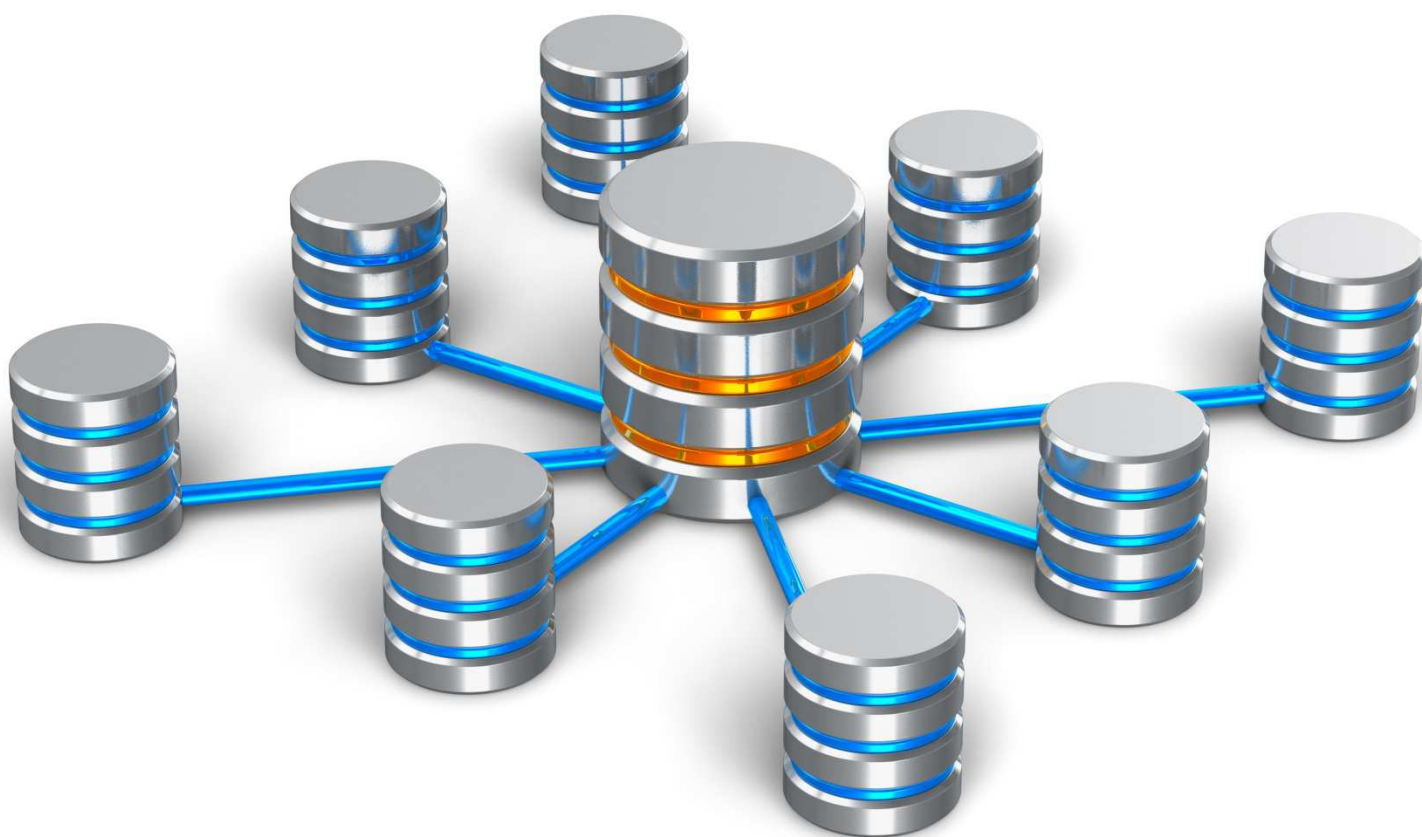
JRC TECHNICAL REPORTS

SETIS Database: data management for R&I key performance indicators

*Design and
implementation*

Pasimeni, F., Fiorini, A.

2017



This publication is a Technical report by the Joint Research Centre (JRC), the European Commission's science and knowledge service. It aims to provide evidence-based scientific support to the European policymaking process. The scientific output expressed does not imply a policy position of the European Commission. Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use that might be made of this publication.

Contact information

Name: Francesco Pasimeni

Email: Francesco.PASIMENI@ec.europa.eu

Tel.: +31 224565243

JRC Science Hub

<https://ec.europa.eu/jrc>

JRC108754

EUR 28853 EN

PDF	ISBN 978-92-79-75369-5	ISSN 1831-9424	doi: 10.2760/37901
Print	ISBN 978-92-79-75370-1	ISSN 1018-5593	doi: 10.2760/99811

Luxembourg: Publications Office of the European Union, 2017

© European Union, 2017

Reuse is authorised provided the source is acknowledged. The reuse policy of European Commission documents is regulated by Decision 2011/833/EU (OJ L 330, 14.12.2011, p. 39).

For any use or reproduction of photos or other material that is not under the EU copyright, permission must be sought directly from the copyright holders.

How to cite this report: Pasimeni, F. and Fiorini, A., *SETIS Database: data management for R&I key performance indicators*, EUR 28853 EN, Publications Office of the European Union, Luxembourg, 2017, ISBN 978-92-79-75369-5, doi:10.2760/37901, JRC108754.

All images © European Union 2017, except: cover page, Oleksiy Mark, 2012. Source: Fotolia.com

Contents

Abstract	1
1 Introduction.....	2
2 Data extraction from PATSTAT	4
2.1 1.Appln_Id.....	5
2.2 2.Person_Id	7
2.3 3.CPC.....	8
2.4 4.Country	9
2.5 6.Company_Family	10
3 Data cleaning.....	11
3.1 Step 1: automatic data cleaning (stage 1)	11
3.1.1 3.1.Sector_DOC_STD_NAME_ID	12
3.1.2 3.2.Ctry_DOC_STD_NAME_ID	13
3.1.3 3.3.Sector_HRM_ID	14
3.1.4 3.4.Ctry_HRM_ID	15
3.1.5 3.5.Nuts_DOC_STD_NAME_ID.....	16
3.1.6 3.6.Nuts_HRM_ID	17
3.2 Step 1: automatic data cleaning (stage 2)	18
3.2.1 c0_from_old_to_new.....	18
3.2.2 c1_distinct_doc_std_name_id.....	19
3.2.3 c2_doct_std_name_id_with_info.....	19
3.2.4 c3_count_doc_std_name_id.....	20
3.2.5 c4_unique_doc_std_name_id	20
3.2.6 c5_unique_on_person_id	21
3.2.7 c6_new_and_multiple_person_id	22
3.2.8 c7_person_id_with_old.....	22
3.2.9 c8_final_to_clean.....	23
3.3 Step 1: automatic data cleaning (stage 3)	24
3.4 Step 2: manual data cleaning.....	26
4 Patent statistics.....	27
4.1 Total fractional	27
4.1.1 q1_List_Fam_Company	27
4.1.2 q2_Count_Fam_Company	27
4.1.3 q3_Fractional	28
4.1.4 q4_Total_Fractional_Company.....	28
4.2 Granted patent families	29
4.2.1 g1_List_Fam_Appln.....	29

4.2.2	g2_List_Fam_Granted	29
4.3	International patent families	30
4.3.1	i1_List_Fam_Auth	30
4.3.2	i2_List_Fam_Int	31
4.4	Fractional and final statistics	32
4.4.1	List of applicant: Company_List	32
4.4.2	o1_List_Fam_CPC	32
4.4.3	o2_List_Fam_Company	33
4.4.4	o3_List_Fam_CPC_Company	33
4.4.5	o4_Count_Fam_CPC_Company	34
4.4.6	o5_Fractional	34
4.4.7	o6_Total	35
4.4.8	Patent Year	36
4.5	Patent projections.....	37
4.5.1	Patent_2010	37
4.5.2	Patent_2011	37
4.5.3	Patent_2012	38
4.5.4	Patent_2013	38
4.5.5	Avg	39
4.5.6	2014e	40
4.5.7	2015e	41
4.6	Trend of Patent	42
5	R&D estimation	43
5.1	e0_List_Subsidaries	43
5.2	e1_List_Groups	44
5.3	e2_Groups_for_UC.....	45
5.4	e3_Sector_UC	45
5.5	e4_R&D_eq7.....	46
5.6	e5_Tech_UC.....	47
5.7	e6_Tech_UC_Total.....	47
5.8	RnD Year	48
5.9	R&D projection	49
5.9.1	Patent_R&D_2010	49
5.9.2	Patent_R&D_2011	49
5.9.3	Patent_R&D_2012	50
5.9.4	Patent_R&D_2013	50
5.9.5	Patent_R&D_Avg	51
5.9.6	RnD_2014e.....	52

5.9.7 RnD_2015e.....	53
5.10 Private R&D	54
6 Conclusions	55
References	56
List of abbreviations and definitions	57
List of figures	58

Abstract

The report aims at presenting the construction of the SETIS Database. It is designed in order to facilitate and accelerate data analysis in the context of monitoring Research, Innovation and Competitiveness of the European Union in the Energy sector. It explains the procedural steps required to implement the database through queries in SQL, and to elaborate patent statistics and estimate R&D expenditure for private companies.

1 Introduction

This technical report aims at presenting the construction of the SETIS Database (DB). The database is built in order to produce Key Performance Indicators (KPIs) needed to assess the Research, Innovation and Competitiveness (RIC) of the European Union (EU) in the Energy sector (European Commission, 2015). The database reflects the methodology presented in Fiorini *et al.* (2017) where the theoretical and conceptual background are explained. Therefore, this report does not deal with these aspects; nevertheless readers are invited to refer to this methodology for further explanations.

This report describes the way in which the database is designed. The need to have an *ad hoc* database, instead of using those available online, depends on several reasons. A dedicated database, defined by a specific research question, facilitates and accelerates the process of data analysis in the context of SETIS. Further, the main data source is PATSTAT, the Worldwide Patent Statistical Database created and maintained by the European Patent Office (EPO), and it has issues with data accuracy and completeness (European Patent Office, 2017). The SETIS database permits to overcome these difficulties, by processing data before the construction of KPIs. Moreover, since the SETIS analysis focuses also on domains beyond patent statistics, a dedicated database gives the opportunity to establish connections to other data sources. Lastly, the database is designed also to keep track of data corrections, which are transferred to future versions of the SETIS DB.

The objective of this report is to have a technical document explaining the procedural steps needed to create the SETIS DB. For this scope, all set of queries, programmed in Structured Query Language (SQL) are presented and explained in the report. The database is constructed by using conventional tools (Microsoft Excel and Microsoft Access) that, unfortunately, do not guarantee high standard of computational capacity. However, the layout of the DB is properly designed in order to work around this issue, as shown in Figure 1.

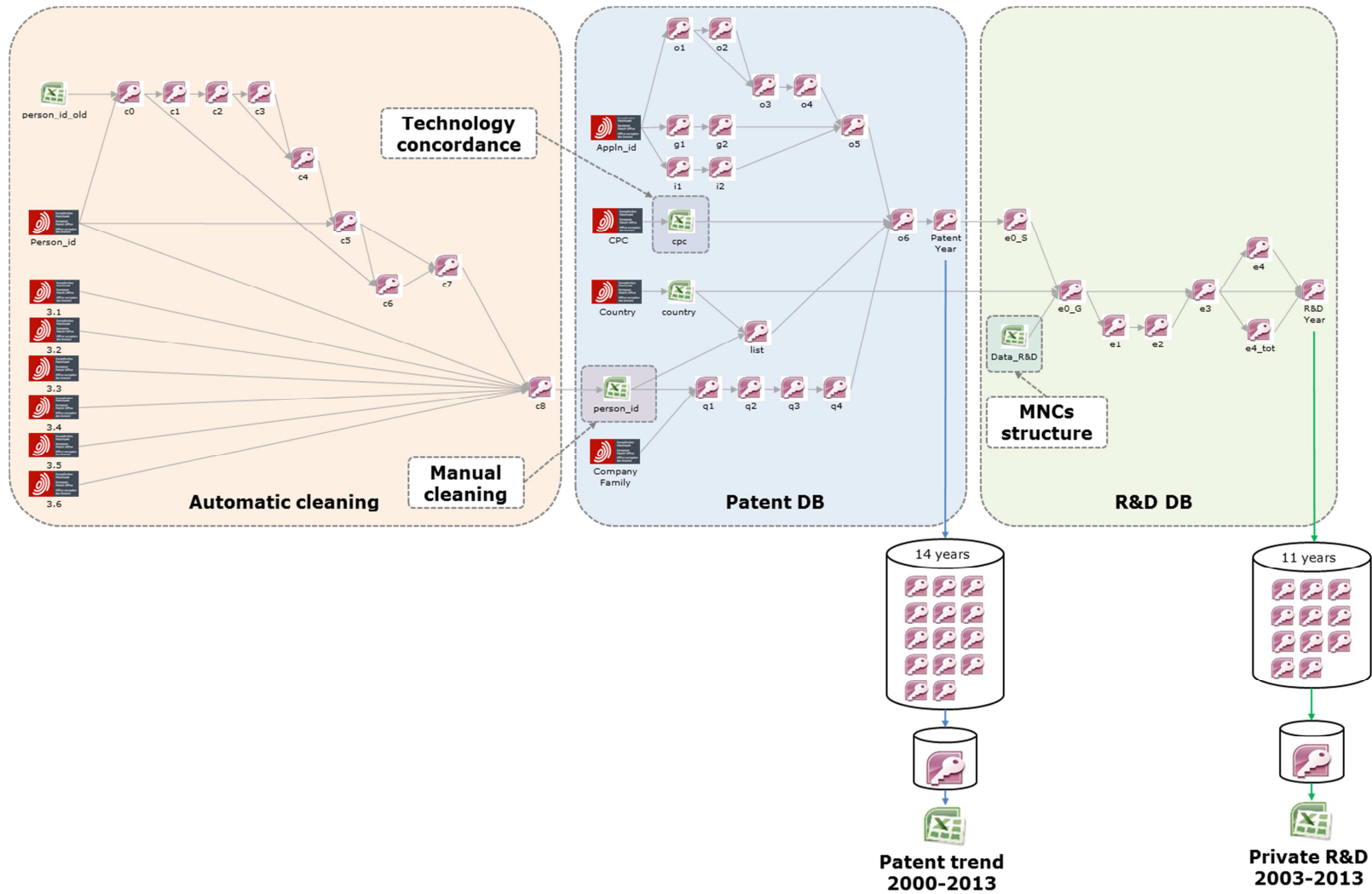
The SETIS DB consists of three main sections. The first one is dedicated to the data cleaning process (section 3). It is the most time-consuming activity since it involves an automatic cleaning procedure and a subsequent manual effort. At the same time, this process is also the most important since it permits to improve quality of the raw data by eliminating errors or blanks entries present in PATSTAT. With the clean dataset, it is possible to calculate patent statistics (section 4). This part of the database analyses patent data in order to construct different type of indicators: total patent families as a proxy of inventions, granted patent families and international patent families. It also provides information on the patent offices where inventions are protected. The final part of the SETIS DB permits to estimate R&D expenditures for private companies (section 5). The estimation procedure uses patent statistics in combination with companies' financial data obtained by additional data sources: the EU Industrial R&D Investment Scoreboard¹ and Orbis².

The three parts of the database are run one after the other. The automatic step of the data cleaning process is the first activity performed and it is run only once at the beginning. The manual cleaning, instead, can go on every time there is the need to improve data quality. These changes are then automatically uploaded to the DB, giving the opportunity to refine indicators continuously. The two following parts, regarding patent statistics and R&D estimates, are split by years. In this way, it is possible to overcome limitations of conventional tools. Hence, both sections are run a number of times, correspondent to the number of years analysed. Then, a final step is designed to combine all information in one all-inclusive outcome.

¹ <http://iri.jrc.ec.europa.eu/scoreboard.html>

² ORBIS database is a Bureau van Dijk (BvD) product: <https://orbis.bvdinfo.com/>.

