

# Time Series Database Interface: R Padi to Fame (TSpadi)

May 7, 2008

## 1 Introduction

The code from the vignette that generates this guide can be loaded into an editor with `edit(vignette("TSpadi"))`. This uses the default editor, which can be changed using `options()`. It should be possible to view the pdf version of the guide for this package with `print(vignette("TSpadi"))`.

Once R is started, the functions in this package are made available with

```
> library("TSpadi")
```

This will also load required packages *TSdbi*, *DBI*, *methods*, and *tframe*. Some examples below also require *zoo*, and *tseries*.

The main application of TSpadi is to set up an interface to a server which extracts data from a database and supplies it to R programs. In these examples, Fame is the backend database.

*TSpadi* does not support writing data to a database, so examples below are restricted to an already available database. It should be possible to write a TSdbi package directly on top of the R package *fame*, to standardize the API and use more complete Fame functionality, but that has not yet been done.

### 1.1 Examples Using TSdbi with ets

These examples use a database called "ets" which is available at the Bank of Canada. This set of examples illustrates how the programs might be used if a larger database is available. Typically a large database would be installed using database scripts directly rather than from R with *TSput* or *TSreplace*.

The following are wrapped in `if (!inherits(conets, "try-error"))` so that the vignette will build even when the database is not available. This seems to require an explicit call to `print()`, but that is not usually needed to display results below. Another artifact of this is that results printed in the if block do not display until the end of the block. Also, graphics are not displayed in the vignette because, when they are not generated, the missing file causes an error building the vignette.

```

> m <- dbDriver("padi")
> conets <- try(TSconnect(m, dbname = "ets"))
> if (!inherits(conets, "try-error")) {
  options(TSconnection = conets)
  TSdoc("M.SDR.CCUSMA02.ST")
  print(TSmeta("M.SDR.CCUSMA02.ST"))
  z <- TSget("M.SDR.CCUSMA02.ST")
  EXCH.IDs <- t(matrix(c("M.SDR.CCUSMA02.ST", "SDR/USD exchange rate",
    "M.CAN.CCUSMA02.ST", "CAN/USD exchange rate", "M.MEX.CCUSMA02.ST",
    "MEX/USD exchange rate", "M.JPN.CCUSMA02.ST", "JPN/USD exchange rate",
    "M.EMU.CCUSMA02.ST", "Euro/USD exchange rate", "M.OTO.CCUSMA02.ST",
    "OECD /USD exchange rate", "M.G7M.CCUSMA02.ST", "G7 /USD exchange rate",
    "M.E15.CCUSMA02.ST", "Euro 15. /USD exchange rate"),
    2, 8))
  print(TSdates(EXCH.IDs[, 1]))
  z <- TSdates(EXCH.IDs[, 1])
  print(start(z))
  print(end(z))
  tfplot(TSget(serIDs = "V122646", conets))
}

> if (!inherits(conets, "try-error")) {
  print(TSdescription(TSget("V122646", TSdescription = TRUE)))
  print(TSdoc(TSget("V122646", TSdoc = TRUE)))
  tfplot(TSget("V122646", names = "V122646", conets))
}

> if (!inherits(conets, "try-error")) {
  z <- TSget("V122646", TSdescription = TRUE)
  tfplot(z, Title = strsplit(TSdescription(z), ", "))
}

> if (!inherits(conets, "try-error")) {
  ETSgdp <- annualizedGrowth(aggregate(TSget("V1992067"), nfrequency = 4,
    FUN = mean), lag = 4, names = "GDP Y/Y Growth")
  tfplot(ETSgdp)
}

```

## 2 TS PADI Server

Building a padi server will require some programming effort. The padi interface is getting to be fairly old and, although it still works, some of the underlying code should probably be replaced with a newer approach, perhaps based on something like SOAP.

Code and a description of a prototype of a standard for a Time Series Protocol for Application - Database Interface (TS PADI) is still available at

<http://www.bank-banque-canada.ca/pgilbert>. The code includes a working interface to a Fame database.