Regional Cooperation for
Limited Area Modeling in Central Europe

LACE
nwp central europe

ODB & MANDALAY

Alena Trojáková

DHMZ SHMU ARSO METEO Slovenia OMSZ ZAMG ROMANIA ANM

ODB

LACE
nwp central europe

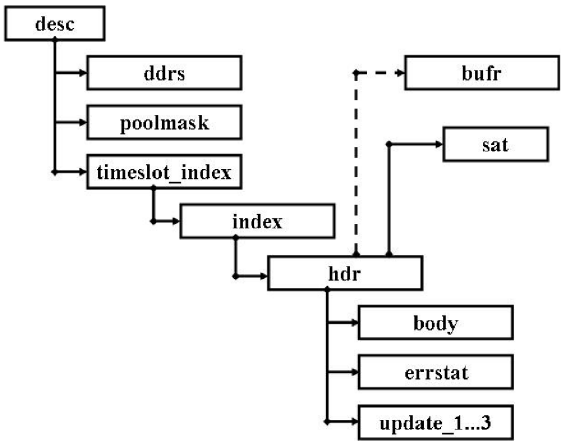
- **Observational DataBase** (ODB) is hierarchical database software developed at ECMWF to manage very large observational data volumes
- ODB components:
  - **ODB/DDL** Data Definition Language (flexible data layout definition of database)
  - **ODB/SQL** query language (fast data retrieval)
  - **ODB Fortran90 interface** layer (data manipulation as create, update and remove, execution of sql-queries and retrieval of data, control of MPI and/or OpenMP-parallelization)
- ODB content:
  - observation identification information (date, position, station ID)
  - observed values
  - various flags indicating quality and validity of an observation (active,
  - departure from observed value (obs-guess, obs-analysis)
  - bias corrections, satellite specific information like zenith angle, field of view, ...
  - other important observational processing and meteorological information

ODB

- **ODB structure**

  - basic building blocks called table (can be seen as a matrice (2D-array)) with a number of rows and columns containing numerical data (example hdr: general information of one report (date, time, station ID))

- **data are organized into a *tree-like* structure**



- **structure allows ”repeating” information using parent/child relationship: each parent can have many children but each child only has one parent**

ODB/DDL - Data Definition Layout

LACE
nwp central europe

- DDL file defines the structure (hierarchy)
- ASCII file
- consists of uniquely named TABLEs
- tables are made up of uniquely named COLUMNs (or attributes)
  notation: `column_name@table_name`
- each COLUMN has a specific type
  - integer/real/string
  - packed
  - YYYYMMDD, HHMMSS (storage of date)
  - bitfield type (maximum 32 one-bit members per type,
    notation: `column_name.bitfield_name@table_name`
  - @LINK to define connections between TABLEs

```
CREATE TABLE table_name AS ()
  column_name1 data_type1,
  column_name2 data_type2,
  ...
```

ODB - relation between tables

LACE
nwp central europe

CREATE TABLE hdr AS (
lat real,
lon real,
statid string,
obstype int
date YYYYMMDD
body @LINK
);

CREATE TABLE body AS (
varno pk5int,
obsvalue pk9real,
);

lat	lon	statid	obstype	date
50.4	15.5	11518	1	20100913

...

lat	lon	statid	obstype	date
50.4	14.5	11520	1	20100913

lat	lon	statid	obstype	date
52.4	16.5	11582	1	20100913

...

body@LINK

(offset=j, length=3)

varno	obsvalue
39	297.5
41	6.0
58	0.92

...

varno	obsvalue
39	298.5
41	5.0
58	0.87

...

ODB/SQL - data retrieval

LACE nwp central europe

- **data extraction by query language ODB/SQL via so-called views**

  [CREATE VIEW view_name AS]
  SELECT [DISTINCT] column_name (s)
  FROM table(s)
  WHERE cond ORDERBY sort_column_name(s) [ASC/DESC]

can be used in an interactive way via ODB-tools (odbsql,...)