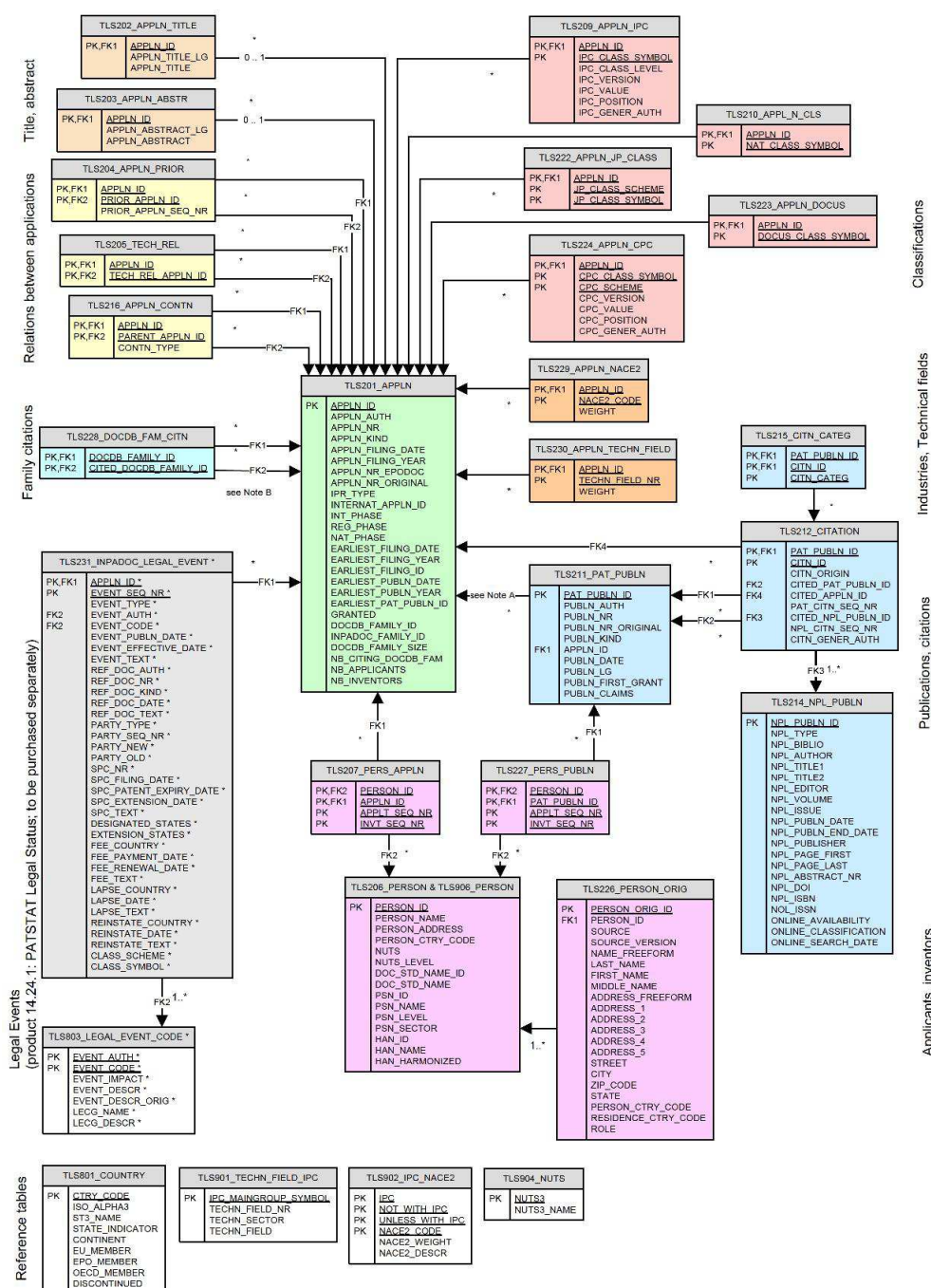
Figure 1. Layout of the SETIS database



2 Data extraction from PATSTAT

PATSTAT contains bibliographical and legal status patent data from leading industrialised and developing countries, extracted from the EPO's databases. In order to facilitate the understanding of this document, the local model diagram of the database is provided (Figure 2)³. This section presents the queries that are run in PATSTAT in order to retrieve data needed for the scope of the SETIS analyses.

Figure 2. PATSTAT logical model diagram



Source: European Patent Office (2017).

³ For further information on the structure, please refer to the documentation section of the PATSTAT database: <https://www.epo.org/searching-for-patents/business/patstat.html#tab-3>.

2.1 1.Appln_Id

This query (Figure 3) extracts the following information:

- List of patent families, and, for each patent family
 - list of patent applications, and, for each application in this family
 - authority office where has been filed,
 - if it is granted or not,
 - the priority year as well as the filing year,
 - the list of person_id of the applicant(s),
 - the total number of both applicant(s) and inventor(s)
 - the CPC code(s) associated

Note that for each year the same query is run. Since the available tools (Microsoft Excel and Microsoft Access) do not provide enough memory to work with many data all together, the strategy is to divide data processing by years. For this reason the where condition "YEAR (tls201_appln.earliest_filing_date) between 20XX and 20XX" is necessary.

The where condition "tls201_appln.appln_kind in ('A','W')" permits to extract only patent (A) or PCT application in the international phase (W). Further, "tls207_pers_appln.applt_seq_nr > 0" is needed to make sure that we select applicant only (European Patent Office, 2016). Moreover, "tls201_appln.appln_id < 900000000" is the condition that permits to eliminate those false patent application introduced in PATSTAT because of database issues. Lastly, "tls224_appln_cpc.cpc_class_symbol like 'Y0%'" defines the boundaries of our dataset, which is limited to Climate Change and Mitigation Technologies.

In the query, the function "SELECT DISTINCT" is used in order to avoid duplicates coming from the table tls224_appln_cpc. This table has 3 primary keys: appln_id, cpc_class_symbol and cpc_scheme. The information contained in the cpc_scheme regards who allocates the cpc_class_symbol to a specific appln_id (European Patent Office, United States Patent and Trademark Office or National Offices). This qualitative information is not relevant for the scope of the SETIS analyses, but if considered instead, it could bring duplicated records, since the same appln_id could have the same cpc_class_symbol and two different cpc_scheme. This justifies the use of the "SELECT DISTINCT" function.

Figure 3. Extraction query in PATSTAT, Appln_Id

```
SELECT DISTINCT
  tls201_appln.appln_id,
  tls201_appln.appln_auth,
  tls201_appln.appln_kind,
  tls201_appln.granted,
  YEAR (tls201_appln.earliest_filing_date) AS Year,
  YEAR (tls201_appln.appln_filing_date) AS Appl_Year,
  tls201_appln.docdb_family_id,
  tls207_pers_appln.person_id,
  MAX (b.invt_seq_nr) AS Num_Invt,
  MAX (b.applt_seq_nr) AS Num_Applt,
  tls224_appln_cpc.cpc_class_symbol
FROM
  tls201_appln
INNER JOIN
  tls207_pers_appln
ON tls207_pers_appln.appln_id = tls201_appln.appln_id
INNER JOIN
  tls207_pers_appln b
ON b.appln_id = tls201_appln.appln_id
INNER JOIN
  tls224_appln_cpc
ON tls224_appln_cpc.appln_id = tls201_appln.appln_id
INNER JOIN
  tls206_person
ON tls206_person.person_id = tls207_pers_appln.person_id
WHERE
  tls201_appln.appln_id < 900000000
AND
  tls201_appln.appln_kind IN ('A','W')
AND
  YEAR (tls201_appln.earliest_filing_date) BETWEEN 20XX AND 20XX
AND
  tls207_pers_appln.applt_seq_nr > 0
AND
  tls224_appln_cpc.cpc_class_symbol LIKE 'Y0%'
GROUP BY
  tls201_appln.appln_id,
  tls201_appln.appln_auth,
  tls201_appln.appln_kind,
  tls201_appln.granted,
  YEAR (tls201_appln.earliest_filing_date),
  YEAR (tls201_appln.appln_filing_date),
  tls201_appln.docdb_family_id,
  tls207_pers_appln.person_id,
  tls224_appln_cpc.cpc_class_symbol
ORDER BY
  tls207_pers_appln.person_id,
  YEAR (tls201_appln.earliest_filing_date),
  YEAR (tls201_appln.appln_filing_date),
  tls201_appln.docdb_family_id,
  tls224_appln_cpc.cpc_class_symbol;
```


2.2 2.Person_Id

This query (Figure 4) aims at extracting all information regarding applicants in the selected years that have filed at least one patent with CPC code in the "Y0" schema. doc_std_name_id (from DOCDB⁴) and psn_id (from EEE-PPAT⁵) are needed for the cleaning process described in section 3.

Note that PATSTAT assigns a new person_id to each applicant every time a new patent application is processed and when it is not possible to assign, in a systematic and automatic way, the same person_id that the same applicant had already received in the past. Therefore, there are cases in which two different person_id identify the same applicant. For these reasons, a different ID is associated to person_id in order to standardise (doc_std_name_id) or harmonise (psn_id) the person (applicant). Unfortunately, these two IDs are not equivalent and often they are not assigned correctly, therefore an additional effort is necessary in order to harmonise this information.

Figure 4. Extraction query in PATSTAT, Person_Id

```
SELECT
  tls206_person.person_id,
  tls206_person.doc_std_name_id,
  tls206_person.doc_std_name,
  tls206_person.psn_id,
  tls206_person.psn_name,
  tls206_person.nuts AS no_clean_nuts,
  tls206_person.person_ctype AS no_clean_person_ctype,
  tls206_person.psn_sector AS no_clean_sector
FROM
  tls206_person
WHERE
  tls206_person.person_id in
  (
    SELECT DISTINCT
      tls207_pers_appln.person_id
    FROM
      tls201_appln
    INNER JOIN
      tls207_pers_appln
        ON tls207_pers_appln.appln_id = tls201_appln.appln_id
    INNER JOIN
      tls224_appln_cpc
        ON tls224_appln_cpc.appln_id = tls201_appln.appln_id
    WHERE
      tls201_appln.appln_id < 900000000
    AND
      tls201_appln.appln_kind IN ('A','W')
    AND
      YEAR (tls201_appln.earliest_filing_date) BETWEEN 2000 AND 2016
    AND
      tls207_pers_appln.applt_seq_nr > 0
    AND
      tls224_appln_cpc.cpc_class_symbol LIKE 'Y0%'
    GROUP BY
      tls207_pers_appln.person_id
  )
ORDER BY
  tls206_person.person_id,
  tls206_person.doc_std_name_id,
  tls206_person.doc_std_name,
  tls206_person.psn_id,
  tls206_person.psn_name;
```

⁴ Database of the European Patent Office.

⁵ ECOOM-EUROSTAT-EPO PATSTAT Person Augmented Table (EEE-PPAT)

2.3 3.CPC

Each patent application is classified by means of CPC codes⁶. From PATSTAT it is possible to extract the complete list of these codes, under the "Y0" schema (Figure 5). CPC codes are structured in classes, sub-classes, groups and sub-groups. The example below clarifies these subdivisions:

— Y → Class

- Y02E → Sub-Class
 - Y02E 10 → Group
 - Y02E 10/1 → Sub-Group (1digit)
 - Y02E 10/12 → Sub-Group (2digits)
 - Y02E 10/125 → Sub-Group (3digits)

Once the CPC list is extracted from PASTAT, CPC codes are grouped in order to define the concordance between these code and technologies relevant in the context of SETIS. This exercise permits to identify, through the CPC codes, both Energy Union Priorities and SET-Plan Actions (European Commission, 2016)⁷.

Figure 5. Extraction query in PATSTAT, CPC

```
SELECT
  LEFT (tls224_appln_cpc.cpc_class_symbol,4) AS Subclass,
  LEFT (tls224_appln_cpc.cpc_class_symbol,8) AS Groups,
  LEFT (tls224_appln_cpc.cpc_class_symbol,10) AS Subgroup_1digit,
  LEFT (tls224_appln_cpc.cpc_class_symbol,11) AS Subgroup_2digits,
  tls224_appln_cpc.cpc_class_symbol AS CPC_CODE
FROM
  tls224_appln_cpc
WHERE
  tls224_appln_cpc.cpc_class_symbol LIKE 'Y0%'
GROUP BY
  LEFT (tls224_appln_cpc.cpc_class_symbol,4),
  LEFT (tls224_appln_cpc.cpc_class_symbol,8),
  LEFT (tls224_appln_cpc.cpc_class_symbol,10),
  LEFT (tls224_appln_cpc.cpc_class_symbol,11),
  tls224_appln_cpc.cpc_class_symbol
ORDER BY
  tls224_appln_cpc.cpc_class_symbol;
```

⁶ The Cooperative Patent Classification (CPC) is an international classification system for technical documents. For more details please refer to section 2.3 in Fiorini *et al.* (2017).

⁷ Annex I in Fiorini *et al.* (2017) provides the full list of CPC codes used for the technology concordance.

2.4 4.Country

The construction of the database is finalised by adding the country table, in which the list of all country is extracted from table `tls801_country` in PATSTAT (Figure 6). It has the following fields:

- `ctry_code`
- `iso_alpha3`
- `st3_name`
- `state_indicator`
- `continent`
- `eu_member`
- `epo_member`
- `oecd_member`
- `discontinued`

To these fields a new one ("Country") is added in order to have the identification country codes consistent with those used officially by the European Commission (EC) for the Member States (MS). For only two countries the code is modifies: Greece (from GR to EL) and United Kingdom (from GB to UK). In addition to this list of attributes, more information can be added to each country, as for example the macro classification of the 28 Member States countries (EU28), which facilitates aggregation of data and results.

Figure 6. Extraction query in PATSTAT, Country

```
SELECT *  
FROM  
    tls801_country;
```


2.5 6.Company_Family

This query focuses on doc_std_name_id and, for each of these IDs, it provides the list of patent families in which it has participated, regardless any CPC classification (Figure 7). These data are used to calculate the overall patenting activity of an applicant, as explained in sections 4.1.

Figure 7. Extraction query in PATSTAT, Company_Family

```
SELECT
    tls206_person.doc_std_name_id,
    tls201_appln.docdb_family_id,
    YEAR (tls201_appln.earliest_filing_date) AS Year
FROM
    tls201_appln
INNER JOIN
    tls207_pers_appln
    ON tls207_pers_appln.appln_id = tls201_appln.appln_id
INNER JOIN
    tls206_person
    ON tls206_person.person_id = tls207_pers_appln.person_id
WHERE
    tls201_appln.appln_id < 900000000
AND
    tls201_appln.appln_kind IN ('A','W')
AND
    YEAR (tls201_appln.earliest_filing_date) BETWEEN 20XX AND 20XX
AND
    tls207_pers_appln.applt_seq_nr > 0

GROUP BY
    tls206_person.doc_std_name_id,
    tls201_appln.docdb_family_id,
    YEAR (tls201_appln.earliest_filing_date)
ORDER BY
    tls206_person.doc_std_name_id,
    tls201_appln.docdb_family_id;
```


3 Data cleaning

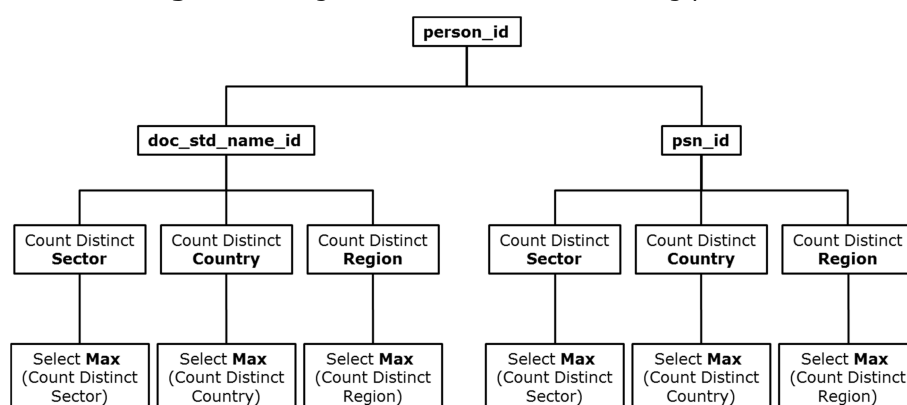
The data cleaning process is needed because of lack of completeness and accuracy in PATSTAT (European Patent Office, 2017). This lack is even more acute for information regarding name, sector, country and region of applicants. Therefore, it is necessary to revise the whole dataset in order to assign these attributes to those `person_id` that have not an information associate, or to correct those assignments that are wrong. Since the cleaning process is time consuming and occupies the majority of the data processing, it is designed in a way that changes generated in one version of the database are maintained in the following versions. Furthermore, since the only stable ID relative to applicants is the `person_id` across all PATSTAT editions, the cleaning procedure needs to be based on this key.

The data cleaning process follows two steps: the first consists of an automatic procedure and the second is a manual correction. The automatic procedure, in turn, requires three distinct stages. The first is executed through queries in PATSTAT (section 3.1), the second transfer information from the old version to the new one (section 3.2), and the third completes the cleaning procedure by an automatic assignment of information by means of an algorithm (section 3.3).

