

# COHERE

## Collaboration of Observational HIV Epidemiological Research Europe

### Version 1.0

### List of QA checks for data transfer

### PLATO II merger

#### List of cohort studies to be contacted:

##### **Bordeaux RCC cohorts**

ANRS CO1/CO10/CO11 EPF, ANRS CO2 SEROCO, ANRS CO3 AQUITAINE, ANRS CO4 FHDH, ANRS CO6 PRIMO, ANRS CO8 COPILOTE, CASCADE, Co-RIS, GEMES-Haemo, HIV-MIP, MADRID HIV Children, PISCIS, VACH

##### **Copenhagen RCC cohorts**

AMACS, ATHENA, CHIC, CHIPS, DANISH HIV Cohort, ECS, EuroSIDA, FRANKFURT HIV, ICC, ICONA, IMIT, ITLR, KOMPNET, MASTER, MoCHIV, MODENA, NSHPC, SAN RAFFAELE, SHCS, ST PIERRE

#### **Bordeaux RCC**

[Institut de Santé Publique,](#)  
d'Épidémiologie et de Développement (ISPED)  
Université Victor Segalen Bordeaux 2  
Casier postal n° 11  
146 rue Léo Saignat  
F-33076 Bordeaux cedex  
France  
Phone: +33 557 571 580  
Fax: +33 557 571 172

<mailto:Genevieve.Chene@isped.u-bordeaux2.fr>

#### **Copenhagen RCC**

[Copenhagen HIV Programme](#)  
University of Copenhagen  
Faculty of Health Sciences  
The Panum Institute  
Building 21.1 Blegdamsvej 3B  
2200 Copenhagen N  
Denmark  
Phone: +45 35 45 57 57  
Fax: +45 35 45 57 58

<mailto:jdl@cphiv.dk>

**List of checks, click on text to jump to SQL code:**

<b>Inclusion/eligibility criteria checks .....</b>	<b>5</b>
<b>ART_SD.....</b>	<b>5</b>
Date of start of ART (ART_SD) is before 1998-01-01 or the date is in the future .....	5
<b>CD4_D.....</b>	<b>5</b>
Date of CD4 measurement is more than 6 months prior to starting ART .....	5
<b>RNA_D.....</b>	<b>5</b>
Date of HIV-RNA measurement is more than 6 months prior to starting ART .....	5
<b>BAS file checks.....</b>	<b>6</b>
<b>PATIENT.....</b>	<b>6</b>
Patient ID (PATIENT) is duplicated .....	6
Patient ID (PATIENT) is present in any of the other files but not in this file .....	6
<b>BIRTH_D.....</b>	<b>6</b>
Date of birth (BIRTH_D) is missing or unknown (coded as 1911-11-11) .....	6
Date of birth (BIRTH_D) in indicates a patient that is more than 100 years old or the date is in the future .....	6
<b>ENROL_D.....</b>	<b>6</b>
Date of enrolment (ENROL_D) is missing .....	6
Date of enrolment (ENROL_D) is before date of birth/1980-01-01 (what ever comes latest or is available) or the date is in the future.....	7
<b>GENDER.....</b>	<b>7</b>
GENDER is not a valid code: 1 (male), 2 (female) or 9 (unknown) .....	7
<b>MODE.....</b>	<b>7</b>
Mode of infection (MODE) is not a valid code or is missing (not 99 = unknown) .....	7
Mode of other infection (MODE_OTH) is present but MODE is different from 90 .....	7
<b>MODE_OTH.....</b>	<b>7</b>
Mode of other infection (MODE_OTH) is missing despite MODE = 90 .....	7
<b>ORIGIN.....</b>	<b>7</b>
Origin (ORIGIN) is not a valid code or is missing (not 99 = unknown) .....	7
<b>RACE.....</b>	<b>7</b>
Race is not a valid code or is missing (neither 98 = prohibited nor 99 = unknown) .....	7
<b>SEROCO_D.....</b>	<b>8</b>
Date of seroconversion (SEROCO_D) is missing (not coded as unknown 1911-11-11) .....	8
Date of seroconversion (SEROCO_D) is before date of birth/1980-01-01 (what ever comes latest or is available) or in the future .....	8
Date of seroconversion (SEROCO_D) is after the date of AIDS (if present) .....	8
<b>RE CART_Y.....</b>	<b>8</b>
Ever received ART status (RE CART_Y) is missing, or is not 1 (yes), 0 (no) or 9 (unknown).....	8
<b>AIDS_Y.....</b>	<b>8</b>
The AIDS status (AIDS_Y) status is missing, or is not a valid code: 1 (yes), 0 (no) or 9 (unknown) .....	8
The AIDS status (AIDS_Y) is 0 or 9 but the variable AIDS_D is not empty .....	8
The AIDS status (AIDS_Y) is 1 but the variable AIDS_D is empty.....	8
<b>AIDS_D.....</b>	<b>9</b>
The date of onset of AIDS (AIDS_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available) or in the future .....	9
The date of onset of AIDS (AIDS_D) is present but there is no matching CDC stage C opportunistic disease in file DIS with a matching date .....	9
The date of onset of AIDS (AIDS_D) is missing but there is a CDC stage C opportunistic AIDS defining disease in file DIS .....	9
<b>LTFU file checks .....</b>	<b>10</b>
<b>PATIENT.....</b>	<b>10</b>
Patient ID (PATIENT) is duplicated .....	10
<b>DROP_Y.....</b>	<b>10</b>
Drop out/loss to follow-up status is not a valid code: No (0) or Yes (1).....	10
Drop out/loss to follow-up status is missing while date (DROP_D) is present .....	10
<b>DROP_D.....</b>	<b>10</b>
Date is missing while reason and status is present. ....	10

