

RESSTE Workshop

Working on data from PREV'AIR

with spacetime & R

Maxime Beauchamp, Laure Malherbe, INERIS
Nicolas Saby, INfoSol, Orléans,
Eric Parent, Liliane Bel, AgroParisTech/INRA, Paris,
Denis Allard, Thomas Opitz, Edith Gabriel, INRA, Avignon

Lancaster Spatial Statistics workshop, the 4th of July 2017

Predicting Air Pollutants

– What:

- Pollutants levels with legal thresholds: O_3 , NO_2 , PM_{10} , $PM_{2.5}$
- Global / european / national scale
- 3 days ahead

Public warnings should not only depend on observations.

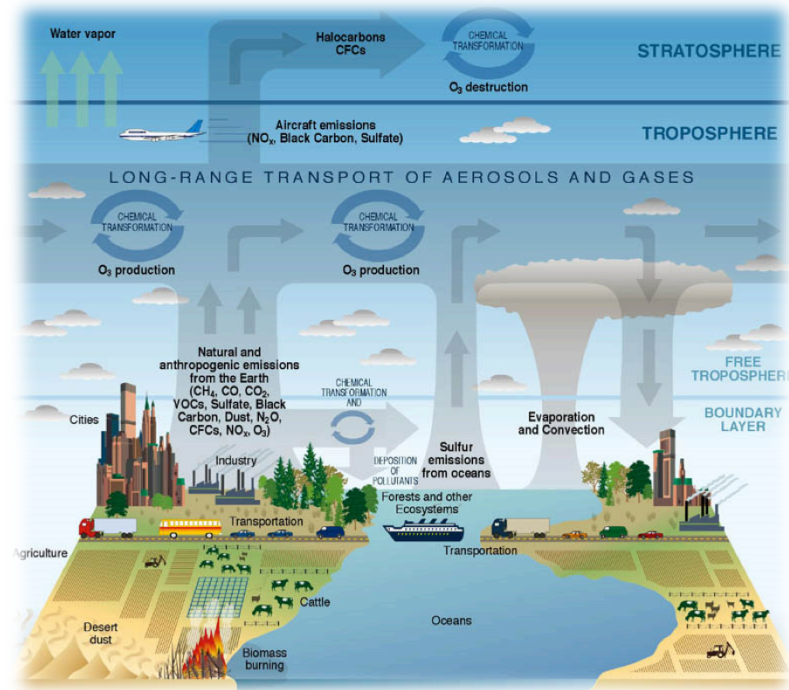
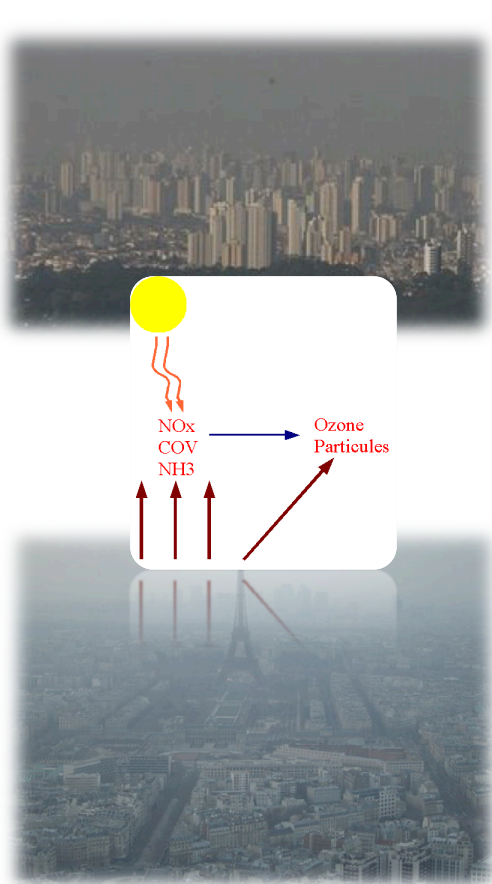
– Where?

- Places with few or no observation

– Why?