3.1 Step 1: automatic data cleaning (stage 1)

Data cleaning uses two IDs that are associated to each `person_id`: `doc_std_name_id` and `psn_id` (former `hrm_id`). These two IDs are used in order to capture as much information as possible related to each `person_id`. The objective of this stage is to catch the largest amount of information regarding name, sector, country and region related to each person (applicant). The methodology considered firstly individuates all `person_id` (applicant only) having at least a patent application in one of the CPC classification under consideration (Y-codes). Then, it looks at the other two associated identification codes (`doc_std_name_id` and `psn_id`). Subsequently, it counts how many times, in the whole PATSTAT dataset and without any constraint in terms of year, a specific sector, country and region is assigned to each of these two identification codes. Lastly, for every `person_id`, it chooses sector, country and region that is associated most frequently (see Figure 8). Every branch of the tree diagram requires a different query. These are run in PATSTAT and are reported in the following sub-sections.

Figure 8. Stage 1 of the automatic cleaning process



3.1.1 3.1.Sector_DOC_STD_NAME_ID

For those applicants with at least one patent filed in the "Y0" schema, this query counts how many times the associated doc_std_name_id is labelled with a specific sector classification in PATSTAT without restrictions on the years (Figure 9). BLANKS and UNKNOWN are not considered.

Figure 9. Extraction query in PATSTAT, Sector_DOC_STD_NAME_ID

```
SELECT
  p2.doc_std_name_id,
  p2.doc_std_name,
  p2.psn_sector AS sector_DOC_STD_NAME_ID,
  count (*) AS Count_Sector_DOC_STD_NAME_ID
FROM
  tls206_person p2
WHERE
  p2.doc_std_name_id IN
  (
    SELECT
      p1.doc_std_name_id
    FROM
      tls206_person p1
    WHERE
      p1.person_id IN
      (
        SELECT
          ap.person_id
        FROM
          tls201_appln a
        INNER JOIN
          tls207_pers_appln ap
            ON ap.appln_id = a.appln_id
        INNER JOIN
          tls224_appln_cpc c
            ON c.appln_id = a.appln_id
        WHERE
          a.appln_id < 900000000
        AND
          a.appln_kind IN ('A','W')
        AND
          ap.applt_seq_nr > 0
        AND
          c.cpc_class_symbol LIKE 'Y0%'
        GROUP BY
          ap.person_id
      )
    GROUP BY
      p1.doc_std_name_id
  )
AND
  p2.psn_sector NOT LIKE ''
AND
  p2.psn_sector NOT LIKE 'UNKNOWN'
GROUP BY
  p2.doc_std_name_id,
  p2.doc_std_name,
  p2.psn_sector
ORDER BY
  p2.doc_std_name_id,
  Count_Sector_DOC_STD_NAME_ID DESC,
  sector_DOC_STD_NAME_ID ASC;
```


3.1.2 3.2.Ctry_DOC_STD_NAME_ID

For those applicants with at least one patent filed in the "Y0" schema, this query counts how many times the associated doc_std_name_id is labelled with a specific country code in PATSTAT without restrictions on the years (Figure 10). BLANKS and UNKNOWN are not considered.

Figure 10. Extraction query in PATSTAT, Ctry_DOC_STD_NAME_ID

```
SELECT
  p2.doc_std_name_id,
  p2.doc_std_name,
  p2.person_ctype AS ctry_DOC_STD_NAME_ID,
  count (*) AS Count_ctry_DOC_STD_NAME_ID
FROM
  tls206_person p2
WHERE
  p2.doc_std_name_id IN
  (
    SELECT
      p1.doc_std_name_id
    FROM
      tls206_person p1
    WHERE
      p1.person_id IN
      (
        SELECT
          ap.person_id
        FROM
          tls201_appln a
        INNER JOIN
          tls207_pers_appln ap
            ON ap.appln_id = a.appln_id
        INNER JOIN
          tls224_appln_cpc c
            ON c.appln_id = a.appln_id
        WHERE
          a.appln_id < 900000000
        AND
          a.appln_kind IN ('A','W')
        AND
          ap.appln_seq_nr > 0
        AND
          c.cpc_class_symbol LIKE 'Y0%'
        GROUP BY
          ap.person_id
      )
    GROUP BY
      p1.doc_std_name_id
  )
AND
  p2.person_ctype NOT LIKE ''
AND
  p2.person_ctype NOT LIKE 'UNKNOWN'
GROUP BY
  p2.doc_std_name_id,
  p2.doc_std_name,
  p2.person_ctype
ORDER BY
  p2.doc_std_name_id,
  Count_ctry_DOC_STD_NAME_ID DESC,
  ctry_DOC_STD_NAME_ID ASC;
```


3.1.3 3.3.Sector_HRM_ID

For those applicants with at least one patent filed in the "Y0" schema, this query counts how many times the associated psn_id is labelled with a specific sector classification in PATSTAT without restrictions on the years (Figure 11). BLANKS and UNKNOWN are not considered.

Figure 11. Extraction query in PATSTAT, Sector_HRM_ID

```
SELECT
  p2.psn_id,
  p2.psn_name,
  p2.psn_sector AS sector_hrm_l2_id,
  count (*) AS Count_sector_hrm_l2_id
FROM
  tls206_person p2
WHERE
  p2.psn_id IN
  (
    SELECT
      p1.psn_id
    FROM
      tls206_person p1
    WHERE
      p1.person_id IN
      (
        SELECT
          ap.person_id
        FROM
          tls201_appln a
        INNER JOIN
          tls207_pers_appln ap
            ON ap.appln_id = a.appln_id
        INNER JOIN
          tls224_appln_cpc c
            ON c.appln_id = a.appln_id
        WHERE
          a.appln_id < 900000000
        AND
          a.appln_kind IN ('A','W')
        AND
          ap.applt_seq_nr > 0
        AND
          c.cpc_class_symbol LIKE 'Y0%'
        GROUP BY
          ap.person_id
      )
    GROUP BY
      p1.psn_id
  )
AND
  p2.psn_sector NOT LIKE ''
AND
  p2.psn_sector NOT LIKE 'UNKNOWN'
GROUP BY
  p2.psn_id,
  p2.psn_name,
  p2.psn_sector
ORDER BY
  p2.psn_id,
  Count_sector_hrm_l2_id DESC,
  sector_hrm_l2_id ASC;
```


3.1.4 3.4.Ctry_HRM_ID

For those applicants with at least one patent filed in the "Y0" schema, this query counts how many times the associated psn_id is labelled with a specific country code in PATSTAT without restrictions on the years (Figure 12). BLANKS and UNKNOWN are not considered.

Figure 12. Extraction query in PATSTAT, Ctry_HRM_ID

```
SELECT
  p2.psn_id,
  p2.psn_name,
  p2.person_ctype AS ctry_hrm_l2_id,
  count (*) AS Count_ctry_hrm_l2_id
FROM
  tls206_person p2
WHERE
  p2.psn_id IN
  (
    SELECT
      p1.psn_id
    FROM
      tls206_person p1
    WHERE
      p1.person_id IN
      (
        SELECT
          ap.person_id
        FROM
          tls201_appln a
        INNER JOIN
          tls207_pers_appln ap
            ON ap.appln_id = a.appln_id
        INNER JOIN
          tls224_appln_cpc c
            ON c.appln_id = a.appln_id
        WHERE
          a.appln_id < 900000000
        AND
          a.appln_kind IN ('A','W')
        AND
          ap.appln_seq_nr > 0
        AND
          c.cpc_class_symbol LIKE 'Y0%'
        GROUP BY
          ap.person_id
      )
    GROUP BY
      p1.psn_id
  )
AND
  p2.person_ctype NOT LIKE ''
AND
  p2.person_ctype NOT LIKE 'UNKNOWN'
GROUP BY
  p2.psn_id,
  p2.psn_name,
  p2.person_ctype
ORDER BY
  p2.psn_id,
  Count_ctry_hrm_l2_id DESC,
  ctry_hrm_l2_id ASC;
```


3.1.5 3.5.Nuts_DOC_STD_NAME_ID

For those applicants with at least one patent filed in the "Y0" schema, this query counts how many times the associated doc_std_name_id is labelled with a specific Nomenclature of Territorial Units for Statistics (NUTS) code⁸ in PATSTAT without restrictions on the years (Figure 13). NUTS level 0 and level 9 are not considered.

Figure 13. Extraction query in PATSTAT, Nuts_DOC_STD_NAME_ID

```
SELECT
  p2.doc_std_name_id,
  p2.doc_std_name,
  p2.nuts AS nuts_DOC_STD_NAME_ID,
  count (*) AS Count_nuts_DOC_STD_NAME_ID
FROM
  tls206_person p2
WHERE
  p2.doc_std_name_id IN
  (
    SELECT
      p1.doc_std_name_id
    FROM
      tls206_person p1
    WHERE
      p1.person_id IN
      (
        SELECT
          ap.person_id
        FROM
          tls201_appln a
        INNER JOIN
          tls207_pers_appln ap
            ON ap.appln_id = a.appln_id
        INNER JOIN
          tls224_appln_cpc c
            ON c.appln_id = a.appln_id
        WHERE
          a.appln_id < 900000000
        AND
          a.appln_kind IN ('A','W')
        AND
          ap.applt_seq_nr > 0
        AND
          c.cpc_class_symbol LIKE 'Y0%'
        GROUP BY
          ap.person_id
      )
    GROUP BY
      p1.doc_std_name_id
  )
AND
  p2.nuts_level BETWEEN 1 AND 8
GROUP BY
  p2.doc_std_name_id,
  p2.doc_std_name,
  p2.nuts
ORDER BY
  p2.doc_std_name_id,
  Count_nuts_DOC_STD_NAME_ID DESC,
  nuts_DOC_STD_NAME_ID ASC;
```

⁸ <http://ec.europa.eu/eurostat/web/nuts>

3.1.6 3.6.Nuts_HRM_ID

For those applicants with at least one patent filed in the "Y0" schema, this query counts how many times the associated psn_id is labelled with a specific NUTS code in PATSTAT without restrictions on the years (Figure 14). NUTS level 0 and level 9 are not considered.

Figure 14. Extraction query in PATSTAT, Nuts_HRM_ID

```
SELECT
  p2.psn_id,
  p2.psn_name,
  p2.nuts AS nuts_hrm_12_id,
  count (*) AS Count_nuts_hrm_12_id
FROM
  tls206_person p2
WHERE
  p2.psn_id IN
  (
    SELECT
      p1.psn_id
    FROM
      tls206_person p1
    WHERE
      p1.person_id IN
      (
        SELECT
          ap.person_id
        FROM
          tls201_appln a
        INNER JOIN
          tls207_pers_appln ap
            ON ap.appln_id = a.appln_id
        INNER JOIN
          tls224_appln_cpc c
            ON c.appln_id = a.appln_id
        WHERE
          a.appln_id < 900000000
        AND
          a.appln_kind IN ('A','W')
        AND
          ap.applt_seq_nr > 0
        AND
          c.cpc_class_symbol LIKE 'Y0%'
        GROUP BY
          ap.person_id
      )
    GROUP BY
      p1.psn_id
  )
AND
  p2.nuts_level BETWEEN 1 AND 8
GROUP BY
  p2.psn_id,
  p2.psn_name,
  p2.nuts
ORDER BY
  p2.psn_id,
  Count_nuts_hrm_12_id DESC,
  nuts_hrm_12_id ASC;
```


3.2 Step 1: automatic data cleaning (stage 2)

This stage consists on the transfer of information present in the old SETIS database to the new one. The European Patent Office (EPO) releases patent data through PATSTAT. Every six months a new version is provided with additional data and with an updated patent reclassification based on CPC codes. For this reason, a new SETIS database needs to be created, without losing information of the previous one. This section explains the process needed to transfer the cleaned attributed from the previous dataset to the new one, and it also explains how new information are added. This stage is run in Microsoft Access, and next sub-sections provide the relative queries.

3.2.1 c0_from_old_to_new

To, all person_id already present (and therefore with already cleaned attributes) in the old DB, this query permits to transfer information to the new one (Figure 15).

Figure 15. Query for cleaning, c0

```
SELECT
    [2_Person_Id_new].person_id,
    [1_Person_Id_Old].A_ID AS old_A_ID,
    [2_Person_Id_new].doc_std_name_id,
    [1_Person_Id_Old].doc_std_name AS old_doc_std_name,
    [1_Person_Id_Old].clean_sector AS old_sector,
    [1_Person_Id_Old].clean_person_ctry_code AS old_person_ctry_code,
    [1_Person_Id_Old].Reg_Code AS old_Reg_Code,
    [1_Person_Id_Old].Prov_Code AS old_Prov_Code
FROM
    1_Person_Id_Old
RIGHT JOIN 2_
    Person_Id_new
    ON [1_Person_Id_Old].person_id = [2_Person_Id_new].person_id
GROUP BY
    [2_Person_Id_new].person_id,
    [1_Person_Id_Old].A_ID,
    [2_Person_Id_new].doc_std_name_id,
    [1_Person_Id_Old].doc_std_name,
    [1_Person_Id_Old].clean_sector,
    [1_Person_Id_Old].clean_person_ctry_code,
    [1_Person_Id_Old].Reg_Code, [1_Person_Id_Old].Prov_Code
ORDER BY
    [2_Person_Id_new].doc_std_name_id;
```


3.2.2 c1_distinct_doc_std_name_id

This query groups all information for every doc_std_name_id in order to count how many new entries are present in the new dataset compared to the previous one for the same identification number (Figure 16).

Figure 16. Query for cleaning, c1

```
SELECT
  c0_from_old_to_new.old_A_ID,
  c0_from_old_to_new.doc_std_name_id,
  c0_from_old_to_new.old_doc_std_name,
  c0_from_old_to_new.old_sector,
  c0_from_old_to_new.old_person_ctry_code,
  c0_from_old_to_new.old_Reg_Code,
  c0_from_old_to_new.old_Prov_Code,
  Count(c0_from_old_to_new.doc_std_name_id) AS count_distinct
FROM
  c0_from_old_to_new
GROUP BY
  c0_from_old_to_new.old_A_ID,
  c0_from_old_to_new.doc_std_name_id,
  c0_from_old_to_new.old_doc_std_name,
  c0_from_old_to_new.old_sector,
  c0_from_old_to_new.old_person_ctry_code,
  c0_from_old_to_new.old_Reg_Code,
  c0_from_old_to_new.old_Prov_Code
ORDER BY
  c0_from_old_to_new.doc_std_name_id;
```

3.2.3 c2_doct_std_name_id_with_info

This query selects only those doc_std_name_id with information, meaning that these come from the old database (Figure 17).

Figure 17. Query for cleaning, c2

```
SELECT
  c1_distinct_doc_std_name_id.old_A_ID,
  c1_distinct_doc_std_name_id.doc_std_name_id,
  c1_distinct_doc_std_name_id.old_doc_std_name,
  c1_distinct_doc_std_name_id.old_sector,
  c1_distinct_doc_std_name_id.old_person_ctry_code,
  c1_distinct_doc_std_name_id.old_Reg_Code,
  c1_distinct_doc_std_name_id.old_Prov_Code,
  c1_distinct_doc_std_name_id.count_distinct
FROM
  c1_distinct_doc_std_name_id
GROUP BY
  c1_distinct_doc_std_name_id.old_A_ID,
  c1_distinct_doc_std_name_id.doc_std_name_id,
  c1_distinct_doc_std_name_id.old_doc_std_name,
  c1_distinct_doc_std_name_id.old_sector,
  c1_distinct_doc_std_name_id.old_person_ctry_code,
  c1_distinct_doc_std_name_id.old_Reg_Code,
  c1_distinct_doc_std_name_id.old_Prov_Code,
  c1_distinct_doc_std_name_id.count_distinct
HAVING
  (((c1_distinct_doc_std_name_id.old_A_ID) Not Like ""))
ORDER BY
  c1_distinct_doc_std_name_id.doc_std_name_id;
```


3.2.4 c3_count_doc_std_name_id

This query counts how many different doc_std_name_id are in the list coming from the previous query (Figure 18). This identifies if there are duplicates for the same ID. If count>1 it means that some manual corrections are needed. Instead, if count=1 it means that old information can be transfer directly to the new dataset as they are.