Date of drop out/loss to follow-up is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available) or the date is in the future .....	10
Date of drop out/loss to follow-up is before any dates given in the other tables .....	10
<b>DROP_RS</b> .....	<b>11</b>
Reason (DROP_RS) is not a valid code or is missing (not 99 = unknown) while date and status are present. ....	11
<b>DEATH_Y</b> .....	<b>11</b>
Death status is not a valid code: No (0) or Yes (1) .....	11
Death status is missing while date (DEATH_D) or reason(s) (DEATH_R1...) are present.....	12
<b>DEATH_D</b> .....	<b>12</b>
Date is missing while reason(s) and status are present. ....	12
Date of death is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available) or the date is in the future.....	12
Date of death is before any dates given in the other tables .....	12
<b>DEATH_R#</b> .....	<b>13</b>
Reason (DEATH_R#) is not a valid code or is missing (not 99 = unknown) while date and status are present. ....	13
<b>DEATH_RC#</b> .....	<b>13</b>
Causal relation (DEATH_RC#) is not a valid code or is missing (not N = not available) while reason is present. ....	13
<b>OVERLAP file checks</b> .....	<b>15</b>
<b>VIS file checks</b> .....	<b>16</b>
<b>All fields</b> .....	<b>16</b>
Duplicate records .....	16
<b>PATIENT</b> .....	<b>16</b>
All patients listed in BAS should have at least one entry in this file.....	16
<b>VIS_D</b> .....	<b>16</b>
The date of onset of visit (VIS_D) is before date of birth or in the future .....	16
<b>ART file checks</b> .....	<b>17</b>
<b>All fields</b> .....	<b>17</b>
Duplicate records .....	17
<b>PATIENT</b> .....	<b>17</b>
Patient ID is missing but there is a record in the file.....	17
<b>ART_ID</b> .....	<b>17</b>
The drug code (ART_ID) is not one of the valid HICDEP extended list of ATC codes or is missing. ....	17
<b>ART_SD</b> .....	<b>17</b>
Date of start of ART (ART_SD) is missing.....	17
Date of start of ART (ART_SD) is before 1998-01-01 or the date is in the future .....	17
<b>ART_ED</b> .....	<b>18</b>
Date of end of ART (ART_ED) is before the start date or the date is in the future .....	18
<b>ART_ID, ART_SD and ART_ED</b> .....	<b>18</b>
There is an overlap between multiple periods of treatment with the same drug .....	18
Combivir (J05AF30-COM) is used at the same time or overlaps use of Zidovudine (J05AF01) and/or Lamivudine (J05AF05).....	18
Trizivir (J05AF30-TZV) is used at the same time or overlaps use of Zidovudine (J05AF01), Lamivudine (J05AF05) and/or Abacavir (J05AF06) .....	18
Kivexa (J05AF30-KIV) is used at the same time or overlaps use of Lamivudine (J05AF05) and/or Abacavir (J05AF06) .....	19
Truvada (J05AF30-TRU) is used at the same time or overlaps use of Emtricitabine (J05AF09) and Tenofovir (J05AF07) .....	19
<b>ART_RS</b> .....	<b>19</b>
Reason for discontinuation (ART_RS) is not one of the valid HICDEP codes .....	19
Reason for discontinuation (ART_RS) is missing (not coded as 99) while stop date is present.....	19
Reason for discontinuation (ART_RS) is present while stop date is missing (not coded as unknown) .	19
<b>DIS file checks</b> .....	<b>20</b>
<b>All fields</b> .....	<b>20</b>
Duplicate records .....	20

<b>PATIENT</b> .....	<b>20</b>
Patient ID is not present but there is a record in the file .....	20
<b>DIS_ID</b> .....	<b>20</b>
Code to identify opportunistic event (DIS_ID) is not a valid code or is missing while date is present ...	20
<b>DIS_D</b> .....	<b>20</b>
Date of disease (DIS_D) is missing .....	20
Date of disease (DIS_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available as a criteria) or the date is in the future .....	20
<b>LAB_CD4 file checks</b> .....	<b>21</b>
<b>All fields</b> .....	<b>21</b>
Duplicate records, multiple measurements on the same date with the same value and on the same date but with different values and same unit .....	21
<b>PATIENT</b> .....	<b>21</b>
All patients listed in BAS should have at least one entry in this file.....	21
Patient ID is not present but there is a record in the file .....	21
<b>CD4_D</b> .....	<b>21</b>
Date of CD4 measurement (CD4_D) is missing.....	21
Date of CD4 measurement (CD4_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available as a criteria) or the date is in the future.....	21
<b>CD4_V</b> .....	<b>21</b>
Value of CD4 measurement (CD4_V) is missing.....	21
Value of CD4 measurement (CD4_V) is not between 0 and 3000 if copies/mL or between 0 and 100 if % .....	22
<b>CD4_U</b> .....	<b>22</b>
Unit (CD4_U) is missing or not a valid unit: 'copies/mL' or '%' .....	22
<b>LAB_RNA file checks</b> .....	<b>23</b>
<b>All fields</b> .....	<b>23</b>
Duplicate records: multiple measurements on the same date with the same value, same limit, same assay or on the same date but with different values, same limit, same assay .....	23
<b>PATIENT</b> .....	<b>23</b>
All patients listed in BAS should have at least one entry in this file.....	23
Patient ID is not present but there is a record in the file .....	23
<b>RNA_D</b> .....	<b>23</b>
Date of viral load measurement (RNA_D) is missing .....	23
Date of viral load measurement (RNA_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available as a criteria) or the date is in the future .....	23
<b>RNA_V</b> .....	<b>23</b>
Value (RNA_V) is missing while date, limit or assay are present.....	23
Value (RNA_V) is below level of detection and it not coded with a negative sign as specified in the SOP (either -1 or detected value as a negative number) .....	24
<b>RNA_L</b> .....	<b>24</b>
Assays lower limit (RNA_L) is missing while date, value or assay is present .....	24
<b>RNA_T</b> .....	<b>24</b>
Assay (RNA_T) is not a valid code or is missing (not 99 = unknown) while date, value or limit are present. ....	24

