

Time Series Database Interface

UseR 2008

Paul Gilbert

Bank of Canada

`pgilbert@bank-banque-canada.ca`

August, 2008

The views expressed are those of the authors.
No responsibility for them should be
attributed to the Bank of Canada.

Motivation:

Provide an abstract layer (API)
for time series database calls
(and back-end data base structures)

- What is a time series?
- Query is (usually) really simple.

Why an API layer?

- Other code can use multiple/different databases.
- Independence from database/vendor.
- Independence from database versions.
- Independence from organization.

Why SQL?

- (not just SQL)
- DBI, network interface, widely used and supported
- Do extra things like panels and vintages.
- (I get to learn something about SQL)

- API (roughly defined)
- Applications: R, ...
- Interface: TSdbi (using DBI)
- Database: PADI, MySQL, SQLite, ...
SQL, Fame, ..., getHistoricalQuote
- (using TSpadi, TSMYSQL, TSQLite, ...)
- (and RMySQL, RSQLite, ...)

- Historical note: PADI is old

```
TSconnect(drv, dbname, ...)
```

```
TSget(serIDs,
```

```
    con=options()$TSconnection, ...)
```

- for SQL dbs this calls a standard SQL function, TSgetSQL.
- in theory the time representation can be specified.

```
TSput(x, serIDs=seriesNames(x),  
      con=options()$TSconnection, ...)  
TSdates(serIDs,  
        con=options()$TSconnection, ...)  
TSdescription(x,  
              con=options()$TSconnection, ...)  
TSdescription assignment function too.  
TSexists  
TSdoc
```


SQL implementation

- tables for each "frequency" of data
- annual, quarterly, monthly, semiannual, weekly, daily, business day, minutely, irregular data with a date, and irregular data with a date and time
- A Q M S W D B I T U
- Meta (documentation and to lookup what table a series is on)
- vintages, panels (optional)

Example table setup

```
dbGetQuery(con, "create table D (  
  id          VARCHAR(40),  
  date        DATE,  
  period      INT,  
  v          double DEFAULT NULL  
);")
```

```
dbGetQuery(con, "CREATE INDEX Dindex_id  
  ON D (id);")
```

```
dbGetQuery(con, "CREATE INDEX Dindex_date  
  ON D (date);")
```

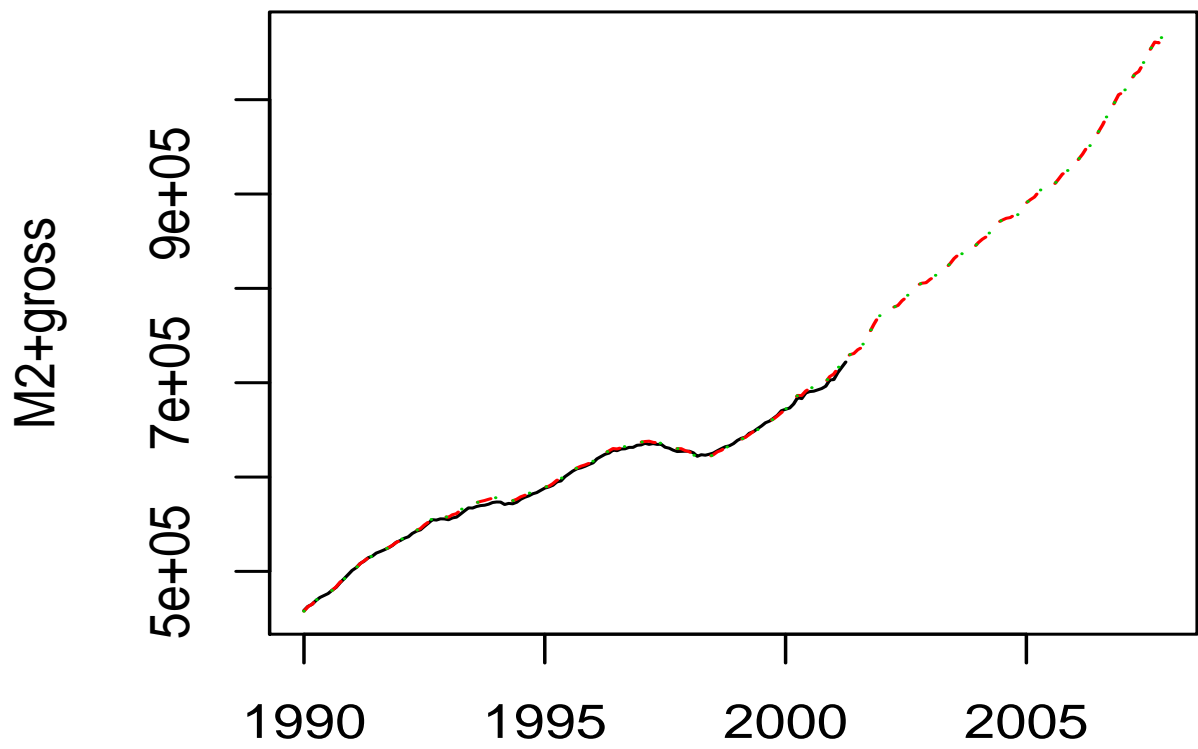
```
dbGetQuery(con, "CREATE INDEX Dindex_period  
  ON D (period);")
```

Example

(skipping details about setting up tables,
permissions, loading data, etc)

```
require("TSMYSQL")
m <- dbDriver("MySQL")
con <- TSconnect("MySQL", dbname="FVvintages")
  # pass user/passwd/host in ~/.my.cnf
z1 <- TSget(serIDs="M2+gross", con=con,
           vintage="v2001-07")
z2 <- TSget(serIDs="M2+gross", con=con,
           vintage="current")
z3 <- TSget(serIDs="M2+gross", con)
  # should default to current
tfplot(z1, z2,z3,
       Title="Selected vintages of gross M2+",
       start=c(1990,1))
```

Selected vintages of gross M2+



- performance?
- other back-ends?
- extensions? start, end, manipulations?

The End

-
-
-