Figure 18. Query for cleaning, c3

```
SELECT
    c2_doct_std_name_id_with_info.doc_std_name_id,
    Count(c2_doct_std_name_id_with_info.doc_std_name_id) AS count_differences
FROM
    c2_doct_std_name_id_with_info
GROUP BY
    c2_doct_std_name_id_with_info.doc_std_name_id
ORDER BY
    c2_doct_std_name_id_with_info.doc_std_name_id;
```

3.2.5 c4_unique_doc_std_name_id

This query selects those doc_std_name_id with unique information that can be transferred directly to the new dataset (Figure 19).

Figure 19. Query for cleaning, c4

```
SELECT
    c2_doct_std_name_id_with_info.old_A_ID,
    c2_doct_std_name_id_with_info.doc_std_name_id,
    c2_doct_std_name_id_with_info.old_doc_std_name,
    c2_doct_std_name_id_with_info.old_sector,
    c2_doct_std_name_id_with_info.old_person_etry_code,
    c2_doct_std_name_id_with_info.old_Reg_Code,
    c2_doct_std_name_id_with_info.old_Prov_Code
FROM
    c3_count_doc_std_name_id
INNER JOIN
    c2_doct_std_name_id_with_info
    ON
    c3_count_doc_std_name_id.doc_std_name_id=c2_doct_std_name_id_with_info.doc_std_name_id
WHERE
    (((c3_count_doc_std_name_id.count_differences)=1))
GROUP BY
    c2_doct_std_name_id_with_info.old_A_ID,
    c2_doct_std_name_id_with_info.doc_std_name_id,
    c2_doct_std_name_id_with_info.old_doc_std_name,
    c2_doct_std_name_id_with_info.old_sector,
    c2_doct_std_name_id_with_info.old_person_etry_code,
    c2_doct_std_name_id_with_info.old_Reg_Code,
    c2_doct_std_name_id_with_info.old_Prov_Code
ORDER BY
    c2_doct_std_name_id_with_info.doc_std_name_id;
```


3.2.6 c5_unique_on_person_id

This query transfers old information to person_id into the new dataset (Figure 20).

Figure 20. Query for cleaning, c5

```
SELECT
  c4_unique_doc_std_name_id.old_A_ID,
  [2_Person_Id_new].person_id,
  [2_Person_Id_new].doc_std_name_id,
  [2_Person_Id_new].psn_id,
  c4_unique_doc_std_name_id.old_doc_std_name,
  c4_unique_doc_std_name_id.old_sector,
  c4_unique_doc_std_name_id.old_person_ctry_code,
  c4_unique_doc_std_name_id.old_Reg_Code,
  c4_unique_doc_std_name_id.old_Prov_Code
FROM
  2_Person_Id_new
INNER JOIN
  c4_unique_doc_std_name_id
    ON [2_Person_Id_new].doc_std_name_id = c4_unique_doc_std_name_id.doc_std_name_id
GROUP BY
  c4_unique_doc_std_name_id.old_A_ID,
  [2_Person_Id_new].person_id,
  [2_Person_Id_new].doc_std_name_id,
  [2_Person_Id_new].psn_id,
  c4_unique_doc_std_name_id.old_doc_std_name,
  c4_unique_doc_std_name_id.old_sector,
  c4_unique_doc_std_name_id.old_person_ctry_code,
  c4_unique_doc_std_name_id.old_Reg_Code,
  c4_unique_doc_std_name_id.old_Prov_Code
ORDER BY
  [2_Person_Id_new].doc_std_name_id;
```


3.2.7 c6_new_and_multiple_person_id

This query lists those person_id that are new into the new dataset and have multiple information (Figure 21). To these IDs, old information set is attached based on doc_std_name_id.

Figure 21. Query for cleaning, c6

```
SELECT
    c0_from_old_to_new.old_A_ID,
    c0_from_old_to_new.person_id,
    c0_from_old_to_new.doc_std_name_id,
    c0_from_old_to_new.psn_id,
    c0_from_old_to_new.old_doc_std_name,
    c0_from_old_to_new.old_sector,
    c0_from_old_to_new.old_person_ctype_code,
    c0_from_old_to_new.old_Reg_Code,
    c0_from_old_to_new.old_Prov_Code
FROM
    c0_from_old_to_new
LEFT JOIN
    c5_unique_on_person_id
    ON c0_from_old_to_new.person_id = c5_unique_on_person_id.person_id
WHERE
    (((c5_unique_on_person_id.doc_std_name_id) Is Null))
GROUP BY
    c0_from_old_to_new.old_A_ID,
    c0_from_old_to_new.person_id,
    c0_from_old_to_new.doc_std_name_id,
    c0_from_old_to_new.psn_id,
    c0_from_old_to_new.old_doc_std_name,
    c0_from_old_to_new.old_sector,
    c0_from_old_to_new.old_person_ctype_code,
    c0_from_old_to_new.old_Reg_Code,
    c0_from_old_to_new.old_Prov_Code
ORDER BY
    c0_from_old_to_new.doc_std_name_id;
```

3.2.8 c7_person_id_with_old

This query merges the previous two queries (c5 and c6), in order to get the final list of person_id with all information (Figure 22).

Figure 22. Query for cleaning, c7

```
SELECT
    c5_unique_on_person_id.*
FROM
    c5_unique_on_person_id

UNION

SELECT
    c6_new_and_multiple_person_id.*
FROM
    c6_new_and_multiple_person_id;
```


3.2.9 c8_final_to_clean

This query finalise the second stage of the automatic cleaning (Figure 23). It adds all information retrieved by the queries presented in section 3.1 (stage 1) to those elaborated in this section (stage 2).

Figure 23. Query for cleaning, c8

```
SELECT
  c7_person_id_with_old.old_A_ID,
  c7_person_id_with_old.person_id,
  c7_person_id_with_old.doc_std_name_id,
  c7_person_id_with_old.psn_id,
  c7_person_id_with_old.old_doc_std_name,
  [2_Person_Id_new].doc_std_name,
  [2_Person_Id_new].psn_name,
  c7_person_id_with_old.old_sector,
  [3_1_Sector_DOC_STD_NAME_ID].sector_DOC_STD_NAME_ID,
  [3_3_Sector_HRM_ID].sector_hrm_l2_id,
  c7_person_id_with_old.old_person_ctry_code,
  [3_2_Ctry_DOC_STD_NAME_ID].ctry_DOC_STD_NAME_ID,
  [3_4_Ctry_HRM_ID].ctry_hrm_l2_id,
  c7_person_id_with_old.old_Reg_Code,
  c7_person_id_with_old.old_Prov_Code,
  [3_5_Nuts_DOC_STD_NAME_ID].nuts_DOC_STD_NAME_ID,
  [3_6_Nuts_HRM_ID].nuts_hrm_l2_id
FROM
  3_6_Nuts_HRM_ID
RIGHT JOIN
  (3_5_Nuts_DOC_STD_NAME_ID
RIGHT JOIN
  (3_4_Ctry_HRM_ID
RIGHT JOIN
  (3_2_Ctry_DOC_STD_NAME_ID
RIGHT JOIN
  (3_3_Sector_HRM_ID
RIGHT JOIN
  (3_1_Sector_DOC_STD_NAME_ID
RIGHT JOIN
  (2_Person_Id_new
INNER JOIN
  c7_person_id_with_old
    ON [2_Person_Id_new].person_id = c7_person_id_with_old.person_id)
    ON [3_1_Sector_DOC_STD_NAME_ID].doc_std_name_id=c7_person_id_with_old.doc_std_name_id)
    ON [3_3_Sector_HRM_ID].psn_id = c7_person_id_with_old.psn_id)
    ON [3_2_Ctry_DOC_STD_NAME_ID].doc_std_name_id=c7_person_id_with_old.doc_std_name_id)
    ON [3_4_Ctry_HRM_ID].psn_id = c7_person_id_with_old.psn_id)
    ON [3_5_Nuts_DOC_STD_NAME_ID].doc_std_name_id=c7_person_id_with_old.doc_std_name_id)
    ON [3_6_Nuts_HRM_ID].psn_id = c7_person_id_with_old.psn_id
GROUP BY
  c7_person_id_with_old.old_A_ID,
  c7_person_id_with_old.person_id,
  c7_person_id_with_old.doc_std_name_id,
  c7_person_id_with_old.psn_id,
  c7_person_id_with_old.old_doc_std_name,
  [2_Person_Id_new].doc_std_name,
  [2_Person_Id_new].psn_name,
  c7_person_id_with_old.old_sector,
  [3_1_Sector_DOC_STD_NAME_ID].sector_DOC_STD_NAME_ID,
  [3_3_Sector_HRM_ID].sector_hrm_l2_id,
  c7_person_id_with_old.old_person_ctry_code,
  [3_2_Ctry_DOC_STD_NAME_ID].ctry_DOC_STD_NAME_ID,
  [3_4_Ctry_HRM_ID].ctry_hrm_l2_id,
  c7_person_id_with_old.old_Reg_Code,
  c7_person_id_with_old.old_Prov_Code,
  [3_5_Nuts_DOC_STD_NAME_ID].nuts_DOC_STD_NAME_ID,
  [3_6_Nuts_HRM_ID].nuts_hrm_l2_id
ORDER BY
  c7_person_id_with_old.doc_std_name_id;
```


3.3 Step 1: automatic data cleaning (stage 3)

The first step of the automatic cleaning terminates with stage 3. It is needed in order to make a correct assignment of name, sector, country and regional code to all person_id. Every person_id have more than one entry for every type of attribute, since they can come from the old database and from the two identification codes used in stage 1. The assignment follows the algorithms in figures below.

Figure 24. Name allocation

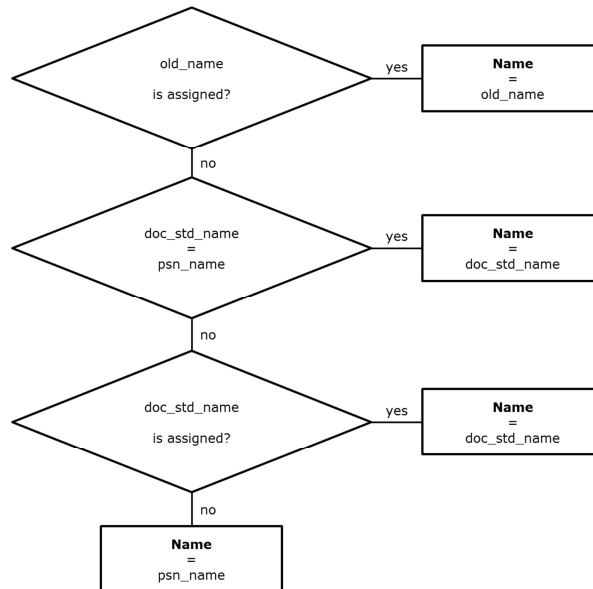


Figure 25. Sector allocation

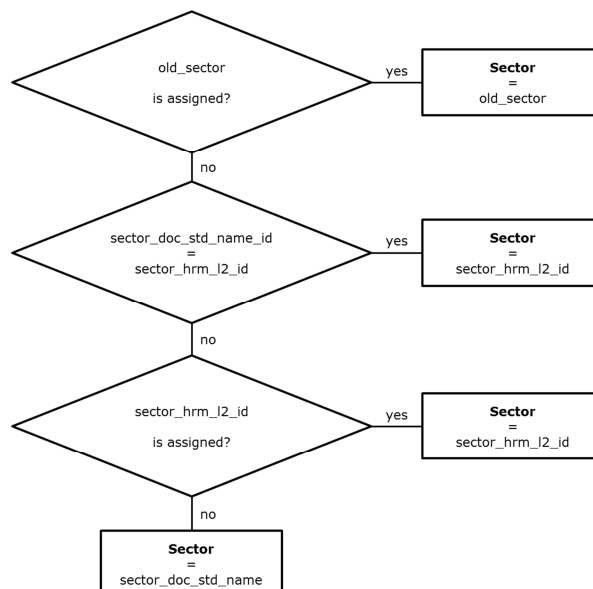


Figure 26. Country allocation

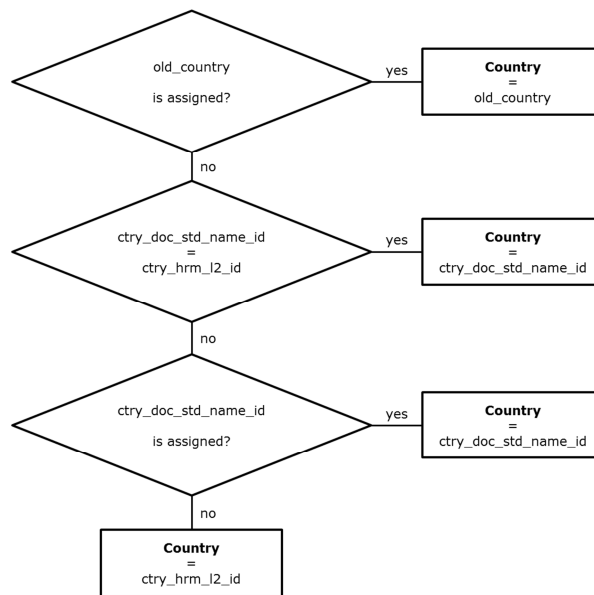
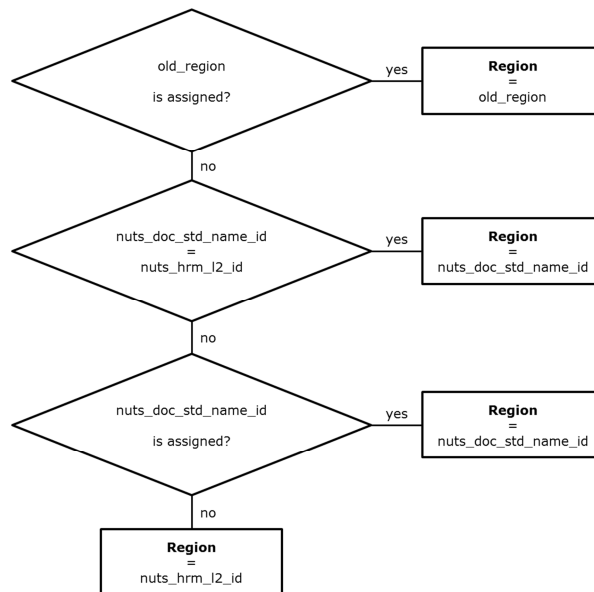


Figure 27. Region allocation



The assignment of sector to person_id reflects often that one attributed to the psn_id (Figure 25) because this information is provided by EEE-PPAT that harmonises it in relation to this ID. Therefore it is considered to be more accurate than that one coming from doc_std_name_id. On the contrary, name, country and region are assigned mainly by referring to the information coming from doc_std_name_id, because it brings more detailed information in the name, such as the companies' legal status, which simplifies and makes faster the next step of manual correction.

3.4 Step 2: manual data cleaning

The output of the automatic data cleaning process is the full list of all applicants with information regarding their name, sector, country and region. However, the first step does not guarantee that all these attributes are assigned to all or that are assigned correctly. For example, it might happen that a `person_id` was not present in the old SETIS DB, being therefore is never cleaned before. Further, the same ID might have not information at all, and these cannot be obtained via the previous step 1. Hence, there might be cases in which empty values are present for some of the required attributes. These issues motivate the manual correction which occurs after the automatic data cleaning process.

The manual correction consists of assigning name, sector, country and region to those `person_id` that have not information assigned automatically via the previous step. The first focus is on the country: online searches are used in order to find the missing information for those `person_id` with country not assigned. These searches are based on names associated to `person_id` (`doc_std_name` and `psn_name`) and help also to retrieve other attribute or verify those already present (as for example regional code and/or sector).

The manual cleaning then focuses on the names and aims at their harmonisation. As already explained, a `person_id` does not identify a distinct entity, but it is created and attached to applicants when they are not recognised to be the same entity already present in PATSTAT. Therefore, two or more `person_id` can be associated to the same entity. Hence, name harmonisation is required in order to establish a relation between those `person_id` identifying the same entity. This operation is done by using an open source desktop tool, named OpenRefine⁹. However, additional manual name harmonisation might be necessary if inconsistencies are noticed.

Based on the harmonised names, a new ID is created. It is called `JRC_ID` and identifies all distinct entities in the list, even if they have more than one identification codes in PATSTAT. This new ID represents the unit of analysis of the following sections regarding patent statistics and R&D estimations. The `JRC_ID` is also relevant in the SETIS DB allowing the establishment of connections with data from other sources, as described in section 5. Moreover, since the manual cleaning can be run at any time, the `JRC_ID` is not stable but it is adjusted automatically every time corrections are made.