## ***Inclusion/eligibility criteria checks***

### **ART\_SD**

Date of start of ART (ART\_SD) is before 1998-01-01 or the date is in the future

**SQL code**

```
SELECT *
FROM ART
WHERE (ART_SD > {fn NOW()}) OR (ART_SD < '1998-01-01')
```

### **CD4\_D**

Date of CD4 measurement is more than 6 months prior to starting ART

**SQL code**

```
SELECT LAB_CD4.PATIENT, LAB_CD4.CD4_D, LAB_CD4.CD4_V
FROM (SELECT PATIENT, MIN(ART_SD) AS FIRST_START
      FROM ART GROUP BY PATIENT) TRX_START
INNER JOIN LAB_CD4 ON
      TRX_START.PATIENT = LAB_CD4.PATIENT AND
      DATEADD(mm, -6, TRX_START.FIRST_START) > LAB_CD4.CD4_D
```

### **RNA\_D**

Date of HIV-RNA measurement is more than 6 months prior to starting ART

**SQL code**

```
SELECT LAB_RNA.PATIENT, LAB_RNA.RNA_D, LAB_RNA.RNA_V
FROM (SELECT PATIENT, MIN(ART_SD) AS FIRST_START
      FROM ART GROUP BY PATIENT) TRX_START
INNER JOIN LAB_RNA ON
      TRX_START.PATIENT = LAB_RNA.PATIENT AND
      DATEADD(mm, -6, TRX_START.FIRST_START) > LAB_RNA.RNA_D
```

## BAS file checks

### PATIENT

Patient ID (PATIENT) is duplicated

```
SQL code
SELECT PATIENT
FROM   BAS
GROUP BY PATIENT
HAVING (COUNT(*) > 1)
```

Patient ID (PATIENT) is present in any of the other files but not in this file

```
SQL code
SELECT PATIENT
FROM (
SELECT PATIENT
FROM   LTFU
UNION
SELECT PATIENT
FROM   ART
UNION
SELECT PATIENT
FROM   VIS
UNION
SELECT PATIENT
FROM   DIS
UNION
SELECT PATIENT
FROM   LAB_CD4
UNION
SELECT PATIENT
FROM   LAB_RNA
) ALL_PATIENTS
WHERE NOT (PATIENT IN (SELECT PATIENT FROM BAS))
```

### BIRTH\_D

Date of birth (BIRTH\_D) is missing or unknown (coded as 1911-11-11)

```
SQL code
SELECT *
FROM   BAS
WHERE (BIRTH_D IS NULL)
```

Date of birth (BIRTH\_D) indicates a patient that is more than 100 years old or the date is in the future

```
SQL code
SELECT *
FROM   BAS
WHERE (BIRTH_D > {fn NOW()}) OR (DATEDIFF(year, BIRTH_D, {fn NOW()}) > 100)
```

### ENROL\_D

Date of enrolment (ENROL\_D) is missing

```
SQL code
```

```
SELECT *
FROM BAS
WHERE (ENROL_D IS NULL)
```

Date of enrolment (ENROL\_D) is before date of birth/1980-01-01 (what ever comes latest or is available) or the date is in the future

```
SQL code
SELECT *
FROM BAS
WHERE (ENROL_D > {fn NOW()}) OR (ENROL_D < BIRTH_D) OR (ENROL_D <
'1980-01-01')
```

## GENDER

GENDER is not a valid code: 1 (male), 2 (female) or 9 (unknown)

```
SQL code
SELECT *
FROM BAS
WHERE NOT (GENDER IN (1, 2, 9) OR (GENDER IS NULL))
```

## MODE

Mode of infection (MODE) is not a valid code or is missing (not 99 = unknown)

```
SQL code
SELECT *
FROM BAS
WHERE (NOT (MODE IN (1, 2, 3, 4, 5, 6, 7, 8, 90, 99))) OR (MODE IS
NULL)
```

Mode of other infection (MODE\_OTH) is present but MODE is different from 90

```
SQL code
SELECT *
FROM BAS
WHERE (MODE <> 90) AND (NOT (MODE_OTH IS NULL))
```

## MODE\_OTH

Mode of other infection (MODE\_OTH) is missing despite MODE = 90

```
SQL code
SELECT *
FROM BAS
WHERE (MODE = 90)
AND (MODE_OTH IS NULL)
```

## ORIGIN

Origin (ORIGIN) is not a valid code or is missing (not 99 = unknown)

```
SQL code
SELECT *
FROM BAS
WHERE (NOT (ORIGIN IN (10, 11, 12, 20, 30, 40, 50, 51, 52, 60, 70, 71,
72, 99)))
OR (ORIGIN IS NULL)
```

## RACE

Race is not a valid code or is missing (neither 98 = prohibited nor 99 = unknown)

```
SQL code
```

```

SELECT *
FROM BAS
WHERE (NOT (RACE IN (10, 20, 21, 22, 30, 40, 50, 60, 1020, 1040, 2030,
3040, 98, 99)))
OR (RACE IS NULL)