Examples:

- **find distinct values of obstype and sort them DESCending**

```
select distinct obstype from hdr orderby obstype desc
```

- **vertical profile of MEAN and STD for O-G for sensor HIRS**

```
select count(*), satid,obstype,varno,sensor,press,avg(fg_depar),stdev(fg_depar)
from hdr,body,sat
where obsvalue is not NULL and status.active@body = 1 and sensor = 0
```

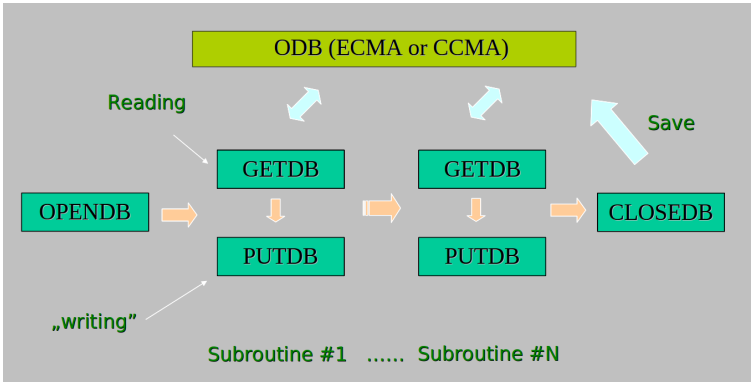
- **find location and values of all active SYNOP observations**

```
select lat,lon,obsvalue from hdr,body where obstype = 1 and status.active@body
```

ODB Fortran90 interface

LACE
nwp central europe

- layer to provide database access to:
  - open & close database
  - attach to & execute
    precompiled ODB/SQL queries
  - load, update & store queried data
  - inquire information about metadata



- allow use MPI
- selected data can be asked to be "partly-exchanged" across processors; but default data selection applies to the local pools only
- each query need to be pre-compiled/linked with the main user program
- each ARPEGE/IFS cycle has its own ODB version !

Practical aspects

Practical aspects

LACE
nwp central europe

- ODB usage in ARPEGE/ALADIN:
  - ALDODB - master for configuration 002,131,701
  - BATOR - master for ODB creation
  - ODBTOOLS - master for ODB manipulation
  - MANDALAY - master for ODB conversion to ASCII
- each query need to be pre-compiled/linked with the main user program
- each ARPEGE/IFS cycle has its own ODB version !
- ODB content:
  - observation identification information (date, position, station ID)
  - observed values
  - various flags indicating quality and validity of an observation (active,
  - departure from observed value (obs-guess, obs-analysis)
  - bias corrections, satellite specific information like zenith angle, field of view, ...
  - other important observational processing and meteorological information

ODB browsing

LACE
nwp central europe

- **odbsql - ”dynamic” retrieval based**
  - **compilation is done on the fly**
  - **available in an ODB-standalone package only**

```
odbsql  -q 'select obstype,statid,lat,lon,varno from hdr,body '
```

- **mandalay - ”static” retrieval based**
  - **retrieval are based on predefined and user defined views (mandalay.sql):**

```
CREATE VIEW mandalay AS
SELECT
  obstype,statid,lat,lon,varno
FROM  hdr,body
```

  - **in case of change recompilation is needed (or a wrapper for re-compilation)**
  - **suitable for oper. application or frequently used request (observational monitoring,...)**
  - **export VERSION=1**
  - **export DEGRE=1**

```
mpirun -np 1 ./MANDLAY CMAFILE
```

ODB in ARPEGE/ALADIN

LACE
nwp central europe

- **OPENDB - opens ECMA/CCMA databases**
- **GETDB**
  - execute one or more SQL queries (as defined in ctxinitdb.F90
  - calls ODB_select, allocates matrices ROBHDR,ROBODY,...
  - then calls ODB_get to fill out the observational matrices
    - ROBHDR: index & hdr - tables related data
    - ROBODY: body, errstat, update,.. - tables related data
    - MLNKH2B: coupling between ROBHDR & ROBODY

```
HDR_LOOP: do jobs=1, NROWS_ROBHDR
  ROBHDR(jobs,MDBLAT) = <some_thing>
  BODY_LOOP: do jbody= MLNKH2B(jobs), MLNKH2B(jobs+1) - 1
      if ( ROBODY(jbody,MDBVNM) == <varno> ) then
          ROBODY(jbody, MDBOMF) = <some_thing>
      endif
    enddo BODY_LOOP
enddo HDR_LOOP
```

- **PUTDB**
  -returns the contents of the updated matrices bask to (in-memory) database data structures via routine ctxputdb.F90
  - calls ODB_put, deallocates matrices and calls ODB_cancel
- **CLOSEDB - closes ECMA/CCMA databases**