In conclusion, the step 2 of the cleaning data process focuses mainly on data harmonisation. It aims at providing a completed and detailed set of data regarding applicants. Since improvements and data refinement can be always required, the structure of the SETIS DB allows to automatically implementing these change by creating an adaptable `JRC_ID`, which is the unit of analysis of the following two sections.

⁹ <http://openrefine.org/>

4 Patent statistics

This section explains the procedural steps required to construct indicators related to patents. These steps are run in Microsoft Access and permit to automatically calculate patents statistics. Although some query could have been combined in one, allowing for lower computation, these are run separately in order to facilitate check of intermediate results between one step and the other. The following sub-sections illustrate how different patent statistics are calculated through the relative queries: total fractional for all applicants, regardless technological classifications (section 4.1), granted patent families (section 4.2), international patent families (section 4.3) and final patent statistics (section 4.4).

4.1 Total fractional

This section explains the computation steps needed to calculate the total fractional for every applicant (or JRC_ID) regardless the technological classification. As a starting point, it uses the extraction query run in PATSTAT and already presented in section 2.5.

4.1.1 q1_List_Fam_Company

This query associates the JRC_ID to the doc_std_name_id (Figure 28).

Figure 28. Query for total fractional, q1

```
SELECT
    Family_Company.docdb_family_id,
    IIf(Person_Id.JRC_ID>900000000,Person_Id.JRC_ID,[Family_Company].doc_std_name_id) AS Doc_JRC_Id
FROM
    Family_Company
LEFT JOIN
    Person_Id
    ON Family_Company.doc_std_name_id = Person_Id.doc_std_name_id
GROUP BY
    Family_Company.docdb_family_id,
    IIf(Person_Id.JRC_ID>900000000,Person_Id.JRC_ID,[Family_Company].doc_std_name_id)
ORDER BY
    Family_Company.docdb_family_id;
```

4.1.2 q2_Count_Fam_Company

This query counts the total number of participants in every family, which is the denominator of the fractional count (Figure 29).

Figure 29. Query for total fractional, q2

```
SELECT
    q1_List_Fam_Company.docdb_family_id,
    Count(q1_List_Fam_Company.Doc_JRC_Id) AS CountOfDoc_JRC_Id
FROM
    q1_List_Fam_Company
GROUP BY
    q1_List_Fam_Company.docdb_family_id;
```


4.1.3 q3_Fractional

This query computes the fractional for each participant in every family (Figure 30).

Figure 30. Query for total fractional, q3

```
SELECT
    q1_List_Fam_Company.docdb_family_id,
    q1_List_Fam_Company.Doc_JRC_Id,
    1/q2_Count_Fam_Company.CountOfDoc_JRC_Id AS Fractional
FROM
    q1_List_Fam_Company
INNER JOIN
    q2_Count_Fam_Company
    ON q1_List_Fam_Company.docdb_family_id = q2_Count_Fam_Company.docdb_family_id
GROUP BY
    q1_List_Fam_Company.docdb_family_id,
    q1_List_Fam_Company.Doc_JRC_Id,
    1/q2_Count_Fam_Company.CountOfDoc_JRC_Id;
```

4.1.4 q4_Total_Fractional_Company

This query computes the total fractional for all applicants in the list, regardless technological classifications (Figure 31).

Figure 31. Query for total fractional, q4

```
SELECT
    q3_Fractional.Doc_JRC_Id,
    Sum(q3_Fractional.Fractional) AS Tot_Fractional
FROM
    q3_Fractional
WHERE
    (q3_Fractional.Doc_JRC_Id)>900000000
GROUP BY
    q3_Fractional.Doc_JRC_Id;
```


4.2 Granted patent families

This section explains the computation steps needed to calculate granted patent families.

4.2.1 g1_List_Fam_Appln

This query lists all patent applications in every families and says if these are granted (1) or not (0), (Figure 32).

Figure 32. Granted patent families, g1

```
SELECT
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Appln_Id.appln_id,
    Appln_Id.granted
FROM
    Appln_Id
GROUP BY
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Appln_Id.appln_id,
    Appln_Id.granted
ORDER BY
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Appln_Id.appln_id;
```

4.2.2 g2_List_Fam_Granted

This query calculates the share of granted applications in every family (Figure 33).

Figure 33. Granted patent families, g2

```
SELECT
    g1_List_Fam_Appln.docdb_family_id,
    g1_List_Fam_Appln.Year,
    Count(g1_List_Fam_Appln.appln_id) AS Tot_appln,
    Sum(g1_List_Fam_Appln.granted) AS Tot_granted,
    Sum(g1_List_Fam_Appln.granted)/Count(g1_List_Fam_Appln.appln_id) AS Share_granted
FROM
    g1_List_Fam_Appln
GROUP BY
    g1_List_Fam_Appln.docdb_family_id,
    g1_List_Fam_Appln.Year;
```


4.3 International patent families

This section explains the computation steps needed to calculate international patent families.

4.3.1 i1_List_Fam_Auth

This query lists all the applications authorities in every family (Figure 34).

Figure 34. International patent families, i1

```
SELECT
  Appln_Id.docdb_family_id,
  Appln_Id.Year,
  Appln_Id.appln_auth,
  IIF(Appln_Id.appln_auth="EP", 1, 0) AS EP,
  IIF(Appln_Id.appln_auth="US", 1, 0) AS US,
  IIF(Appln_Id.appln_auth="CN", 1, 0) AS CN,
  IIF(Appln_Id.appln_auth="JP", 1, 0) AS JP,
  IIF(Appln_Id.appln_auth="KR", 1, 0) AS KR,
  IIF(Appln_Id.appln_auth="AL"
    or Appln_Id.appln_auth="AT"
    or Appln_Id.appln_auth="BE"
    or Appln_Id.appln_auth="BG"
    or Appln_Id.appln_auth="CH"
    or Appln_Id.appln_auth="CY"
    or Appln_Id.appln_auth="CZ"
    or Appln_Id.appln_auth="DE"
    or Appln_Id.appln_auth="DK"
    or Appln_Id.appln_auth="EE"
    or Appln_Id.appln_auth="ES"
    or Appln_Id.appln_auth="FI"
    or Appln_Id.appln_auth="FR"
    or Appln_Id.appln_auth="GB"
    or Appln_Id.appln_auth="GR"
    or Appln_Id.appln_auth="HR"
    or Appln_Id.appln_auth="HU"
    or Appln_Id.appln_auth="IE"
    or Appln_Id.appln_auth="IS"
    or Appln_Id.appln_auth="IT"
    or Appln_Id.appln_auth="LI"
    or Appln_Id.appln_auth="LT"
    or Appln_Id.appln_auth="LU"
    or Appln_Id.appln_auth="LV"
    or Appln_Id.appln_auth="MC"
    or Appln_Id.appln_auth="MK"
    or Appln_Id.appln_auth="MT"
    or Appln_Id.appln_auth="NL"
    or Appln_Id.appln_auth="NO"
    or Appln_Id.appln_auth="PL"
    or Appln_Id.appln_auth="PT"
    or Appln_Id.appln_auth="RO"
    or Appln_Id.appln_auth="RS"
    or Appln_Id.appln_auth="SE"
    or Appln_Id.appln_auth="SI"
    or Appln_Id.appln_auth="SK"
    or Appln_Id.appln_auth="SM"
    or Appln_Id.appln_auth="TR", 1, 0) AS EPO_Ctry
FROM
  Appln_Id
GROUP BY
  Appln_Id.docdb_family_id,
  Appln_Id.Year,
  Appln_Id.appln_auth
ORDER BY
  Appln_Id.docdb_family_id,
  Appln_Id.Year;
```


4.3.2 i2_List_Fam_Int

This query determines whether a family is international or not. It is international when there are at least two different patent offices in the same family (Figure 35).

Figure 35. International patent families, i2

```
SELECT
    i1_List_Fam_Auth.docdb_family_id,
    i1_List_Fam_Auth.Year,
    Count(i1_List_Fam_Auth.appln_auth) AS Tot_auth,
    IIf(Count([i1_List_Fam_Auth].[appln_auth])>1,1,0) AS International,
    Sum(i1_List_Fam_Auth.EP)/Count(i1_List_Fam_Auth.appln_auth) AS EPO,
    Sum(i1_List_Fam_Auth.US)/Count(i1_List_Fam_Auth.appln_auth) AS USPTO,
    Sum(i1_List_Fam_Auth.CN)/Count(i1_List_Fam_Auth.appln_auth) AS SIPO,
    Sum(i1_List_Fam_Auth.JP)/Count(i1_List_Fam_Auth.appln_auth) AS JPO,
    Sum(i1_List_Fam_Auth.KR)/Count(i1_List_Fam_Auth.appln_auth) AS KIPO,
    Sum(i1_List_Fam_Auth.EPO_Ctry)/Count(i1_List_Fam_Auth.appln_auth) AS EPO_Ctry,
    (Count(i1_List_Fam_Auth.appln_auth)-
        (Sum(i1_List_Fam_Auth.EP)+Sum(i1_List_Fam_Auth.US)+Sum(i1_List_Fam_Auth.CN)+Sum(i1_List_Fam_Auth.JP)+Sum(i1_List_Fam_Auth.KR)+Sum(i1_List_Fam_Auth.EPO_Ctry)))/Count(i1_List_Fam_Auth.appln_auth) AS Other_Auth
FROM
    i1_List_Fam_Auth
GROUP BY
    i1_List_Fam_Auth.docdb_family_id,
    i1_List_Fam_Auth.Year
ORDER BY
    i1_List_Fam_Auth.docdb_family_id,
    i1_List_Fam_Auth.Year;
```


4.4 Fractional and final statistics

This section explains the computational steps needed to calculate energy fractional for every applicant (or JRC_ID). It also provides procedural steps needed to combine all together patent statistics.

4.4.1 List of applicant: Company_List

This query creates the full list of unique JRC_ID, with the related attributes: name, sector, country, region, etc (Figure 36).

Figure 36. Company list

```
SELECT
    Person_Id.A_ID,
    Person_Id.JRC_Id,
    Person_Id.doc_std_name,
    Person_Id.clean_sector,
    Countries.Country,
    Person_Id.Reg_Code,
    Person_Id.Prov_Code,
    Countries.EU28,
    Countries.EEA,
    Countries.world_player,
    Countries.world_player_code
FROM
    Countries
RIGHT JOIN
    Person_Id
    ON Countries.ctrtry_code = Person_Id.clean_person_ctrtry_code
GROUP BY
    Person_Id.A_ID,
    Person_Id.JRC_Id,
    Person_Id.doc_std_name,
    Person_Id.clean_sector,
    Countries.Country,
    Person_Id.Reg_Code,
    Person_Id.Prov_Code,
    Countries.EU28,
    Countries.EEA,
    Countries.world_player, Countries.world_player_code
ORDER BY
    Person_Id.doc_std_name,
    Countries.Country;
```

4.4.2 o1_List_Fam_CPC

This query creates a list of all families and it also associates all the CPC codes present in every family (Figure 37).

Figure 37. Patent families, o1

```
SELECT
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Appln_Id.cpc_class_symbol
FROM
    Appln_Id
GROUP BY
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Appln_Id.cpc_class_symbol
ORDER BY
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Appln_Id.cpc_class_symbol;
```


4.4.3 o2_List_Fam_Company

This query creates a list of all families and it also associates all participants (JRC_ID) in every family (Figure 38).

Figure 38. Patent families, o2

```
SELECT
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Person_Id.JRC_Id
FROM
    Person_Id
INNER JOIN
    Appln_Id
    ON Person_Id.person_id = Appln_Id.person_id
GROUP BY
    Appln_Id.docdb_family_id,
    Appln_Id.Year,
    Person_Id.JRC_Id
ORDER BY
    Appln_Id.docdb_family_id,
    Appln_Id.Year;
```

4.4.4 o3_List_Fam_CPC_Company

This query combines the previous two queries (o1 and o2) and creates a list of all combination of CPC codes and participants in every family (Figure 39).

Figure 39. Patent families, o3

```
SELECT
    o1_List_Fam_CPC.docdb_family_id,
    o1_List_Fam_CPC.Year,
    o1_List_Fam_CPC.cpc_class_symbol,
    o2_List_Fam_Company.JRC_Id
FROM
    o1_List_Fam_CPC
INNER JOIN
    o2_List_Fam_Company
    ON (o1_List_Fam_CPC.docdb_family_id = o2_List_Fam_Company.docdb_family_id)
    AND (o1_List_Fam_CPC.Year = o2_List_Fam_Company.Year)
GROUP BY
    o1_List_Fam_CPC.docdb_family_id,
    o1_List_Fam_CPC.Year,
    o1_List_Fam_CPC.cpc_class_symbol,
    o2_List_Fam_Company.JRC_Id
ORDER BY
    o1_List_Fam_CPC.docdb_family_id,
    o1_List_Fam_CPC.Year,
    o1_List_Fam_CPC.cpc_class_symbol,
    o2_List_Fam_Company.JRC_Id;
```


4.4.5 o4_Count_Fam_CPC_Company

This query counts the number of combination in every family (Figure 40). This number is the denominator of the fractional count.

Figure 40. Patent families, o4

```
SELECT
    o3_List_Fam_CPC_Company.docdb_family_id,
    o3_List_Fam_CPC_Company.Year,
    Count(o3_List_Fam_CPC_Company.docdb_family_id) AS CountOfdocdb_family_id
FROM
    o3_List_Fam_CPC_Company
GROUP BY
    o3_List_Fam_CPC_Company.docdb_family_id,
    o3_List_Fam_CPC_Company.Year
ORDER BY
    o3_List_Fam_CPC_Company.docdb_family_id;
```

4.4.6 o5_Fractional

This query provides the fractional count for all combination of CPC codes and participants in every family (Figure 41).

Figure 41. Patent families, o5

```
SELECT
    o3_List_Fam_CPC_Company.docdb_family_id,
    o3_List_Fam_CPC_Company.Year,
    o3_List_Fam_CPC_Company.cpc_class_symbol,
    o3_List_Fam_CPC_Company.JRC_Id,
    1/o4_Count_Fam_CPC_Company.CountOfdocdb_family_id AS Fractional,
    [g2_List_Fam_Granted].[Share_granted]/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS
    Fractional_Granted,
    i2_List_Fam_Int.[International]/o4_Count_Fam_CPC_Company.CountOfdocdb_family_id AS
    Fractional_International,
    i2_List_Fam_Int.EPO/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS EPO,
    i2_List_Fam_Int.USPTO/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS USPTO,
    i2_List_Fam_Int.SIPO/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS SIPO,
    i2_List_Fam_Int.JPO/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS JPO,
    i2_List_Fam_Int.KIPO/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS KIPO,
    i2_List_Fam_Int.EPO_Ctry/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS EPO_Ctry,
    i2_List_Fam_Int.Other_Auth/[o4_Count_Fam_CPC_Company].[CountOfdocdb_family_id] AS Other_Auth
FROM
    i2_List_Fam_Int
INNER JOIN
    (g2_List_Fam_Granted
INNER JOIN
    (o3_List_Fam_CPC_Company
INNER JOIN
    o4_Count_Fam_CPC_Company
        ON (o3_List_Fam_CPC_Company.Year = o4_Count_Fam_CPC_Company.Year)
        AND (o3_List_Fam_CPC_Company.docdb_family_id=o4_Count_Fam_CPC_Company.docdb_family_id))
        ON (g2_List_Fam_Granted.Year = o3_List_Fam_CPC_Company.Year)
        AND (g2_List_Fam_Granted.docdb_family_id = o3_List_Fam_CPC_Company.docdb_family_id)
        AND (g2_List_Fam_Granted.Year = o4_Count_Fam_CPC_Company.Year)
        AND (g2_List_Fam_Granted.docdb_family_id = o4_Count_Fam_CPC_Company.docdb_family_id))
        ON (i2_List_Fam_Int.docdb_family_id = o4_Count_Fam_CPC_Company.docdb_family_id)
        AND (i2_List_Fam_Int.Year = o4_Count_Fam_CPC_Company.Year)
        AND (i2_List_Fam_Int.docdb_family_id = o3_List_Fam_CPC_Company.docdb_family_id)
        AND (i2_List_Fam_Int.Year = o3_List_Fam_CPC_Company.Year)
ORDER BY
    o3_List_Fam_CPC_Company.docdb_family_id,
    o3_List_Fam_CPC_Company.Year,
    o3_List_Fam_CPC_Company.cpc_class_symbol,
    o3_List_Fam_CPC_Company.JRC_Id,
    1/o4_Count_Fam_CPC_Company.CountOfdocdb_family_id;
```


4.4.7 o6_Total

This query permits to combine in one table all patent statistics calculated previously (Figure 42).