```

## SEROCO\_D

Date of seroconversion (SEROCO\_D) is missing (not coded as unknown 1911-11-11)

```

SQL code
SELECT *
FROM BAS
WHERE (SEROCO_D IS NULL)

```

Date of seroconversion (SEROCO\_D) is before date of birth/1980-01-01 (what ever comes latest or is available) or in the future

```

SQL code
SELECT *
FROM BAS
WHERE (SEROCO_D > {fn NOW()}) OR (SEROCO_D < BIRTH_D) OR (SEROCO_D <
'1980-01-01')

```

Date of seroconversion (SEROCO\_D) is after the date of AIDS (if present)

```

SQL code
SELECT *
FROM BAS
WHERE (SEROCO_D > AIDS_D)

```

## RECART\_Y

Ever received ART status (RECART\_Y) is missing, or is not 1 (yes), 0 (no) or 9 (unknown)

```

SQL code
SELECT *
FROM BAS
WHERE (RECART_Y IS NULL) OR NOT (RECART_Y IN (0, 1, 9))

```

## AIDS\_Y

The AIDS status (AIDS\_Y) status is missing, or is not a valid code: 1 (yes), 0 (no) or 9 (unknown)

```

SQL code
SELECT *
FROM BAS
WHERE AIDS_Y IS NULL OR NOT(AIDS_Y IN (0, 1, 9))

```

The AIDS status (AIDS\_Y) is 0 or 9 but the variable AIDS\_D is not empty

```

SQL code
SELECT *
FROM BAS
WHERE AIDS_Y IN (0, 9)
AND (NOT (AIDS_D IS NULL))

```

The AIDS status (AIDS\_Y) is 1 but the variable AIDS\_D is empty

```

SQL code

```

```

SELECT *
FROM BAS
WHERE (AIDS_Y = 1) AND (AIDS_D IS NULL)

```

## AIDS\_D

The date of onset of AIDS (AIDS\_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available) or in the future

```

SQL code
SELECT *
FROM BAS
WHERE (AIDS_D > {fn NOW()}) OR (AIDS_D < BIRTH_D) OR (AIDS_D < '1980-01-01')

```

The date of onset of AIDS (AIDS\_D) is present but there is no matching CDC stage C opportunistic disease in file DIS with a matching date

```

SQL code
SELECT *
FROM BAS
WHERE (NOT (PATIENT IN (SELECT BAS.PATIENT FROM BAS INNER JOIN DIS ON
BAS.PATIENT = DIS.PATIENT AND BAS.AIDS_D = DIS.DIS_DATE))) AND (NOT
(AIDS_D IS NULL))

```

The date of onset of AIDS (AIDS\_D) is missing but there is a CDC stage C opportunistic AIDS defining disease in file DIS

```

SQL code
SELECT *
FROM BAS
WHERE (AIDS_D IS NULL) AND PATIENT IN (SELECT DISTINCT PATIENT FROM DIS)

```

## LTFU file checks

### PATIENT

Patient ID (PATIENT) is duplicated

```
SQL code
SELECT PATIENT
FROM LTFU
GROUP BY PATIENT
HAVING (COUNT(*) > 1)
```

### DROP\_Y

Drop out/loss to follow-up status is not a valid code: No (0) or Yes (1)

```
SQL code
SELECT *
FROM LTFU
WHERE NOT (DROP_Y IN (0, 1))
```

Drop out/loss to follow-up status is missing while date (DROP\_D) is present

```
SQL code
SELECT *
FROM LTFU
WHERE (DROP_Y IS NULL) AND NOT (DROP_D IS NULL)
```

### DROP\_D

Date is missing while reason and status is present.

```
SQL code
SELECT *
FROM LTFU
WHERE NOT (DROP_Y IS NULL) AND (DROP_D IS NULL)
```

Date of drop out/loss to follow-up is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available) or the date is in the future

```
SQL code
SELECT *
FROM BAS INNER JOIN LTFU ON BAS.PATIENT = LTFU.PATIENT
WHERE (DROP_D > {fn NOW()}) OR (DROP_D < BIRTH_D) OR (DROP_D < '1980-01-01')
```

Date of drop out/loss to follow-up is before any dates given in the other tables

```
SQL code
```

```

SELECT LTFU.PATIENT, LTFU.DROP_D, LATEST_DATES.LATEST_DATE
FROM (SELECT PATIENT, MAX(REG_D) AS LATEST_DATE FROM (
SELECT PATIENT, BIRTH_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, ENROL_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, SEROCO_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, AIDS_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, DROP_D AS REG_D
FROM LTFU
UNION
SELECT PATIENT, DEATH_D AS REG_D
FROM LTFU
UNION
SELECT PATIENT, VIS_D AS REG_D
FROM VIS
UNION
SELECT PATIENT, ART_SD AS REG_D
FROM ART
UNION
SELECT PATIENT, ART_ED AS REG_D
FROM ART
UNION
SELECT PATIENT, DIS_D AS REG_D
FROM DIS
UNION
SELECT PATIENT, CD4_D AS REG_D
FROM LAB_CD4
UNION
SELECT PATIENT, RNA_D AS REG_D
FROM LAB_RNA
) REG_DATES
GROUP BY PATIENT) LATEST_DATES INNER JOIN
LTFU ON LATEST_DATES.PATIENT = LTFU.PATIENT AND LATEST_DATES.LATEST_DATE
> LTFU.DROP_D

```

## DROP\_RS

Reason (DROP\_RS) is not a valid code or is missing (not 99 = unknown) while date and status are present.

```

SQL code
SELECT *
FROM LTFU
WHERE NOT(DROP_RS IN (1, 2, 3, 4, 5, 6, 7, 8, 9)) OR (DROP_RS IS NULL)

```

## DEATH\_Y

Death status is not a valid code: No (0) or Yes (1)

```

SQL code
SELECT *
FROM LTFU
WHERE NOT (DEATH_Y IN (0, 1))

```

Death status is missing while date (DEATH\_D) or reason(s) (DEATH\_R1...) are present

**SQL code**

```
SELECT *
FROM LTFU
WHERE (DEATH_Y IS NULL) AND (NOT (DEATH_D IS NULL) OR NOT (DEATH_R1 IS
NULL))
```

Please repeat these for every instance of DEATH\_R#

## DEATH\_D

Date is missing while reason(s) and status are present.

