Connected Bradford FDM Specification

v1.2

last updated – 20230308 - JDB

Connected Bradford has adopted the OMOP Common Data Model (CDM) for the majority of its patient-centred healthcare data.

Some datasets in Connected Bradford have features that do not sit well within either the OMOP conceptual vocabulary or the structure of the CDM. The aim of the Flexible Data Model (FDM) is to provide a set of design principles and a data model for non-OMOP datasets such that they can be structured and annotated in an OMOP-friendly way, and are therefore easy to combine with OMOP CDM data during analysis.

*CONTENTS*

[FDM design principles 2](#_Toc126323588)

[The Flexible Data Model 3](#_Toc126323589)

[FDM master dataset specification 3](#_Toc126323590)

[FDM source dataset specification 4](#_Toc126323591)

[Guide to building an FDM source dataset 8](#_Toc126323592)

[Derived datasets 11](#_Toc126323593)

[Derived dataset specification 11](#_Toc126323594)

[Appendices and Guides 13](#_Toc126323595)

[Example Observation Period Builder Script 13](#_Toc126323596)

[visit\_occurence - notes 15](#_Toc126323597)

[Populating the visit\_occurence table: 15](#_Toc126323598)

[FDM review checklist 17](#_Toc126323599)

# FDM design principles

*must* = mandatory

*should* = high priority but can be flexible if absolutely necessary

*may* = optional

1. The FDM is a data model that is applied to a lesser or greater extent to a range of different datasets
2. A query or analysis written for an OMOP CDM dataset *must* be able to be run on an FDM dataset with a minimum of changes - hence we follow the structure and conventions of an OMOP CDM as closely as possible.
3. An FDM dataset *must* only contain data from a single source organisation, and *may* only contain data from a part of a single source
4. An FDM dataset *must* only contain data from individuals with records in the source organisation dataset
5. An FDM dataset can be updated at any time, so in building the dataset, a scripted process *must* be developed to support a full refresh. Delta loads *should* also be supported (and this may be required for the bigger datasets).
6. An FDM dataset *must* contain a defined minimum set of OMOP tables, called the FDM Subset, to facilitate CDM and cross-FDM linkage. These *must* be subsets of the corresponding FDM master tables, i.e. they should only contain data that comes from the FDM master.
7. An FDM dataset *may* contain additional OMOP tables beyond the FDM Subset
8. An FDM dataset *must* contain one or more source-specific tables
9. Source-specific tables containing person-centred data *should* have one unique person-centred event per row, which means one thing that happened to the person or involving the person per row
10. Tables and columns *must* adopt OMOP names and naming conventions wherever possible
11. Data values *must* be transformed to OMOP standard concepts wherever possible. When this is done, the source values *must* also be held alongside the OMOP standard concepts in columns following OMOP naming conventions i.e. \_source\_value
12. An FDM dataset *must* contain a data dictionary table called data\_dictionary\*, containing a description of the contents of all non-OMOP columns in the dataset
13. An FDM must have default visualisations completed
14. Once completed the FDM must be reviewed by the developer and a peer for technical completeness and data quality(high level - not detailed)

# 

# The Flexible Data Model

## FDM master dataset specification

A master dataset containing the combined set of CBradford individuals across all datasets will be required to assist production of source datasets and derived datasets. The design requirements for the FDM master are as follows:

|  |  |
| --- | --- |
| Naming pattern | Current version   * CY\_FDM\_master   Previous version superseded on yyyymmdd   * CY\_FDM\_master\_{yyyymmdd} |
| FDM Subset | OMOP tables that must be included in the master dataset:   * person * observation\_period * care\_site   The master person table is used to provide harmonised demographics for use in all source and derived datasets.  The master care\_site table is needed to help avoid collisions between care\_site\_id across datasets.  The master observation\_period table is used to provide limits on plausible date ranges for observational data to assist data cleansing in the source and derived datasets. |
| cb\_change\_log | Tracks changes to source-specific FDM datasets, columns as follows:   * dataset\_name * source\_name * source\_alias * source\_part * release\_date * superseded\_date * changes   source\_name and source\_part are the same labels used in dataset\_name  source\_alias can be used to hold a more descriptive source and part name |

## FDM source dataset specification

Source datasets contain data from a single source data provider.