Figure 42. Patent families, o6

```
SELECT
    Company_List.A_ID,
    o5_Fractional.JRC_Id,
    Company_List.doc_std_name AS Name,
    Company_List.clean_sector AS Sector,
    Company_List.Country,
    Company_List.Reg_Code,
    Company_List.Prov_Code,
    Company_List.EU28,
    Company_List.EEA,
    Company_List.world_player_code AS World_player,
    o5_Fractional.Year,
    q4_Total_Fractional_Company.Tot_Fractional,
    CPC_List_All.Energy_Union_Priority,
    CPC_List_All.Actions,
    CPC_List_All.Technology,
    o5_Fractional.cpc_class_symbol,
    Sum(o5_Fractional.Fractional) AS CPC_Fractional,
    Sum(o5_Fractional.Fractional_Granted) AS Tot_Fractional_Granted,
    Sum(o5_Fractional.Fractional_International) AS Tot_Fractional_International,
    Sum(o5_Fractional.EPO) AS EPO,
    Sum(o5_Fractional.USPTO) AS USPTO,
    Sum(o5_Fractional.SIPO) AS SIPO,
    Sum(o5_Fractional.JPO) AS JPO,
    Sum(o5_Fractional.KIPO) AS KIPO,
    Sum(o5_Fractional.EPO_Ctry) AS EPO_Ctry,
    Sum(o5_Fractional.Other_Auth) AS Other_Auth
FROM
    CPC_List_All
RIGHT JOIN
    ((o5_Fractional
LEFT JOIN
    Company_List
        ON o5_Fractional.JRC_Id = Company_List.JRC_Id)
LEFT JOIN
    q4_Total_Fractional_Company
        ON o5_Fractional.JRC_Id = q4_Total_Fractional_Company.Doc_JRC_Id
    ON CPC_List_All.CPC_CODE = o5_Fractional.cpc_class_symbol
GROUP BY
    Company_List.A_ID,
    o5_Fractional.JRC_Id,
    Company_List.doc_std_name,
    Company_List.clean_sector,
    Company_List.Country,
    Company_List.Reg_Code,
    Company_List.Prov_Code,
    Company_List.EU28,
    Company_List.EEA,
    Company_List.world_player_code,
    o5_Fractional.Year,
    q4_Total_Fractional_Company.Tot_Fractional,
    CPC_List_All.Energy_Union_Priority,
    CPC_List_All.Actions,
    CPC_List_All.Technology,
    o5_Fractional.cpc_class_symbol;
```


4.4.8 Patent Year

This query aggregates data of the previous query at technology level, defined via the concordance between CPC codes and technology classification (Figure 43).

Figure 43. Patent families, Year

```
SELECT
    o6_Total.A_ID,
    o6_Total.JRC_Id,
    o6_Total.Name,
    o6_Total.Sector,
    o6_Total.Country,
    o6_Total.Reg_Code,
    o6_Total.Prov_Code,
    o6_Total.EU28,
    o6_Total.EEA,
    o6_Total.World_player,
    o6_Total.Year,
    o6_Total.Tot_Fractional,
    o6_Total.Energy_Union_Priority,
    o6_Total.Actions,
    o6_Total.Technology,
    Sum(o6_Total.CPC_Fractional) AS Energy_Tech_Fractional,
    Sum(o6_Total.Count_Family) AS Tot_Families,
    Sum(o6_Total.Tot_Fractional_Granted) AS Granted,
    Sum(o6_Total.Tot_Fractional_International) AS International,
    Sum(o6_Total.Tot_Fractional_EP_US_CN) AS in_EP_US_CN
FROM
    o6_Total
GROUP BY
    o6_Total.A_ID,
    o6_Total.JRC_Id,
    o6_Total.Name,
    o6_Total.Sector,
    o6_Total.Country,
    o6_Total.Reg_Code,
    o6_Total.Prov_Code,
    o6_Total.EU28,
    o6_Total.EEA,
    o6_Total.World_player,
    o6_Total.Year,
    o6_Total.Tot_Fractional,
    o6_Total.Energy_Union_Priority,
    o6_Total.Actions,
    o6_Total.Technology
ORDER BY
    o6_Total.Name;
```


4.5 Patent projections

This section presents queries needed to calculate patent projections for years following the latest one, for which the dataset is complete (in this example, 2013 is the latest complete year).

4.5.1 Patent_2010

This query sums patent fractional at country, aggregated via the "world player" classification, and technology level for 2010 (Figure 44).

Figure 44. Patent projection, 2010

```
SELECT
  [2010].World_player,
  [2010].Energy_Union_Priority,
  [2010].Actions,
  [2010].Technology,
  Sum([2010].Energy_Tech_Fractional) AS 2010,
  [2010].Year
FROM
  2010
WHERE
  ((([2010].Sector)<>'INDIVIDUAL'))
GROUP BY
  [2010].World_player,
  [2010].Energy_Union_Priority,
  [2010].Actions,
  [2010].Technology,
  [2010].Year;
```

4.5.2 Patent_2011

This query sums patent fractional at country, aggregated via the "world player" classification, and technology level for 2011 (Figure 45).

Figure 45. Patent projection, 2011

```
SELECT
  [2011].World_player,
  [2011].Energy_Union_Priority,
  [2011].Actions,
  [2011].Technology,
  Sum([2011].Energy_Tech_Fractional) AS 2011,
  [2011].Year
FROM
  2011
WHERE
  ((([2011].Sector)<>'INDIVIDUAL'))
GROUP BY
  [2011].World_player,
  [2011].Energy_Union_Priority,
  [2011].Actions,
  [2011].Technology,
  [2011].Year;
```


4.5.3 Patent_2012

This query sums patent fractional at country, aggregated via the "world player" classification, and technology level for 2012 (Figure 46).

Figure 46. Patent projection, 2012

```
SELECT
    [2012].World_player,
    [2012].Energy_Union_Priority,
    [2012].Actions,
    [2012].Technology,
    Sum([2012].Energy_Tech_Fractional) AS 2012,
    [2012].Year
FROM
    2012
WHERE
    ((([2012].Sector)<>'INDIVIDUAL'))
GROUP BY
    [2012].World_player,
    [2012].Energy_Union_Priority,
    [2012].Actions,
    [2012].Technology,
    [2012].Year;
```

4.5.4 Patent_2013

This query sums patent fractional at country, aggregated via the "world player" classification, and technology level for 2013 (Figure 47).

Figure 47. Patent projection, 2013

```
SELECT
    [2013].World_player,
    [2013].Energy_Union_Priority,
    [2013].Actions,
    [2013].Technology,
    Sum([2013].Energy_Tech_Fractional) AS 2013,
    [2013].Year
FROM
    2013
WHERE
    ((([2013].Sector)<>'INDIVIDUAL'))
GROUP BY
    [2013].World_player,
    [2013].Energy_Union_Priority,
    [2013].Actions,
    [2013].Technology,
    [2013].Year;
```


4.5.5 Avg

This query calculates first the change rates between two consecutive years (2010-2011, 2011-2012 and 2012-2013) and then calculates the average (Figure 48).

Figure 48. Patent projection, Average

```
SELECT
    Patent_2010.World_player,
    Patent_2010.Energy_Union_Priority,
    Patent_2010.Actions,
    Patent_2010.Technology,
    ([Patent_2011].[2011]-[Patent_2010].[2010])/[Patent_2010].[2010] AS 201011,
    ([Patent_2012].[2012]-[Patent_2011].[2011])/[Patent_2011].[2011] AS 201112,
    ([Patent_2013].[2013]-[Patent_2012].[2012])/[Patent_2012].[2012] AS 201213,
    ((([Patent_2011].[2011]-[Patent_2010].[2010])/[Patent_2010].[2010]+([Patent_2012].[2012]-
    [Patent_2011].[2011])/[Patent_2011].[2011]+([Patent_2013].[2013]-
    [Patent_2012].[2012])/[Patent_2012].[2012])/3 AS [Avg]
FROM
    ((Patent_2010
INNER JOIN
    Patent_2011
        ON (Patent_2010.World_player = Patent_2011.World_player)
        AND (Patent_2010.Energy_Union_Priority = Patent_2011.Energy_Union_Priority)
        AND (Patent_2010.Actions = Patent_2011.Actions)
        AND (Patent_2010.Technology = Patent_2011.Technology))
INNER JOIN
    Patent_2012
        ON (Patent_2011.World_player = Patent_2012.World_player)
        AND (Patent_2011.Energy_Union_Priority = Patent_2012.Energy_Union_Priority)
        AND (Patent_2011.Actions = Patent_2012.Actions)
        AND (Patent_2011.Technology = Patent_2012.Technology))
INNER JOIN
    Patent_2013
        ON (Patent_2012.World_player = Patent_2013.World_player)
        AND (Patent_2012.Energy_Union_Priority = Patent_2013.Energy_Union_Priority)
        AND (Patent_2012.Actions = Patent_2013.Actions)
        AND (Patent_2012.Technology = Patent_2013.Technology));
```


4.5.6 2014e

This query estimates patent for 2014 (Figure 49).

Figure 49. Patent projection, 2014e

```
SELECT
  Null AS A_ID,
  Null AS JRC_Id,
  Null AS Name,
  Null AS Sector,
  [2013].Country,
  Null AS Reg_Code,
  Null AS Prov_Code,
  [2013].EU28,
  [2013].EEA,
  [2013].World_player,
  "2014(e)" AS [Year],
  Null AS Tot_Fractional,
  [2013].Energy_Union_Priority,
  [2013].Actions,
  [2013].Technology,
  Sum([2013].Energy_Tech_Fractional)*(1+Avg([Avg].Avg)) AS Energy_Tech_Fractional,
  Null AS Granted,
  Null AS International,
  Null AS EPO,
  Null AS USPTO,
  Null AS SIPO,
  Null AS JPO,
  Null AS KIPO,
  Null AS EPO_Ctry,
  Null AS Other_Auth
FROM
  [Avg]
INNER JOIN
  2013
  ON (Avg.Technology = [2013].Technology)
  AND (Avg.Actions = [2013].Actions)
  AND (Avg.Energy_Union_Priority = [2013].Energy_Union_Priority)
  AND (Avg.World_player = [2013].World_player)
WHERE
  ((([2013].Sector)<>'INDIVIDUAL'))
GROUP BY
  [2013].Country,
  [2013].EU28,
  [2013].EEA,
  [2013].World_player,
  [2013].Energy_Union_Priority,
  [2013].Actions,
  [2013].Technology;
```


4.5.7 2015e

This query estimates patent for 2015 (Figure 50).

Figure 50. Patent projection, 2015e

```
SELECT
  Null AS A_ID,
  Null AS JRC_Id,
  Null AS Name,
  Null AS Sector,
  [2014e].Country,
  Null AS Reg_Code,
  Null AS Prov_Code,
  [2014e].EU28,
  [2014e].EEA,
  [2014e].World_player,
  "2015(e)" AS [Year],
  Null AS Tot_Fractional,
  [2014e].Energy_Union_Priority,
  [2014e].Actions,
  [2014e].Technology,
  Sum([2014e].[Energy_Tech_Fractional])*(1+Avg([Avg].[Avg])) AS Energy_Tech_Fractional,
  Null AS Granted,
  Null AS International,
  Null AS EPO,
  Null AS USPTO,
  Null AS SIPO,
  Null AS JPO,
  Null AS KIPO,
  Null AS EPO_Ctry,
  Null AS Other_Auth
FROM
  [Avg]
INNER JOIN
  2014e
  ON (Avg.World_player = [2014e].World_player)
  AND (Avg.Energy_Union_Priority = [2014e].Energy_Union_Priority)
  AND (Avg.Actions = [2014e].Actions)
  AND (Avg.Technology = [2014e].Technology)
GROUP BY
  [2014e].Country,
  [2014e].EU28,
  [2014e].EEA,
  [2014e].World_player,
  [2014e].Energy_Union_Priority,
  [2014e].Actions,
  [2014e].Technology;
```


4.6 Trend of Patent

This query combines all patent statistics calculated for every year, plus the projections for two years following the latest completed one (Figure 51).

Figure 51. Trend of Patent

```
SELECT * FROM 2000 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2001 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2002 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2003 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2004 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2005 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2006 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2007 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2008 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2009 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2010 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2011 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2012 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2013 where sector<>'INDIVIDUAL' and country IS NOT NULL
UNION
SELECT * FROM 2014e
UNION
SELECT * FROM 2015e;
```


5 R&D estimation

This section presents the procedure needed for the estimation of the R&D expenditure in private companies. It is based on patent statistics resulting from previous section 4 in combination with company financial data. These additional data are collected through different data providers (EU Industrial R&D Investment Scoreboard and Orbis) and are connected to the SETIS DB by means of the JRC_ID. In fact, the estimation procedure requires a preliminary connection between companies in the database and those present in these additional data providers for which financial data are available. The JRC_ID, in combination to an additional identification code, named A_ID, is also used to defining the parent-subsidiary relationship between companies and to re-construct the ownership structure of Multinational Corporations (MNCs). The following subsections illustrate queries (run in Microsoft Access) needed to calculate R&D expenditures.

5.1 e0_List_Subsidiaries

This query gives a list of subsidiaries of MNCs (Figure 52).

Figure 52. R&D estimation, e0

```
SELECT
    Patent.A_ID,
    Patent.JRC_Id,
    Patent.Name,
    Patent.Sector,
    Patent.Country,
    Patent.Reg_Code,
    Patent.Prov_Code,
    Patent.EU28,
    Patent.EEA,
    Patent.World_player,
    Patent.Year,
    Patent.Tot_Fractional,
    Sum(Patent.Energy_Tech_Fractional) AS Energy_Fractional
FROM
    Patent
GROUP BY
    Patent.A_ID,
    Patent.JRC_Id,
    Patent.Name,
    Patent.Sector,
    Patent.Country,
    Patent.Reg_Code,
    Patent.Prov_Code,
    Patent.EU28,
    Patent.EEA,
    Patent.World_player,
    Patent.Year,
    Patent.Tot_Fractional
HAVING
    (((Patent.A_ID)>0) AND ((Patent.Sector)="COMPANY"));
```


5.2 e1_List_Groups

This query gives a list of parent companies of MNCs, and calculates groups' statistics: total fractional, total energy fractional, unitary cost (Figure 53).

Figure 53. R&D estimation, e1

```
SELECT
    [Data_R&D].ID_Own,
    [Data_R&D].Name_LV1_own AS Group_Name,
    [Data_R&D].CTRY_Own AS Group_Country,
    Countries.world_player_code AS Group_Region,
    [Data_R&D].Sector AS Group_Sector,
    Count(e0_List_Subsiaries.Tot_Fractional) AS Group_Size,
    [Data_R&D].[Total_R&D] AS [Group_R&D],
    [Data_R&D].Net_Sales AS Group_Sales,
    [Data_R&D].Employees AS Group_Employees,
    [Data_R&D].Year,
    Sum(e0_List_Subsiaries.Tot_Fractional) AS Group_Fractional,
    Sum(e0_List_Subsiaries.Energy_Fractional) AS Group_Energy_Fractional,
    [Data_R&D].[Total_R&D]/Sum([e0_List_Subsiaries].Tot_Fractional) AS UC_Group,
    [Data_R&D].UC_Limit,
    IIf((([Data_R&D].[Total_R&D]/Sum([e0_List_Subsiaries].Tot_Fractional))<[Data_R&D].
        UC_Limit,1,0) AS Check_UC,
    [Data_R&D].Parent_Limit
FROM
    ([Data_R&D]
RIGHT JOIN
    e0_List_Subsiaries
    ON ([Data_R&D].ID_Own = e0_List_Subsiaries.A_ID)
    AND ([Data_R&D].Year = e0_List_Subsiaries.Year))
LEFT JOIN
    Countries
    ON [Data_R&D].CTRY_Own = Countries.Country
GROUP BY
    [Data_R&D].ID_Own,
    [Data_R&D].Name_LV1_own,
    [Data_R&D].CTRY_Own,
    Countries.world_player_code,
    [Data_R&D].Sector,
    [Data_R&D].[Total_R&D],
    [Data_R&D].Net_Sales,
    [Data_R&D].Employees,
    [Data_R&D].Year,
    [Data_R&D].UC_Limit,
    [Data_R&D].Parent_Limit
ORDER BY
    [Data_R&D].Name_LV1_own;
```


5.3 e2_Groups_for_UC

This query gives the list of parent companies of MNCs for which the group unitary cost is lower a predefined limit (Figure 54).