**SQL code**

```
SELECT *
FROM LTFU
WHERE (DEATH_D IS NULL) AND (NOT (DEATH_Y IS NULL) OR NOT (DEATH_R1 IS
NULL))
```

Please repeat these for every instance of DEATH\_R#

Date of death is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available) or the date is in the future

**SQL code**

```
SELECT *
FROM BAS INNER JOIN LTFU ON BAS.PATIENT = LTFU.PATIENT
WHERE (DEATH_D > {fn NOW()}) OR (DEATH_D < BIRTH_D) OR (DEATH_D <
SEROCO_D) OR (DEATH_D < '1980-01-01')
```

Date of death is before any dates given in the other tables

**SQL code**

```

SELECT LTFU.PATIENT, LTFU.DEATH_D, LATEST_DATES.LATEST_DATE
FROM (SELECT PATIENT, MAX(REG_D) AS LATEST_DATE FROM (
SELECT PATIENT, BIRTH_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, ENROL_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, SEROCO_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, AIDS_D AS REG_D
FROM BAS
UNION
SELECT PATIENT, DROP_D AS REG_D
FROM LTFU
UNION
SELECT PATIENT, DEATH_D AS REG_D
FROM LTFU
UNION
SELECT PATIENT, VIS_D AS REG_D
FROM VIS
UNION
SELECT PATIENT, ART_SD AS REG_D
FROM ART
UNION
SELECT PATIENT, ART_ED AS REG_D
FROM ART
UNION
SELECT PATIENT, DIS_D AS REG_D
FROM DIS
UNION
SELECT PATIENT, CD4_D AS REG_D
FROM LAB_CD4
UNION
SELECT PATIENT, RNA_D AS REG_D
FROM LAB_RNA
) REG_DATES
GROUP BY PATIENT) LATEST_DATES INNER JOIN
LTFU ON LATEST_DATES.PATIENT = LTFU.PATIENT AND LATEST_DATES.LATEST_DATE
> LTFU.DEATH_D

```

**DEATH\_R#**

Reason (DEATH\_R#) is not a valid code or is missing (not 99 = unknown) while date and status are present.

```

SQL code
SELECT *
FROM LTFU
WHERE (NOT (DEATH_D IS NULL) OR NOT (DEATH_Y IS NULL)) AND (
NOT (DEATH_R1 IN (1, 2, 3, 4, 4.1, 5, 6, 7, 7.1, 7.2, 8, 8.1, 8.2, 9,
10, 20, 90, 91, 92, 93, 99)) OR (DEATH_R1 IS NULL))

```

**DEATH\_RC#**

Causal relation (DEATH\_RC#) is not a valid code or is missing (not N = not available) while reason is present.

```

SQL code

```

```
SELECT *  
FROM LTFU  
WHERE ((DEATH_RC1 IS NULL) OR NOT (DEATH_RC1 IN ('I', 'U', 'C', 'N')))  
AND NOT (DEATH_R1 IS NULL)
```

## ***OVERLAP file checks***

Patients listed here should be crossed checked between datasets during a merger. It should be checked that values in PAT\_OTH and COH\_OTH do actually yield a match in the BAS table of the cohort specified in COH\_OTH

## **VIS file checks**

### **All fields**

Duplicate records

**SQL code**

```
SELECT PATIENT, VIS_D
FROM VIS
GROUP BY PATIENT, VIS_D
HAVING (COUNT(*) > 1)
```

### **PATIENT**

All patients listed in BAS should have at least one entry in this file

**SQL code**

```
SELECT *
FROM BAS
WHERE NOT(PATIENT IN (SELECT DISTINCT PATIENT FROM VIS))
```

### **VIS\_D**

The date of onset of visit (VIS\_D) is before date of birth or in the future

**SQL code**

```
SELECT *
FROM BAS INNER JOIN VIS ON BAS.PATIENT = VIS.PATIENT
WHERE (VIS_D > {fn NOW()}) OR (VIS_D < BIRTH_D)
```

## ART file checks

### All fields

Duplicate records

**SQL code**

```
SELECT PATIENT, ART_ID, ART_SD, ART_ED
FROM ART
GROUP BY PATIENT, ART_ID, ART_SD, ART_ED
HAVING (COUNT(*) > 1)
UNION
SELECT PATIENT, ART_ID, ART_SD
FROM ART
GROUP BY PATIENT, ART_ID, ART_SD
HAVING (COUNT(*) > 1)
UNION
SELECT PATIENT, ART_ID, ART_ED
FROM ART
GROUP BY PATIENT, ART_ID, ART_ED
HAVING (COUNT(*) > 1)
```

### PATIENT

Patient ID is missing but there is a record in the file

**SQL code**

```
SELECT *
FROM ART
WHERE PATIENT IS NULL
```

### ART\_ID

The drug code (ART\_ID) is not one of the valid HICDEP extended list of ATC codes or is missing.