|  |  |
| --- | --- |
| Dataset naming pattern | Current version, complete source   * CY\_FDM\_{source-name}   Current version, part of source   * CY\_FDM\_{source-name}\_{part}   Previous version superseded on yyyymmdd, complete source   * CY\_FDM\_{source-name}\_{yyyymmdd}   Previous version superseded on yyyymmdd, part of source   * CY\_FDM\_{source-name}\_{part}\_{yyyymmdd} |
| FDM Subset | OMOP tables that *must* be included in every FDM dataset and linked together using \_id columns following OMOP CDM conventions:   * person * observation\_period * care\_site * visit\_occurrence\*   In FDM datasets, observation\_period should be based on the first and last events observed per person in the source dataset.  OMOP tables that *may* be included if helpful and if the data formats fit well:   * provider * visit\_detail   Note that providers cannot be harmonised across datasets due to not having unique identifiers for clinical providers.  Other OMOP clinical event tables may be considered for inclusion, such as observation, measurement, drug\_exposure etc. However, if these tables are to be included then you should consider whether a CDM model would be more appropriate.  [\*see Populating the visit\_occurrence table below](#fuum7veqptne) |
| Source-specific tables | The dataset must contain at least one source-specific table of data that cannot be fully transformed to the OMOP CDM.  The preferred data format is one person-specific event per row.  OMOP naming conventions must be observed for **tables**:   * all lowercase, underscore-separated nouns * examples: education\_assessment, social\_care\_assessment, ncmp\_measure * prefix tbl\_ for all source data tables.   OMOP naming conventions and value types must be observed for **columns**:   * primary keys are named {event}\_id, e.g. assessment\_id, ncmp\_measurement\_id * dates are stored as both dates and datetimes values and are named {event}\_date and {event}\_datetime * column names are underscore-separated, all lowercase * further examples: assessment\_id, assessment\_datetime, assessment\_reason, value\_as\_number, value\_as\_string   OMOP **standard concepts** must be used wherever possible:   * Some concepts can be harmonised with the CDM using standard concepts and column naming, e.g. gender\_concept\_id, place\_of\_service\_concept\_id, unit\_concept\_id * Where these are used, also provide corresponding \_source\_value and \_source\_concept\_id columns if possible   Standard Columns   * person\_id must be the first column in the source specific data. * Where the data contains a single event date, this must be retained as is, with the addition of the field fdm\_date in date time format yyyy-mm-dd hh:ss:mmm (or date format yyyy-mm-dd if no time element is available) * Where the data contains “date from” and “date to” fields, these must be retained as is, with the addition of fields in the standard OMOP format : {table\_name}\_start\_date and {table\_name}\_end\_date for compatibility with the CDM? Eg. <https://ohdsi.github.io/CommonDataModel/cdm54.html>   …where in the OMOP CDM:  condition table = condition\_start\_date, condition\_end\_date  observation\_period table = observation\_period\_start\_date, observation\_period\_end\_date  etc.  So if the srcode table then it’s srcode\_start\_date, srcode\_end\_date; if it’s the spell table then it’s spell\_start\_date, spell\_end\_date etc   * Where the data contains a single event date, this must be retained as is, with the addition of the fields {table\_name}\_start\_date and {table\_name}\_end\_date, using the single date for both of these new fields. * Using a datetime format yyyy-mm-dd hh:ss:mmm (or date format yyyy-mm-dd if no time element is available). * In all cases where the fdm dates are built the data dictionary must state which fields have been used to derive these.   Format of source tables   * In all cases the formats for source tables will take the form: person\_id, *xxx*\_start\_date, *xxx*\_end\_date,…*source data fields.*   Naming Conventions for tables   * tbl\_ for all source data tables * cb\_ for connected Bradford tables used for any other purpose. * tmp\_ for tables used as part of the build process. These should be deleted before the FDM is released for use. * Omop standard for : person, observation\_period, care\_site, visit\_occurrence. |
| cb\_change\_log | Tracks changes to the source-specific FDM dataset. This should be built and updated in the FDM master (see [FDM master dataset specification](#_u47hzdx4m6x9)) but the rows relevant to each FDM source should be copied to the FDM source dataset so the information is available to all users.  Columns as follows:   * dataset\_name * source\_name * source\_alias * source\_part * release\_date * superseded\_date * changes   source\_name and source\_part are the same labels used in dataset\_name  source\_alias can be used to hold a more descriptive source and part name |
| data\_dictionary | This is a document summarising the data source, any assumptions and any information that may assist the researcher in determining its usefulness.  It will also contain table specifications containing a minimum of five columns:   * Table - describes the table, and its source * column * value\_type * example\_values * description   Together, these describe the contents of the columns in the non-OMOP tables in the dataset.  Further columns may be added if more detail is available. |
| Data Visualisation | Google Data Studio visualisations completed as per the templated visualisation tool |
| Peer Review | FDM and associated documentation to be validated |
| Check List | FDM [check list](#dwsqjlkksllf) to be completed |