ODB in ARPEGE/ALADIN

LACE nwp central europe

- **correspondence of ODB/SQL and ARPEGE/IFS variables:**

```
INTEGER(KIND=JPIM) :: mdbdat ! 'date@hdr'
INTEGER(KIND=JPIM) :: mdbrfl ! 'report_rdbflag@hdr'
INTEGER(KIND=JPIM) :: mdbrst ! 'report_status@hdr'
INTEGER(KIND=JPIM) :: mdbrev1 ! 'report_event1@hdr'
INTEGER(KIND=JPIM) :: mdbrble ! 'report_blacklist@hdr'
INTEGER(KIND=JPIM) :: mdbsid ! 'statid@hdr'
INTEGER(KIND=JPIM) :: mdblat ! 'lat@hdr'
INTEGER(KIND=JPIM) :: mdblon ! 'lon@hdr'
INTEGER(KIND=JPIM) :: mdbalt ! 'stalt@hdr'
...
INTEGER(KIND=JPIM) :: mdbvnm ! 'varno@body'
INTEGER(KIND=JPIM) :: mdbvar ! 'obsvalue@body'
INTEGER(KIND=JPIM) :: mdbomn ! 'an_depar@body'
INTEGER(KIND=JPIM) :: mdbomf ! 'fg_depar@body'
INTEGER(KIND=JPIM) :: mdbflg ! 'datum_anflag@body'
...
```

- **for complete definitions see ./arpifs/common/yomdb_vars.h**

ODB flags used by CANARI

LACE
nwp central europe

- **Taillefer (2002): CANARI technical documentation**
  https://www.umr-cnrm.fr/gmapdoc/IMG/ps/canari_doc_cy25t1.ps
- **datum_anflag@body is coded over 29bits**

```
bits 1 to 4 : final quality code
bits 5 to 8 : first-guess quality code
bits 9 to 12 : spatial quality control code
bits 13 to 16 : variational quality code (not used in CANARI)
bits 17 to 20 : blacklist code
bit 21 : if set to 1, parameter used in the surface pressure analysis
bit 22 : if set to 1, parameter used in the wind and temperature analysis
bit 23 : if set to 1, parameter used in the relative humidity analysis
bit 24 : if set to 1, parameter used in the 2 meters temperature analysis
bit 25 : if set to 1, parameter used in the 2 meters relative humidity analysis
bit 26 : if set to 1, parameter used in the 10 meters wind analysis
bit 27 : if set to 1, parameter used in the precipitations analysis (not coded yet
bit 28 : if set to 1, parameter used in the snow analysis
bit 29 : if set to 1, parameter used in the SST analysis
```

```
odbsql  -q 'select statid,datum_anflag.ut2@body from hdr,body where varno == 39 '
```

ODB flags used by CANARI

- **datum_rdbflag@body coded over 30 bits, the first half concerns the quality of the vertical coordinate and the second half the quality of the parameter itself.**

```
bit 1 0 no human control
      1 human control
bit 2 0 no correction by the meteorological databank preprocessing
      1 correction by the meteorological databank preprocessing
...
bits 7 and 8 0 correct parameter versus previous analysis
             1 probably correct parameter versus previous analysis
             2 probably incorrect parameter versus previous analysis
             3 incorrect parameter versus previous analysis
bit 9 0 parameter no used by the previous analysis
      1 parameter used by the previous analysis
```

- **for complete definitions of bits see ./odb/ddl/type_definitions.h**

```
odbsql  -q 'select statid,"datum_rdbflag.*@body" from hdr,body where varno == 39 '
```

Exercises

- find which observation types, variables and obs values are in your ECMA
- find station locations used in CANARI T2m analysis
- find/plot first guess,analysis departures of all active SYNOP observations

End

LACE
nwp central europe

Thank you for your attention !

DHMZ SHMU ARSO METEO Slovenia OMSZ ZAMG ROMANIA ANM

Exercises

- **find which observation types, variables and obs values are in your ECMA**
  `select obstype,varno,obsvalue from hdr, body`
- **find station locations used in CANARI T2m analysis**
  `select lat,lon,obsvalue from hdr,body where datum_anflag.ut2@body==1`
- **find/plot first guess,analysis departures of all active SYNOP observations**
  `select lon,lat,fg_depar,an_depar from hdr,body where datum_anflag.ut2@body==1`