**SQL code**

```
SELECT *
FROM ART
WHERE NOT (ART_ID IN (SELECT ART_ID FROM ART_CODE_DRUG))
```

This code assumes that the list of valid codes as been implemented as a table called ART\_CODE\_DRUG.

### ART\_SD

Date of start of ART (ART\_SD) is missing

**SQL code**

```
SELECT *
FROM ART
WHERE (ART_SD IS NULL)
```

Date of start of ART (ART\_SD) is before 1998-01-01 or the date is in the future

**SQL code**

```
SELECT *
FROM ART
WHERE (ART_SD > {fn NOW()}) OR (ART_SD < '1998-01-01')
```

## ART\_ED

Date of end of ART (ART\_ED) is before the start date or the date is in the future

### SQL code

```
SELECT *
FROM ART
WHERE (ART_SD > ART_ED) OR (ART_ED > {fn NOW()})
```

## ART\_ID, ART\_SD and ART\_ED

There is an overlap between multiple periods of treatment with the same drug

### SQL code

```
SELECT DISTINCT *
FROM ART ART_FIRST INNER JOIN
      ART ART_SECOND ON ART_FIRST.PATIENT = ART_SECOND.PATIENT AND
      ART_FIRST.ART_ID = ART_SECOND.ART_ID AND ART_FIRST.ART_SD <>
      ART_SECOND.ART_SD AND ART_FIRST.ART_ED <> ART_SECOND.ART_ED
WHERE (ART_FIRST.ART_SD >= ART_SECOND.ART_SD) AND (ART_FIRST.ART_SD <=
      ART_SECOND.ART_ED) OR (ART_FIRST.ART_ED >= ART_SECOND.ART_SD) AND
      (ART_FIRST.ART_ED <= ART_SECOND.ART_ED)
```

Please note that the above query will show duplicate records for each overlap and single records for nested periods

Combivir (J05AF30-COM) is used at the same time or overlaps use of Zidovudine (J05AF01) and/or Lamivudine (J05AF05)

### SQL code

```
SELECT ART.PATIENT, ART.ART_ID, ART.ART_SD, ART.ART_ED, ART.ART_RS,
      ART_OVERLAP.ART_ID AS ART_ID_OVERLAP, ART_OVERLAP.ART_SD AS
      ART_SD_OVERLAP, ART_OVERLAP.ART_ED AS ART_ED_OVERLAP, ART_OVERLAP.ART_RS
      AS ART_RS_OVERLAP
FROM ART INNER JOIN ART ART_OVERLAP ON ART.PATIENT = ART_OVERLAP.PATIENT
      AND ART.ART_ID <> ART_OVERLAP.ART_ID
WHERE (ART.ART_ID = 'J05AF30-COM') AND (ART_OVERLAP.ART_ID = 'J05AF01'
      OR ART_OVERLAP.ART_ID = 'J05AF05') AND ((ART.ART_SD > ART_OVERLAP.ART_SD
      AND ART.ART_SD < ART_OVERLAP.ART_ED) OR (ART.ART_ED > ART_OVERLAP.ART_SD
      AND ART.ART_ED < ART_OVERLAP.ART_ED) OR (ART_OVERLAP.ART_SD > ART.ART_SD
      AND ART_OVERLAP.ART_SD < ART.ART_ED) OR (ART_OVERLAP.ART_ED > ART.ART_SD
      AND ART_OVERLAP.ART_ED < ART.ART_ED))
```

Trizivir (J05AF30-TZV) is used at the same time or overlaps use of Zidovudine (J05AF01), Lamivudine (J05AF05) and/or Abacavir (J05AF06)

### SQL code

```
SELECT ART.PATIENT, ART.ART_ID, ART.ART_SD, ART.ART_ED, ART.ART_RS,
      ART_OVERLAP.ART_ID AS ART_ID_OVERLAP, ART_OVERLAP.ART_SD AS
      ART_SD_OVERLAP, ART_OVERLAP.ART_ED AS ART_ED_OVERLAP, ART_OVERLAP.ART_RS
      AS ART_RS_OVERLAP
FROM ART INNER JOIN ART ART_OVERLAP ON ART.PATIENT = ART_OVERLAP.PATIENT
      AND ART.ART_ID <> ART_OVERLAP.ART_ID
WHERE (ART.ART_ID = 'J05AF30-TZV') AND (ART_OVERLAP.ART_ID = 'J05AF01'
      OR ART_OVERLAP.ART_ID = 'J05AF05' OR ART_OVERLAP.ART_ID = 'J05AF06') AND
      ((ART.ART_SD > ART_OVERLAP.ART_SD AND ART.ART_SD < ART_OVERLAP.ART_ED)
      OR (ART.ART_ED > ART_OVERLAP.ART_SD AND ART.ART_ED < ART_OVERLAP.ART_ED)
      OR (ART_OVERLAP.ART_SD > ART.ART_SD AND ART_OVERLAP.ART_SD < ART.ART_ED)
      OR (ART_OVERLAP.ART_ED > ART.ART_SD AND ART_OVERLAP.ART_ED <
      ART.ART_ED))
```

Kivexa (J05AF30-KIV) is used at the same time or overlaps use of Lamivudine (J05AF05) and/or Abacavir (J05AF06)