# 

# Guide to building an FDM source dataset

*Note on ETL of new information received into FDM master*

ETL of person, observation\_period and care\_site data from the source into the FDM master tables must be completed prior to building an FDM source. Responsibility for this lies with the Senior Database Manager. For each person, observation period should be based on date of birth as the minimum and date of death + 42 days as maximum. The ETL needs to take into account any new deaths information received and update observation periods accordingly.

|  |  |
| --- | --- |
| Step 1 | Set up a juypter notebook for code that will be developed to build the dataset and support full refresh and delta load procedures. |
| Step 2 | Create a source-specific FDM dataset and add into this the person and care\_site FDM Subset tables from the FDM master, filtering in only the individuals and care sites with records in the source. |
| Step 4 | If there are person and care\_site -related fields in the FDM source that are not covered in the FDM master equivalents, add these columns to the FDM subset tables. These will need to be added to the data\_dictionary table in Step 10 as they are not covered by OMOP documentation. |
| Step 5 | The observation\_period table should contain the minimum and maximum event dates for each person within the source, but having filtered out implausible dates for each person. The maximum allowable date for an observation period end is the date the data was extracted from source.  Use FDM master observation\_period to filter out implausible dates from each table for each person. Use the remaining min/max date per person across all tables, with the date of source data extraction as maximum, to define the observation period for each person and use this to populate the observation\_period table. |
| Step 6 | Each person-centred event in a source specific table should link to a visit\_occurrence record via visit\_occurrence\_id. Investigate the structure of visit-like events in the source and plan how to represent these in visit\_occurrence and source-specific event tables. This should inform whether to build visit\_occurence or the source-specific event tables first. Refer to bookmark [Populating the visit\_occurrence table](#kix.etaoq8z0ckm9) for more information on how to populate the columns of the visit\_occurrence table. |
| Step 7 | Work out which fields should be mapped to OMOP. This should be done in discussion with the CBradford team, and should be done whenever there are concepts in the OMOP vocabulary that cover the concepts in the source data.  OMOP concepts can be explored here:  <https://athena.ohdsi.org/search-terms/start>  After a matching OMOP concept is identified it should be mapped to its standard concept using the “Maps to” relationship - this can be done via the athena website or queried directly from the CDM vocabulary tables.  The initially matched concept goes into a \_source\_concept\_id field, e.g. assessment\_source\_concept\_id  The source value that was mapped goes into a \_source\_value field, e.g. assessment\_source\_value  The standard concept ID goes into a \_concept\_id field, e.g. assessment\_concept\_id  Note that it’s possible to produce more than one standard concept for a source, in which case these are entered into separate records linked to the same visit. |
| Step 8 | Work out column names to be applied so that they match OMOP conventions as closely as possible, e.g. all lowercase, underscore-separated, and use OMOP names where appropriate.  For example: person\_id, gender\_concept\_id, gender\_source\_value, education\_assessment\_type, education\_assessment\_score |
| Step 9 | Work out which values need to be transformed to match OMOP CDM data formats. For example, check OMOP date and datetime formats; apply the value\_as\_number, value\_as\_string, value\_as\_date convention. |
| Step 9 | The observation\_period table should contain the minimum and maximum event dates for each person within the source, but having filtered out implausible dates for each person. The maximum allowable date for an observation period end is the date the data was extracted from source.  Use FDM master observation\_period to filter out implausible dates from each table for each person. Use the remaining min/max date per person across all tables, with the date of source data extraction as maximum, to define the observation period for each person and use this to populate the observation\_period table.  An example script that can be adapted for your use is available [here](#_pyh5oqfp4yeb) |
| Step 10 | By this stage you have worked out:   * Which activities have implausible dates and should be dropped (Step 5) * How to link person-centred activity to visits (Step 6) * How to map concepts (Step 7) * How to name the columns (Step 8) * How to transform value types (Step 9)   Now create and populate the source-specific tables. |
| Step 11 | Create a data\_dictionary table in the source-specific dataset. This must contain definitions for any fields that are not from the OMOP CDM and therefore not covered by published OMOP documentation. Refer to [FDM source dataset specification](#_u7q7m4uc65ce). |
| Step 12  Step 13  Step 14 | Build the Data visualisations using Data Studio (Documentation to be provided)  Peer review of the FDM - [complete review checklist](#dwsqjlkksllf) \*  Update change\_log in FDM master and copy to source FDM. Refer to [FDM master dataset specification](#_u47hzdx4m6x9) and [FDM source dataset specification](#_u7q7m4uc65ce). |
| Step 15 | Review scripted code and write documentation to support future full refresh and delta load procedures. Final scripts should be made available via github. |

# Derived datasets

Derived Datasets are data freezes that are generated by pulling data from one or more FDM or CDM datasets, and may be produced to support specific projects or certain topic areas.