Figure 54. R&D estimation, e2

```
SELECT
    e1_List_Groups.*
FROM
    e1_List_Groups
WHERE
    ((([e1_List_Groups].Check_UC)=1));
```

5.4 e3_Sector_UC

Based on the sector of economic activity of all groups, this query calculates the sector unitary cost per group region, or residence country (Figure 55).

Figure 55. R&D estimation, e3

```
SELECT
    e2_Groups_for_UC.Year,
    e2_Groups_for_UC.Group_Region,
    e2_Groups_for_UC.Group_Sector,
    Count(e2_Groups_for_UC.Group_Size) AS Tot_Groups,
    Sum(e2_Groups_for_UC.Group_Size) AS Tot_Subsidaries,
    Sum(e2_Groups_for_UC.[Group_R&D]) AS [R&D_Sector],
    Sum(e2_Groups_for_UC.Group_Fractional) AS Fractional_Sector,
    Sum([e2_Groups_for_UC].[Group_R&D])/Sum([e2_Groups_for_UC].Group_Fractional) AS Sector_UC,
    Avg(e2_Groups_for_UC.Parent_Limit) AS Min_sector_size,
    IIf(Count([e2_Groups_for_UC].Group_Size)>=Avg([e2_Groups_for_UC].Parent_Limit),1,0) AS [Check]
FROM
    e2_Groups_for_UC
GROUP BY
    e2_Groups_for_UC.Year,
    e2_Groups_for_UC.Group_Region,
    e2_Groups_for_UC.Group_Sector
HAVING
    (((IIf(Count([e2_Groups_for_UC].[Group_Size])>=Avg([e2_Groups_for_UC].[Parent_Limit]),1,0))=1));
```


5.5 e4_R&D_eq7

This query permits to estimate R&D expenditure at technology level for all subsidiaries of the MNCs (Figure 56).

Figure 56. R&D estimation, e4

```

SELECT
    e1_List_Groups.ID_Own,
    e1_List_Groups.Group_Name,
    e1_List_Groups.Group_Country,
    e1_List_Groups.Group_Region,
    e1_List_Groups.Group_Sector,
    e3_Sector_UC.Tot_Groups AS Sector_Size,
    e3_Sector_UC.Sector_UC,
    e1_List_Groups.Group_Size,
    e1_List_Groups.UC_Group AS Group_UC,
    e1_List_Groups.[Group_R&D],
    e1_List_Groups.Group_Sales,
    e1_List_Groups.Group_Employees,
    e1_List_Groups.Group_Fractional,
    e1_List_Groups.Group_Energy_Fractional,
    Patent.JRC_Id,
    Patent.Name,
    Patent.Sector,
    Patent.Country, Patent.Reg_Code, Patent.Prov_Code,
    Patent.EU28, Patent.EEA,
    Patent.World_player,
    Patent.Year,
    Patent.Tot_Fractional,
    Patent.Energy_Union_Priority, Patent.Actions, Patent.Technology,
    Patent.Energy_Tech_Fractional, Patent.Granted, Patent.International,
    [Patent].[Energy_Tech_Fractional]*[e3_Sector_UC].[Sector_UC] AS RnD_eq7
FROM e3_Sector_UC
RIGHT JOIN (e1_List_Groups
RIGHT JOIN Patent
    ON e1_List_Groups.ID_Own = Patent.A_ID)
    ON (e3_Sector_UC.Group_Region = e1_List_Groups.Group_Region)
    AND (e3_Sector_UC.Year = e1_List_Groups.Year)
    AND (e3_Sector_UC.Group_Sector = e1_List_Groups.Group_Sector)
GROUP BY
    e1_List_Groups.ID_Own,
    e1_List_Groups.Group_Name,
    e1_List_Groups.Group_Country,
    e1_List_Groups.Group_Region,
    e1_List_Groups.Group_Sector,
    e3_Sector_UC.Tot_Groups,
    e3_Sector_UC.Sector_UC,
    e1_List_Groups.Group_Size,
    e1_List_Groups.UC_Group,
    e1_List_Groups.[Group_R&D],
    e1_List_Groups.Group_Sales,
    e1_List_Groups.Group_Employees,
    e1_List_Groups.Group_Fractional,
    e1_List_Groups.Group_Energy_Fractional,
    Patent.JRC_Id,
    Patent.Name,
    Patent.Sector,
    Patent.Country, Patent.Reg_Code, Patent.Prov_Code,
    Patent.EU28, Patent.EEA,
    Patent.World_player,
    Patent.Year,
    Patent.Tot_Fractional,
    Patent.Energy_Union_Priority, Patent.Actions, Patent.Technology,
    Patent.Energy_Tech_Fractional, Patent.Granted, Patent.International
HAVING (((Patent.Sector)="COMPANY"));

```


5.6 e5_Tech_UC

This query calculates the technology unitary cost, based on data regarding subsidiaries and detailed by country of residence (Figure 57).

Figure 57. R&D estimation, e5

```
SELECT
    [e4_R&D_eq7].Year,
    [e4_R&D_eq7].World_player,
    [e4_R&D_eq7].Energy_Union_Priority,
    [e4_R&D_eq7].Actions,
    [e4_R&D_eq7].Technology,
    Count([e4_R&D_eq7].Name) AS Tot_Subsidaries,
    Sum([e4_R&D_eq7].[RnD_eq7]) AS SumOfRnD_3,
    Sum([e4_R&D_eq7].Energy_Tech_Fractional) AS SumOfEnergy_Tech_Fractional,
    Sum([e4_R&D_eq7].[RnD_eq7])/Sum([e4_R&D_eq7].Energy_Tech_Fractional) AS Tech_UC
FROM
    [e4_R&D_eq7]
WHERE
    ((([e4_R&D_eq7].[RnD_eq7])>0))
GROUP BY
    [e4_R&D_eq7].Year,
    [e4_R&D_eq7].World_player,
    [e4_R&D_eq7].Energy_Union_Priority,
    [e4_R&D_eq7].Actions,
    [e4_R&D_eq7].Technology;
```

5.7 e6_Tech_UC_Total

This query calculates the technology unitary cost, based on data regarding subsidiaries without information on their corporate tax residence (Figure 58).

Figure 58. R&D estimation, e6

```
SELECT
    [e4_R&D_eq7].Year,
    [e4_R&D_eq7].Energy_Union_Priority,
    [e4_R&D_eq7].Actions,
    [e4_R&D_eq7].Technology,
    Count([e4_R&D_eq7].Name) AS Tot_Subsidaries,
    Sum([e4_R&D_eq7].RnD_eq7) AS SumOfRnD_3,
    Sum([e4_R&D_eq7].Energy_Tech_Fractional) AS SumOfEnergy_Tech_Fractional,
    Sum([e4_R&D_eq7].[RnD_eq7])/Sum([e4_R&D_eq7].Energy_Tech_Fractional) AS Tech_UC
FROM
    [e4_R&D_eq7]
WHERE
    ((([e4_R&D_eq7].RnD_eq7)>0))
GROUP BY
    [e4_R&D_eq7].Year,
    [e4_R&D_eq7].Energy_Union_Priority,
    [e4_R&D_eq7].Actions,
    [e4_R&D_eq7].Technology;
```


5.8 RnD Year

This query permits to combine in one table all R&D estimates calculated previously, for the complete dataset of companies (Figure 59).

Figure 59. R&D estimation, Year

```

SELECT
[e4_RnD_eq7].ID_Own,
[e4_RnD_eq7].Group_Name, [e4_RnD_eq7].Group_Country,
[e4_RnD_eq7].Group_Region, [e4_RnD_eq7].Group_Sector,
[e4_RnD_eq7].Sector_Size, [e4_RnD_eq7].Sector_UC,
[e4_RnD_eq7].Group_Size, [e4_RnD_eq7].[Group_RnD],
[e4_RnD_eq7].Group_UC, [e4_RnD_eq7].Group_Sales,
[e4_RnD_eq7].Group_Employees, [e4_RnD_eq7].Group_Fractional,
[e4_RnD_eq7].Group_Energy_Fractional,
[e4_RnD_eq7].JRC_Id,
[e4_RnD_eq7].Name,
[e4_RnD_eq7].Sector,
[e4_RnD_eq7].Country, [e4_RnD_eq7].Reg_Code, [e4_RnD_eq7].Prov_Code,
IIf([e4_RnD_eq7].ID_Own,IIf([e4_RnD_eq7].Country=[e4_RnD_eq7].Group_Country,"Home_Country",[e4_RnD_eq7].World_player)) AS Destination,
[e4_RnD_eq7].EU28, [e4_RnD_eq7].EEA,
[e4_RnD_eq7].World_player,
[e4_RnD_eq7].Year,
[e4_RnD_eq7].Tot_Fractional,
[e4_RnD_eq7].Energy_Union_Priority, [e4_RnD_eq7].Actions, [e4_RnD_eq7].Technology,
e5_Tech_UC.Tot_Subsidaries AS Tot_for_UC,
IIf([e5_Tech_UC].Tech_UC>0,[e5_Tech_UC].Tech_UC,[e6_Tech_UC_Total].Tech_UC) AS Tech_UC,
[e4_RnD_eq7].Energy_Tech_Fractional,
[e4_RnD_eq7].Granted,
IIf((IIf([e4_RnD_eq7].[RnD_eq7]>0,[e4_RnD_eq7].[RnD_eq7],[e4_RnD_eq7].Energy_Tech_Fractional*[e5_Tech_UC].Tech_UC)>0,IIf([e4_RnD_eq7].[RnD_eq7]>0,[e4_RnD_eq7].[RnD_eq7],[e4_RnD_eq7].Energy_Tech_Fractional*[e5_Tech_UC].Tech_UC)),[e4_RnD_eq7].Energy_Tech_Fractional*[e6_Tech_UC_Total].Tech_UC) AS RnD_Expenditure,
IIf([e4_RnD_eq7].[RnD_eq7]>0,"Eq_7","Eq_11") AS Formula
FROM e6_Tech_UC_Total
RIGHT JOIN (e5_Tech_UC
RIGHT JOIN [e4_RnD_eq7]
ON (e5_Tech_UC.Energy_Union_Priority = [e4_RnD_eq7].Energy_Union_Priority)
AND (e5_Tech_UC.Actions = [e4_RnD_eq7].Actions)
AND (e5_Tech_UC.Technology = [e4_RnD_eq7].Technology)
AND (e5_Tech_UC.World_player = [e4_RnD_eq7].World_player)
AND (e5_Tech_UC.Year = [e4_RnD_eq7].Year))
ON (e6_Tech_UC_Total.Technology = [e4_RnD_eq7].Technology)
AND (e6_Tech_UC_Total.Actions = [e4_RnD_eq7].Actions)
AND (e6_Tech_UC_Total.Energy_Union_Priority = [e4_RnD_eq7].Energy_Union_Priority)
AND (e6_Tech_UC_Total.Year = [e4_RnD_eq7].Year)
GROUP BY
[e4_RnD_eq7].ID_Own,
[e4_RnD_eq7].Group_Name, [e4_RnD_eq7].Group_Country,
[e4_RnD_eq7].Group_Region, [e4_RnD_eq7].Group_Sector,
[e4_RnD_eq7].Sector_Size, [e4_RnD_eq7].Sector_UC,
[e4_RnD_eq7].Group_Size, [e4_RnD_eq7].[Group_RnD],
[e4_RnD_eq7].Group_UC, [e4_RnD_eq7].Group_Sales,
[e4_RnD_eq7].Group_Employees, [e4_RnD_eq7].Group_Fractional,
[e4_RnD_eq7].Group_Energy_Fractional,
[e4_RnD_eq7].JRC_Id,
[e4_RnD_eq7].Name,
[e4_RnD_eq7].Sector,
[e4_RnD_eq7].Country, [e4_RnD_eq7].Reg_Code, [e4_RnD_eq7].Prov_Code,
[e4_RnD_eq7].EU28, [e4_RnD_eq7].EEA,
[e4_RnD_eq7].World_player,
[e4_RnD_eq7].Year,
[e4_RnD_eq7].Tot_Fractional,
[e4_RnD_eq7].Energy_Union_Priority, [e4_RnD_eq7].Actions, [e4_RnD_eq7].Technology,
e5_Tech_UC.Tot_Subsidaries,
[e4_RnD_eq7].Energy_Tech_Fractional, [e4_RnD_eq7].Granted, [e4_RnD_eq7].International,
e5_Tech_UC.Tech_UC,
[e4_RnD_eq7].RnD_eq7,
e6_Tech_UC_Total.Tech_UC
ORDER BY [e4_RnD_eq7].Name;

```


5.9 R&D projection

This section introduces the queries needed to calculate R&D projections for years following the latest one, for which the dataset is complete (in this example, 2013 is the latest complete year).

5.9.1 Patent_R&D_2010

This query sums patent fractional and R&D estimates at country, aggregated via the "world player" classification, and technology level for 2010 (Figure 60).

Figure 60. R&D projection, 2010

```
SELECT
  RnD_2010.World_player,
  RnD_2010.Energy_Union_Priority,
  RnD_2010.Actions,
  RnD_2010.Technology,
  Avg(RnD_2010.Tech_UC) AS 2010_UC,
  Sum(RnD_2010.Energy_Tech_Fractional) AS 2010,
  RnD_2010.Year
FROM
  RnD_2010
GROUP BY
  RnD_2010.World_player,
  RnD_2010.Energy_Union_Priority,
  RnD_2010.Actions,
  RnD_2010.Technology,
  RnD_2010.Year;
```

5.9.2 Patent_R&D_2011

This query sums patent fractional and R&D estimates at country, aggregated via the "world player" classification, and technology level for 2011 (Figure 61).

Figure 61. R&D projection, 2011

```
SELECT
  RnD_2011.World_player,
  RnD_2011.Energy_Union_Priority,
  RnD_2011.Actions,
  RnD_2011.Technology,
  Avg(RnD_2011.Tech_UC) AS 2011_UC,
  Sum(RnD_2011.Energy_Tech_Fractional) AS 2011,
  RnD_2011.Year
FROM
  RnD_2011
GROUP BY
  RnD_2011.World_player,
  RnD_2011.Energy_Union_Priority,
  RnD_2011.Actions,
  RnD_2011.Technology,
  RnD_2011.Year;
```


5.9.3 Patent_R&D_2012

This query sums patent fractional and R&D estimates at country, aggregated via the "world player" classification, and technology level for 2012 (Figure 62).

Figure 62. R&D projection, 2012

```
SELECT
  RnD_2012.World_player,
  RnD_2012.Energy_Union_Priority,
  RnD_2012.Actions,
  RnD_2012.Technology,
  Avg(RnD_2012.Tech_UC) AS 2012_UC,
  Sum(RnD_2012.Energy_Tech_Fractional) AS 2012,
  RnD_2012.Year
FROM
  RnD_2012
GROUP BY
  RnD_2012.World_player,
  RnD_2012.Energy_Union_Priority,
  RnD_2012.Actions,
  RnD_2012.Technology,
  RnD_2012.Year;
```

5.9.4 Patent_R&D_2013

This query sums patent fractional and R&D estimates at country, aggregated via the "world player" classification, and technology level for 2013 (Figure 63).

Figure 63. R&D projection, 2013

```
SELECT
  RnD_2013.World_player,
  RnD_2013.Energy_Union_Priority,
  RnD_2013.Actions,
  RnD_2013.Technology,
  Avg(RnD_2013.Tech_UC) AS 2013_UC,
  Sum(RnD_2013.Energy_Tech_Fractional) AS 2013,
  RnD_2013.Year
FROM
  RnD_2013
GROUP BY
  RnD_2013.World_player,
  RnD_2013.Energy_Union_Priority,
  RnD_2013.Actions,
  RnD_2013.Technology,
  RnD_2013.Year;
```


5.9.5 Patent_R&D_Avg

This query calculates first the change rates between two consecutive years (2010-2011, 2011-2012 and 2012-2013) and then it calculates the average.