**SQL code**

```
SELECT ART.PATIENT, ART.ART_ID, ART.ART_SD, ART.ART_ED, ART.ART_RS,
ART_OVERLAP.ART_ID AS ART_ID_OVERLAP, ART_OVERLAP.ART_SD AS
ART_SD_OVERLAP, ART_OVERLAP.ART_ED AS ART_ED_OVERLAP, ART_OVERLAP.ART_RS
AS ART_RS_OVERLAP
FROM ART INNER JOIN ART ART_OVERLAP ON ART.PATIENT = ART_OVERLAP.PATIENT
AND ART.ART_ID <> ART_OVERLAP.ART_ID
WHERE (ART.ART_ID = 'J05AF30-KIV') AND (ART_OVERLAP.ART_ID = 'J05AF05'
OR ART_OVERLAP.ART_ID = 'J05AF06') AND ((ART.ART_SD > ART_OVERLAP.ART_SD
AND ART.ART_SD < ART_OVERLAP.ART_ED) OR (ART.ART_ED > ART_OVERLAP.ART_SD
AND ART.ART_ED < ART_OVERLAP.ART_ED) OR (ART_OVERLAP.ART_SD > ART.ART_SD
AND ART_OVERLAP.ART_SD < ART.ART_ED) OR (ART_OVERLAP.ART_ED > ART.ART_SD
AND ART_OVERLAP.ART_ED < ART.ART_ED))
```

Truvada (J05AF30-TRU) is used at the same time or overlaps use of Emtricitabine (J05AF09) and Tenofovir (J05AF07)

**SQL code**

```
SELECT ART.PATIENT, ART.ART_ID, ART.ART_SD, ART.ART_ED, ART.ART_RS,
ART_OVERLAP.ART_ID AS ART_ID_OVERLAP, ART_OVERLAP.ART_SD AS
ART_SD_OVERLAP, ART_OVERLAP.ART_ED AS ART_ED_OVERLAP, ART_OVERLAP.ART_RS
AS ART_RS_OVERLAP
FROM ART INNER JOIN ART ART_OVERLAP ON ART.PATIENT = ART_OVERLAP.PATIENT
AND ART.ART_ID <> ART_OVERLAP.ART_ID
WHERE (ART.ART_ID = 'J05AF30-TRU') AND (ART_OVERLAP.ART_ID = 'J05AF09'
OR ART_OVERLAP.ART_ID = 'J05AF07') AND ((ART.ART_SD > ART_OVERLAP.ART_SD
AND ART.ART_SD < ART_OVERLAP.ART_ED) OR (ART.ART_ED > ART_OVERLAP.ART_SD
AND ART.ART_ED < ART_OVERLAP.ART_ED) OR (ART_OVERLAP.ART_SD > ART.ART_SD
AND ART_OVERLAP.ART_SD < ART.ART_ED) OR (ART_OVERLAP.ART_ED > ART.ART_SD
AND ART_OVERLAP.ART_ED < ART.ART_ED))
```

## ART\_RS

Reason for discontinuation (ART\_RS) is not one of the valid HICDEP codes

**SQL code**

```
SELECT PATIENT, ART_ID, ART_SD, ART_ED, ART_RS
FROM ART
WHERE NOT (ART_RS IN (1, 1.1, 1.2, 1.3, 1.4, 2, 3, 3.1, 3.2, 4, 5, 5.1,
5.2, 5.3, 6, 7, 8, 8.1, 9, 10, 90, 90.1, 91, 92, 92.1, 92.2, 92.3, 93,
93.1, 94, 94.1, 95, 96, 97, 98, 99))
```

Reason for discontinuation (ART\_RS) is missing (not coded as 99) while stop date is present

**SQL code**

```
SELECT PATIENT, ART_ID, ART_SD, ART_ED, ART_RS
FROM ART
WHERE (ART_RS IS NULL) AND NOT (ART_ED IS NULL)
```

Reason for discontinuation (ART\_RS) is present while stop date is missing (not coded as unknown)

**SQL code**

```
SELECT PATIENT, ART_ID, ART_SD, ART_ED, ART_RS
FROM ART
WHERE (ART_RS IS NULL) AND NOT (ART_ED IS NULL)
```

## DIS file checks

### All fields

Duplicate records

<b>SQL code</b>
<b>SELECT</b> PATIENT, DIS_ID, DIS_D <b>FROM</b> DIS <b>GROUP BY</b> PATIENT, DIS_ID, DIS_D <b>HAVING</b> (COUNT(*) > 1)

### PATIENT

Patient ID is not present but there is a record in the file

<b>SQL code</b>
<b>SELECT</b> PATIENT, DIS_ID, DIS_D <b>FROM</b> DIS <b>WHERE</b> PATIENT IS NULL

### DIS\_ID

Code to identify opportunistic event (DIS\_ID) is not a valid code or is missing while date is present

<b>SQL code</b>
<b>SELECT</b> * <b>FROM</b> DIS <b>WHERE</b> NOT (DIS_ID IN ( <b>SELECT</b> DIS_ID <b>FROM</b> DIS_CODE)) AND NOT (DIS_D IS NULL)

This code assumes that the list of valid codes as been implemented as a table called DIS\_CODE.