The design of a Derived Dataset should follow the FDM closely, but there may be differences due to the fact that a Derived Dataset is not necessarily intended to support data linkage across datasets. As Derived Datasets can pull in data from multiple FDM and CDM datasets, linking across these could result in duplication of information.

## Derived dataset specification

|  |  |
| --- | --- |
| Dataset naming pattern | Current version   * CY\_DD\_{dataset-name}   Previous version superseded on yyyymmdd   * CY\_DD\_{dataset-name}\_{yyyymmdd} |
| DD Subset | OMOP tables that *must* be included in every Derived Dataset:   * person * observation\_period * care\_site * visit\_occurrence   In Derived Datasets, observation\_period should be based on the first and last events observed per person across all sources included in the Derived Dataset.  It should be possible to derive combined person and observation\_period tables directly from the source datasets. With visit\_occurrence, however, this would lead to collisions in visit\_occurrence\_ids, so a new visit\_occurrence table will have to be built. |
| Source-specific tables | This follows the [FDM source dataset specification](#_u7q7m4uc65ce) |
| source\_tables | Lists which source tables were used in the creation of the Derived Dataset. This is not only important provenance information, but also helps a user avoid linking the Derived Dataset to other data that may introduce unintentional duplication of information.  The following columns are required, one row per source table:   * dataset\_name * table\_name * added\_date * removed\_date   removed\_date should be populated if an update to the Derived Dataset is released that removes a source table that was previously included. |
| change\_log | Tracks changes to the Derived Dataset.  Columns as follows:   * dataset\_name * release\_date * superseded\_date * changes |
| data\_dictionary | This is a table containing a minimum of five columns:   * table * column * value\_type * example\_values * description   Together, these describe the contents of the columns in the non-OMOP tables in the dataset.  Further columns may be added if more detail is available. |

## 

# Appendices and Guides

## Example Observation Period Builder Script

The following is an example script that uses one or more Event based tables to determine dates using in the dataset, dates that are not valid and insert into the observation period table

#Qmak\_BDCT\_ObservationPeriod\_Part1

#Last modified : 202200126 - JDB

drop table if exists `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_Eventdates`;

drop table if exists `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_EventdatesValid`;

truncate table `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.observation\_period`;

create table `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_Eventdates`

as

select cast(a.person\_id as int64) as person\_id

,a.DateEvent as EventDate

from `yhcr-prd-phm-bia-core.CY\_MYSPACE\_RSBO.src\_BDCT\_SRCode\_V2JDB` a

,`yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.person` e

where e.person\_id = cast(a.person\_id as int64)

#The following two sections apply to using another source event table.

#Following Section takes the Start date of the hospital spell and pushes it into the valid dates table

insert into `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_Eventdates`

select cast(person\_id as int64) as person\_id,

cast(substr(Start\_Date\_Hospital\_Provider\_Spell,1,4) ||'-'|| substr(Start\_Date\_Hospital\_Provider\_Spell,5,2) ||'-'|| substr(Start\_Date\_Hospital\_Provider\_Spell,7,2) as date) as EventDate

from `yhcr-prd-phm-bia-core.CY\_STAGING\_DATABASE.src\_BDCT\_CDS130InpatientFCEdata010419\_310320Final\_V3`

where IF(LENGTH(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(person\_id,'1',''),'2',''),'3',''),'4',''),'5',''),'6',''),'7',''),'8',''),'9',''),'0',''),'.',''))=0,'A number','Not a number') = 'A number'

#Following Section takes the Discharge date of the hospital spell and pushes it into the valid dates table

insert into `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_Eventdates`

select cast(person\_id as int64) as person\_id,

cast(substr(Discharge\_Date\_From\_Hospital\_Provider\_Spell,1,4) ||'-'|| substr(Discharge\_Date\_From\_Hospital\_Provider\_Spell,5,2) ||'-'|| substr(Discharge\_Date\_From\_Hospital\_Provider\_Spell,7,2) as date) as EventDate

from `yhcr-prd-phm-bia-core.CY\_STAGING\_DATABASE.src\_BDCT\_CDS130InpatientFCEdata010419\_310320Final\_V3`

where IF(LENGTH(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(person\_id,'1',''),'2',''),'3',''),'4',''),'5',''),'6',''),'7',''),'8',''),'9',''),'0',''),'.',''))=0,'A number','Not a number') = 'A number'

and Discharge\_Date\_From\_Hospital\_Provider\_Spell not like '%NULL%'

#End of section relating to additional date sources.