Figure 64. R&D projection, Average

```
SELECT
  [Patent_R&D_2010].World_player,
  [Patent_R&D_2010].Energy_Union_Priority,
  [Patent_R&D_2010].Actions,
  [Patent_R&D_2010].Technology,
  (([Patent_R&D_2011].[2011]-[Patent_R&D_2010].[2010])/[Patent_R&D_2010].[2010] AS 201011,
  ([Patent_R&D_2012].[2012]-[Patent_R&D_2011].[2011])/[Patent_R&D_2011].[2011] AS 201112,
  ([Patent_R&D_2013].[2013]-[Patent_R&D_2012].[2012])/[Patent_R&D_2012].[2012] AS 201213,
  (([Patent_R&D_2011].[2011]-
    [Patent_R&D_2010].[2010])/[Patent_R&D_2010].[2010]+([Patent_R&D_2012].[2012]-
    [Patent_R&D_2011].[2011])/[Patent_R&D_2011].[2011]+([Patent_R&D_2013].[2013]-
    [Patent_R&D_2012].[2012])/[Patent_R&D_2012].[2012])/3 AS [Avg],
  ([Patent_R&D_2011].[2011_UC]-[Patent_R&D_2010].[2010_UC])/[Patent_R&D_2010].[2010_UC] AS
  201011_UC,
  ([Patent_R&D_2012].[2012_UC]-[Patent_R&D_2011].[2011_UC])/[Patent_R&D_2011].[2011_UC] AS
  201112_UC,
  ([Patent_R&D_2013].[2013_UC]-[Patent_R&D_2012].[2012_UC])/[Patent_R&D_2012].[2012_UC] AS
  201213_UC,
  (([Patent_R&D_2011].[2011_UC]-
    [Patent_R&D_2010].[2010_UC])/[Patent_R&D_2010].[2010_UC]+([Patent_R&D_2012].[2012_UC]-
    [Patent_R&D_2011].[2011_UC])/[Patent_R&D_2011].[2011_UC]+([Patent_R&D_2013].[2013_UC]-
    [Patent_R&D_2012].[2012_UC])/[Patent_R&D_2012].[2012_UC])/3 AS Avg_UC
FROM
  (([Patent_R&D_2010]
INNER JOIN
  [Patent_R&D_2011]
    ON ([Patent_R&D_2010].World_player = [Patent_R&D_2011].World_player)
    AND ([Patent_R&D_2010].Energy_Union_Priority = [Patent_R&D_2011].Energy_Union_Priority)
    AND ([Patent_R&D_2010].Actions = [Patent_R&D_2011].Actions)
    AND ([Patent_R&D_2010].Technology = [Patent_R&D_2011].Technology))
INNER JOIN
  [Patent_R&D_2012]
    ON ([Patent_R&D_2011].World_player = [Patent_R&D_2012].World_player)
    AND ([Patent_R&D_2011].Energy_Union_Priority = [Patent_R&D_2012].Energy_Union_Priority)
    AND ([Patent_R&D_2011].Actions = [Patent_R&D_2012].Actions)
    AND ([Patent_R&D_2011].Technology = [Patent_R&D_2012].Technology))
INNER JOIN
  [Patent_R&D_2013]
    ON ([Patent_R&D_2012].World_player = [Patent_R&D_2013].World_player)
    AND ([Patent_R&D_2012].Energy_Union_Priority = [Patent_R&D_2013].Energy_Union_Priority)
    AND ([Patent_R&D_2012].Actions = [Patent_R&D_2013].Actions)
    AND ([Patent_R&D_2012].Technology = [Patent_R&D_2013].Technology));
```


5.9.6 RnD_2014e

This query estimates R&D for 2014 (Figure 65).

Figure 65. R&D projection, 2014e

```
SELECT
  Null AS ID_Own,
  Null AS Group_Name,
  Null AS Group_Country,
  Null AS Group_Region,
  Null AS Group_Sector,
  Null AS Sector_Size,
  Null AS Sector_UC,
  Null AS Group_Size,
  Null AS [Group_R&D],
  Null AS Group_UC,
  Null AS Group_Sales,
  Null AS Group_Employees,
  Null AS Group_Fractional,
  Null AS Group_Energy_Fractional,
  Null AS JRC_Id,
  Null AS Name,
  RnD_2013.Sector,
  RnD_2013.Country,
  Null AS Destination,
  Null AS Reg_Code,
  Null AS Prov_Code,
  RnD_2013.EU28,
  RnD_2013.EEA,
  RnD_2013.World_player,
  "2014e" AS [Year],
  Null AS Tot_Fractional,
  RnD_2013.Energy_Union_Priority,
  RnD_2013.Actions,
  RnD_2013.Technology,
  Null AS Tot_for_UC,
  Avg([RnD_2013].[Tech_UC])*(1+Avg([Patent_R&D_Avg].[Avg_UC])) AS Tech_UC,
  Sum(RnD_2013.Energy_Tech_Fractional)*(1+Avg([Patent_R&D_Avg].Avg)) AS Energy_Tech_Fractional,
  Null AS Granted,
  (Sum([RnD_2013].[Energy_Tech_Fractional])*(1+Avg([Patent_R&D_Avg].[Avg]))*(Avg([RnD_2013].[T
ech_UC])*(1+Avg([Patent_R&D_Avg].[Avg_UC])))) AS RnD_Expenditure,
  Null AS Formula
FROM
  [Patent_R&D_Avg]
INNER JOIN
  RnD_2013
  ON ([Patent_R&D_Avg].Technology = RnD_2013.Technology)
  AND ([Patent_R&D_Avg].Actions = RnD_2013.Actions)
  AND ([Patent_R&D_Avg].Energy_Union_Priority = RnD_2013.Energy_Union_Priority)
  AND ([Patent_R&D_Avg].World_player = RnD_2013.World_player)
GROUP BY
  RnD_2013.Sector,
  RnD_2013.Country,
  RnD_2013.EU28,
  RnD_2013.EEA,
  RnD_2013.World_player,
  RnD_2013.Energy_Union_Priority,
  RnD_2013.Actions, RnD_2013.Technology;
```


5.9.7 RnD_2015e

This query estimates R&D for 2015 (Figure 66).

Figure 66. R&D projection, 2015e

```

SELECT
  Null AS ID_Own,
  Null AS Group_Name,
  Null AS Group_Country,
  Null AS Group_Region,
  Null AS Group_Sector,
  Null AS Sector_Size,
  Null AS Sector_UC,
  Null AS Group_Size,
  Null AS [Group_R&D],
  Null AS Group_UC,
  Null AS Group_Sales,
  Null AS Group_Fractional,
  Null AS Group_Energy_Fractional,
  Null AS Group_Energy_International,
  Null AS JRC_Id,
  Null AS Name,
  RnD_2014e.Sector,
  RnD_2014e.Country,
  Null AS Destination,
  Null AS Reg_Code,
  Null AS Prov_Code,
  RnD_2014e.EU28,
  RnD_2014e.EEA,
  RnD_2014e.World_player,
  "2015e" AS [Year],
  Null AS Tot_Fractional,
  RnD_2014e.Energy_Union_Priority,
  RnD_2014e.Actions,
  RnD_2014e.Technology,
  Null AS Tot_for_UC,
  Avg([RnD_2014e].[Tech_UC])*(1+Avg([Patent_R&D_Avg].[Avg_UC])) AS Tech_UC,
  Sum(RnD_2014e.Energy_Tech_Fractional)*(1+Avg([Patent_R&D_Avg].Avg)) AS Energy_Tech_Fractional,
  Null AS Granted,
  (Sum([RnD_2014e].[Energy_Tech_Fractional])*(1+Avg([Patent_R&D_Avg].[Avg]))*(Avg([RnD_2014e].[
  Tech_UC])*(1+Avg([Patent_R&D_Avg].[Avg_UC])))) AS RnD_Expenditure,
  Null AS Formula
FROM
  [Patent_R&D_Avg]
INNER JOIN
  RnD_2014e
  ON ([Patent_R&D_Avg].Technology = RnD_2014e.Technology)
  AND ([Patent_R&D_Avg].Actions = RnD_2014e.Actions)
  AND ([Patent_R&D_Avg].Energy_Union_Priority = RnD_2014e.Energy_Union_Priority)
  AND ([Patent_R&D_Avg].World_player = RnD_2014e.World_player)
GROUP BY
  RnD_2014e.Sector,
  RnD_2014e.Country,
  RnD_2014e.EU28,
  RnD_2014e.EEA,
  RnD_2014e.World_player,
  RnD_2014e.Energy_Union_Priority,
  RnD_2014e.Actions,
  RnD_2014e.Technology;

```


5.10 Private R&D

This query combines all R&D estimates calculated for every year, plus the projections for two years following the latest completed one (Figure 67).

Figure 67. Private R&D

```
SELECT * FROM RnD_2003  
  
UNION  
  
SELECT * FROM RnD_2004  
  
UNION  
  
SELECT * FROM RnD_2005  
  
UNION  
  
SELECT * FROM RnD_2006  
  
UNION  
  
SELECT * FROM RnD_2007  
  
UNION  
  
SELECT * FROM RnD_2008  
  
UNION  
  
SELECT * FROM RnD_2009  
  
UNION  
  
SELECT * FROM RnD_2010  
  
UNION  
  
SELECT * FROM RnD_2011  
  
UNION  
  
SELECT * FROM RnD_2012  
  
UNION  
  
SELECT * FROM RnD_2013  
  
UNION  
  
SELECT * FROM RnD_2014e  
  
UNION  
  
SELECT * FROM RnD_2015e;
```


6 Conclusions

This report has presented the procedural steps needed to build the SETIS Database. All steps are conceptually explained and the relative queries are also provided.

The database consists of three main sections. The first one regards the data cleaning process. This is the main part of the whole process, since it permits to overcome the lack of data accuracy and completeness coming from PATSTAT. This step is preliminary to the one regarding the calculation of patent statistics, which, in turn, fills the last one regarding the estimation of R&D expenditure at company and technology level.

As a result, the *ad hoc* designed DB facilitates and accelerates the construction of KPIs that SETIS produces in the context of Research, Innovation and Competitiveness of the European Union in the Energy sector.

References

European Commission, 2015. COM(2015) 80 final. A framework strategy for a resilient Energy Union with a forward-looking climate change policy, 25.02.2015, Brussels.

European Commission, 2016. Transforming the European Energy System through INNOVATION. ISBN 978-92-79-63468-0. Publications Office of the European Union, Luxembourg.

European Patent Office, 2016. Sample Queries and Tips for PATSTAT - version 2.1. From: [http://documents.epo.org/projects/babylon/eponet.nsf/0/70e17bf5392da428c1257aaa00385d88/\\$FILE/patstat_sample_queries_and_tips_en.pdf](http://documents.epo.org/projects/babylon/eponet.nsf/0/70e17bf5392da428c1257aaa00385d88/$FILE/patstat_sample_queries_and_tips_en.pdf). Access on: 2 August, 2016

European Patent Office, 2017. PATSTAT - Data Catalog 2017 Spring Edition - version 5.09. From: [http://documents.epo.org/projects/babylon/eponet.nsf/0/3B6E7B599A209A59C12581160040C828/\\$File/patstat_data_catalog_v5_09_en.pdf](http://documents.epo.org/projects/babylon/eponet.nsf/0/3B6E7B599A209A59C12581160040C828/$File/patstat_data_catalog_v5_09_en.pdf). Access on: 12 September, 2017

Fiorini, A., Georgakaki, A., Pasimeni, F., Tzimas, E., 2017. Monitoring R&I in Low-Carbon Energy Technologies. EUR 28446 EN. European Commission, From: <http://ec.europa.eu/jrc/en/publication/eur-scientific-and-technical-research-reports/monitoring-ri-low-carbon-energy-technologies>. Access on: 12 October 2017

List of abbreviations and definitions

CPC	Cooperative Classification system
EC	European Commission
EEE-PPAT	ECOOM-EUROSTAT-EPO PATSTAT Person Augmented Table
EPO	European Patent Offices
EU	European Union
ID	Identification code
KPI	Key Performance Indicator
MNCs	Multinational Corporations
MS	Member State (EU28)
NUTS	Nomenclature of Territorial Units for Statistics
PATSTAT	The Worldwide Patent Statistical Database maintained by EPO
RIC	Research, Innovation and Competitiveness
R&D	Research and Development
SETIS	Strategic Energy Technologies Information System
SETIS DB	SETIS Database
SQL	Structured Query Language

List of figures

Figure 1. Layout of the SETIS database	3
Figure 2. PATSTAT logical model diagram	4
Figure 3. Extraction query in PATSTAT, Appln_Id	6
Figure 4. Extraction query in PATSTAT, Person_Id	7
Figure 5. Extraction query in PATSTAT, CPC	8
Figure 6. Extraction query in PATSTAT, Country	9
Figure 7. Extraction query in PATSTAT, Company_Family	10
Figure 8. Stage 1 of the automatic cleaning process.....	11
Figure 9. Extraction query in PATSTAT, Sector_DOC_STD_NAME_ID	12
Figure 10. Extraction query in PATSTAT, Ctry_DOC_STD_NAME_ID	13
Figure 11. Extraction query in PATSTAT, Sector_HRM_ID	14
Figure 12. Extraction query in PATSTAT, Ctry_HRM_ID.....	15
Figure 13. Extraction query in PATSTAT, Nuts_DOC_STD_NAME_ID	16
Figure 14. Extraction query in PATSTAT, Nuts_HRM_ID	17
Figure 15. Query for cleaning, c0.....	18
Figure 16. Query for cleaning, c1.....	19
Figure 17. Query for cleaning, c2.....	19
Figure 18. Query for cleaning, c3.....	20
Figure 19. Query for cleaning, c4.....	20
Figure 20. Query for cleaning, c5.....	21
Figure 21. Query for cleaning, c6.....	22
Figure 22. Query for cleaning, c7.....	22
Figure 23. Query for cleaning, c8.....	23
Figure 24. Name allocation	24
Figure 25. Sector allocation	24
Figure 26. Country allocation	25
Figure 27. Region allocation.....	25
Figure 28. Query for total fractional, q1.....	27
Figure 29. Query for total fractional, q2.....	27
Figure 30. Query for total fractional, q3.....	28
Figure 31. Query for total fractional, q4.....	28
Figure 32. Granted patent families, g1	29
Figure 33. Granted patent families, g2	29
Figure 34. International patent families, i1	30
Figure 35. International patent families, i2	31
Figure 36. Company list	32
Figure 37. Patent families, o1.....	32

Figure 38. Patent families, o2.....	33
Figure 39. Patent families, o3.....	33
Figure 40. Patent families, o4.....	34
Figure 41. Patent families, o5.....	34
Figure 42. Patent families, o6.....	35
Figure 43. Patent families, Year.....	36
Figure 44. Patent projection, 2010.....	37
Figure 45. Patent projection, 2011.....	37
Figure 46. Patent projection, 2012.....	38
Figure 47. Patent projection, 2013.....	38
Figure 48. Patent projection, Average	39
Figure 49. Patent projection, 2014e	40
Figure 50. Patent projection, 2015e	41
Figure 51. Trend of Patent	42
Figure 52. R&D estimation, e0.....	43
Figure 53. R&D estimation, e1.....	44
Figure 54. R&D estimation, e2.....	45
Figure 55. R&D estimation, e3.....	45
Figure 56. R&D estimation, e4.....	46
Figure 57. R&D estimation, e5.....	47
Figure 58. R&D estimation, e6.....	47
Figure 59. R&D estimation, Year.....	48
Figure 60. R&D projection, 2010.....	49
Figure 61. R&D projection, 2011.....	49
Figure 62. R&D projection, 2012.....	50
Figure 63. R&D projection, 2013.....	50
Figure 64. R&D projection, Average	51
Figure 65. R&D projection, 2014e	52
Figure 66. R&D projection, 2015e	53
Figure 67. Private R&D	54

GETTING IN TOUCH WITH THE EU

In person

All over the European Union there are hundreds of Europe Direct information centres. You can find the address of the centre nearest you at: <http://europea.eu/contact>

On the phone or by email

Europe Direct is a service that answers your questions about the European Union. You can contact this service:

- by freephone: 00 800 6 7 8 9 10 11 (certain operators may charge for these calls),
- at the following standard number: +32 22999696, or
- by electronic mail via: <http://europa.eu/contact>

FINDING INFORMATION ABOUT THE EU

Online

Information about the European Union in all the official languages of the EU is available on the Europa website at: <http://europa.eu>

EU publications

You can download or order free and priced EU publications from EU Bookshop at: <http://bookshop.europa.eu>. Multiple copies of free publications may be obtained by contacting Europe Direct or your local information centre (see <http://europa.eu/contact>).

JRC Mission

As the science and knowledge service of the European Commission, the Joint Research Centre's mission is to support EU policies with independent evidence throughout the whole policy cycle.



EU Science Hub
ec.europa.eu/jrc



@EU_ScienceHub



EU Science Hub - Joint Research Centre



Joint Research Centre



EU Science Hub



Publications Office

doi: 10.2760/37901

ISBN 978-92-79-75369-5