### DIS\_D

Date of disease (DIS\_D) is missing

<b>SQL code</b>
<b>SELECT</b> PATIENT, DIS_ID, DIS_D <b>FROM</b> DIS <b>WHERE</b> DIS_D IS NULL

Date of disease (DIS\_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available as a criteria) or the date is in the future

<b>SQL code</b>
<b>SELECT</b> * <b>FROM</b> DIS INNER JOIN BAS ON DIS.PATIENT = BAS.PATIENT <b>WHERE</b> (DIS_D > {fn NOW()}) OR (DIS_D < BIRTH_D) OR (DIS_D < SEROCO_D) OR (DIS_D < '1980-01-01')

## LAB\_CD4 file checks

### All fields

Duplicate records, multiple measurements on the same date with the same value and on the same date but with different values and same unit

**SQL code**

```
SELECT PATIENT, CD4_D
FROM LAB_CD4
GROUP BY PATIENT, CD4_D, CD4_V
HAVING (COUNT(*) > 1)
UNION
SELECT PATIENT, CD4_D
FROM LAB_CD4
GROUP BY PATIENT, CD4_D, CD4_U
HAVING (COUNT(*) > 1)
```

### PATIENT

All patients listed in BAS should have at least one entry in this file

**SQL code**

```
SELECT *
FROM BAS
WHERE NOT (PATIENT IN (SELECT DISTINCT PATIENT FROM LAB_CD4))
```

Patient ID is not present but there is a record in the file

**SQL code**

```
SELECT PATIENT, CD4_D, CD4_V, CD4_U
FROM LAB_CD4
WHERE PATIENT IS NULL
```

### CD4\_D

Date of CD4 measurement (CD4\_D) is missing

**SQL code**

```
SELECT PATIENT, CD4_D, CD4_V, CD4_U
FROM LAB_CD4
WHERE CD4_D IS NULL
```

Date of CD4 measurement (CD4\_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available as a criteria) or the date is in the future

**SQL code**

```
SELECT *
FROM LAB_CD4 INNER JOIN BAS ON LAB_CD4.PATIENT = BAS.PATIENT
WHERE (CD4_D > {fn NOW()}) OR (CD4_D < BIRTH_D) OR (CD4_D < SEROCO_D)
OR (CD4_D < '1980-01-01')
```

### CD4\_V

Value of CD4 measurement (CD4\_V) is missing

**SQL code**

```
SELECT PATIENT, CD4_D, CD4_V, CD4_U
FROM LAB_CD4
WHERE CD4_V IS NULL
```

Value of CD4 measurement (CD4\_V) is not between 0 and 3000 if copies/mL or between 0 and 100 if %

**SQL code**

```
SELECT *
FROM LAB_CD4
WHERE CD4_V < 0 OR CD4_V > 3000 AND CD4_U = 'copies/mL'
UNION
SELECT *
FROM LAB_CD4
WHERE CD4_V < 0 OR CD4_V > 100 AND CD4_U = '%'
```

It may be necessary to loosen the **WHERE** clauses to use **LIKE** for the CD4\_U code comparison

### CD4\_U

Unit (CD4\_U) is missing or not a valid unit: 'copies/mL' or '%'

**SQL code**

```
SELECT *
FROM LAB_CD4
WHERE (CD4_U IS NULL) OR NOT (CD4_U IN ('copies/mL', '%'))
```

## LAB\_RNA file checks

### All fields

Duplicate records: multiple measurements on the same date with the same value, same limit, same assay or on the same date but with different values, same limit, same assay

#### SQL code

```
SELECT PATIENT, RNA_D, RNA_V, RNA_L, RNA_T
FROM LAB_RNA
GROUP BY PATIENT, RNA_D, RNA_V, RNA_L, RNA_T
HAVING (COUNT(*) > 1)
UNION
SELECT PATIENT, RNA_D, NULL AS RNA_V, RNA_L, RNA_T
FROM LAB_RNA
GROUP BY PATIENT, RNA_D, RNA_L, RNA_T
HAVING (COUNT(*) > 1)
```

Please note that the latter test will not return the value of RNA\_V

### PATIENT

All patients listed in BAS should have at least one entry in this file

#### SQL code

```
SELECT *
FROM BAS
WHERE NOT (PATIENT IN (SELECT DISTINCT PATIENT FROM LAB_RNA))
```

Patient ID is not present but there is a record in the file

#### SQL code

```
SELECT *
FROM LAB_RNA
WHERE PATIENT IS NULL
```

### RNA\_D

Date of viral load measurement (RNA\_D) is missing

#### SQL code

```
SELECT *
FROM LAB_RNA
WHERE RNA_D IS NULL
```

Date of viral load measurement (RNA\_D) is before date of birth/1980-01-01/date of seroconversion (what ever comes latest or is available as a criteria) or the date is in the future

#### SQL code

```
SELECT *
FROM LAB_RNA INNER JOIN BAS ON LAB_RNA.PATIENT = BAS.PATIENT
WHERE (RNA_D > {fn NOW()}) OR (RNA_D < BIRTH_D) OR (RNA_D < SEROCO_D)
OR (RNA_D < '1980-01-01')
```

### RNA\_V

Value (RNA\_V) is missing while date, limit or assay are present

#### SQL code

```
SELECT *
FROM LAB_RNA
WHERE RNA_V IS NULL
```

Value (RNA\_V) is below level of detection and it not coded with a negative sign as specified in the SOP (either -1 or detected value as a negative number)

**SQL code**

```
SELECT *
FROM LAB_RNA
WHERE (RNA_V < RNA_L AND RNA_V > 0)
```

### RNA\_L

Assays lower limit (RNA\_L) is missing while date, value or assay is present

**SQL code**

```
SELECT *
FROM LAB_RNA
WHERE RNA_L IS NULL
```

### RNA\_T

Assay (RNA\_T) is not a valid code or is missing (not 99 = unknown) while date, value or limit are present.

**SQL code**

```
SELECT *
FROM LAB_RNA
WHERE (NOT (RNA_D IS NULL) OR NOT (RNA_V IS NULL) OR NOT (RNA_L IS
NULL)) AND ((RNA_T IS NULL) OR NOT (RNA_T IN
(5, 10, 15, 19, 20, 21, 29, 31, 32, 33, 39, 40, 41, 50, 51, 55, 56, 65,
66, 90, 99)))
```