#add in more of these if you have 3 or more event bases tables that are bing included in the FDM

#-- Second part limits these to the Valid dates for this dataset

create table `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_EventdatesValid`

as

select a.person\_id

, a.EventDate

from `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_Eventdates` a

,`yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.person` e

where e.person\_id = a.person\_id

and e.death\_datetime is not null

and a.EventDate >= e.birth\_datetime

and a.EventDate <= date\_add(e.death\_datetime, INTERVAL 42 day)

and a.EventDate <= (Select extract\_date from `yhcr-prd-phm-bia-core.CY\_LOOKUPS.tbl\_Dataset\_ExtractDateRef` where DataSetName = 'BDCT' );

#-- BUT this only includes persons with a death datetime

#-- So section below includes those with a null deathdatetime

insert into `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_EventdatesValid`

select distinct a.person\_id

, a.EventDate

from `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_Eventdates` a

,`yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.person` e

where e.person\_id = a.person\_id

and e.death\_datetime is null

and a.EventDate >= e.birth\_datetime

and a.EventDate <= (Select extract\_date from `yhcr-prd-phm-bia-core.CY\_LOOKUPS.tbl\_Dataset\_ExtractDateRef` where DataSetName = 'BDCT' )

#third part pushes these into the observation\_period table

insert into `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.observation\_period`

select distinct

ROW\_NUMBER() over (Order by person\_id) as observation\_period\_id

, person\_id

, min(EventDate) as observation\_period\_start\_date

, max(EventDate) as observation\_period\_end\_date

,null as period\_type\_concept\_id

from `yhcr-prd-phm-bia-core.CY\_FDM\_BDCT.tmp\_EventdatesValid`

group by person\_id

## visit\_occurence - notes

The script below can be used to create a blank visit\_occurrence table

create table `yhcr-prd-phm-bia-core.*your FDM dataset here* .visit\_occurrence`

(

visit\_occurrence\_id INT64 not null ,

person\_id INT64 not null ,

visit\_concept\_id INT64 not null ,

visit\_start\_DATE DATE not null ,

visit\_start\_DATETIME DATETIME ,

visit\_end\_DATE DATE not null ,

visit\_end\_DATETIME DATETIME ,

visit\_type\_concept\_id INT64 not null ,

provider\_id INT64 ,

care\_site\_id INT64 ,

visit\_source\_value STRING ,

visit\_source\_concept\_id INT64 ,

admitting\_source\_concept\_id INT64 ,

admitting\_source\_value STRING ,

discharge\_to\_concept\_id INT64 ,

discharge\_to\_source\_value STRING ,

preceding\_visit\_occurrence\_id INT64

)

;

## Populating the visit\_occurence table:

|  |  |  |
| --- | --- | --- |
| visit\_occurrence\_id | required | Generate with FDM prefix |
| person\_id | required |  |
| visit\_concept\_id | required | Take concept\_id from:  select \* from concept where domain\_id = ‘Visit’ and standard\_concept = ‘S’ and invalid\_reason is null;  If there’s no available information to select a visit\_concept\_id, set to zero and see visit\_type\_concept\_id below for how to indicate this. |
| visit\_start\_date | required |  |
| visit\_start\_datetime | required | Set to null if no time available |
| visit\_end\_date | required | Set to visit\_start\_date if missing |
| visit\_end\_datetime | required | Set to null if no time available |
| visit\_type\_concept\_id | desirable | Take concept\_type\_id from:  select \* from concept where domain\_id = ‘Visit Type’ and standard\_concept = ‘S’ and invalid\_reason is null;  Can be set to zero if not required.  If visit\_concept\_id is missing due to the visit being derived from clinical/observational data, set to 44818518 (meaning “Visit derived from EHR record”) |
| provider\_id | desirable |  |
| care\_site\_id | required | Will often be one care site per FDM but may be more so should be populated |
| visit\_source\_value | desirable |  |
| admitting\_source\_concept\_id | desirable | May be especially important for secondary care data |
| admitting\_source\_value | desirable | May be especially important for secondary care data |
| discharge\_to\_source\_id | desirable | May be especially important for secondary care data |
| discharge\_to\_source\_value | desirable | May be especially important for secondary care data |

## FDM review checklist

Separate document link to be inserted here