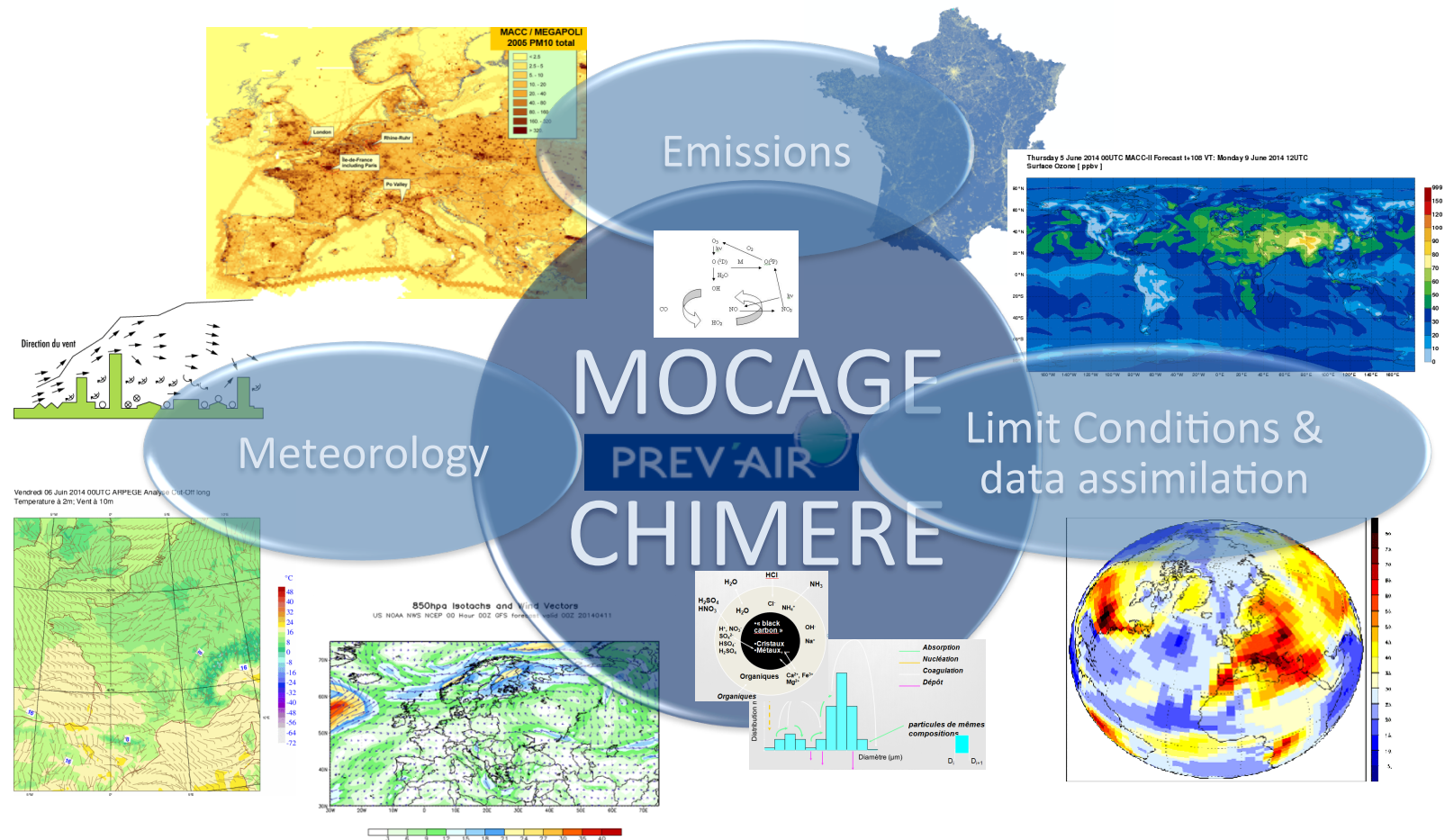
- **Air quality** management in case of polluted periods.
- Sources of pollution

PREV'AIR: How does it work?



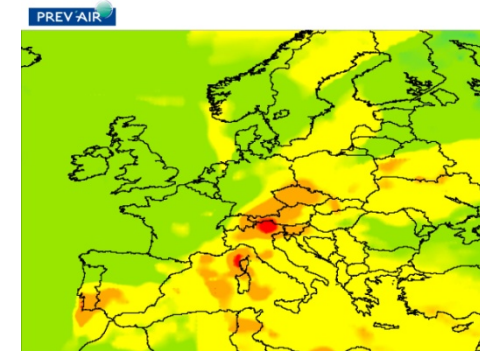
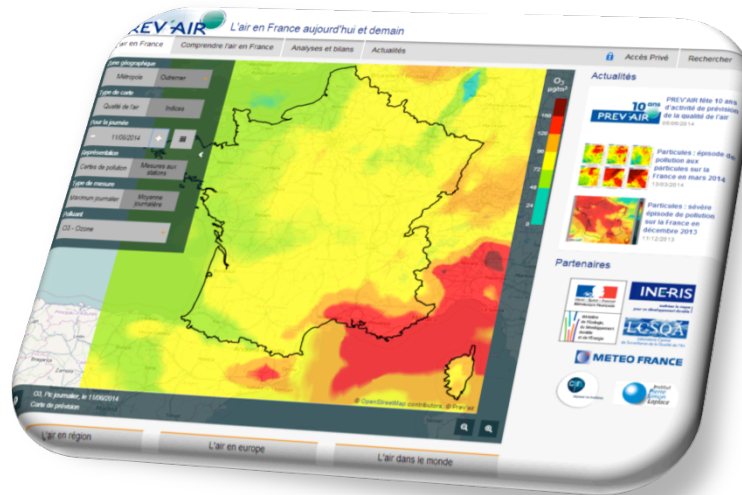
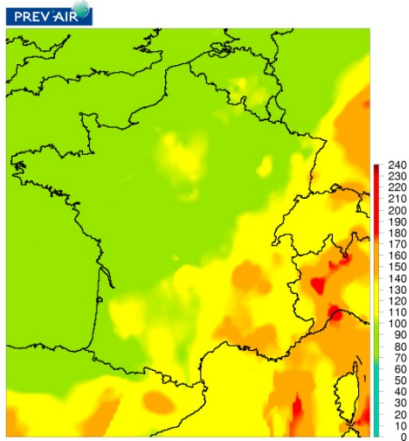
Mathematical models mimicking the dynamics of transport and chemical interactions of chemical pollutants within the troposphere

How does it work?



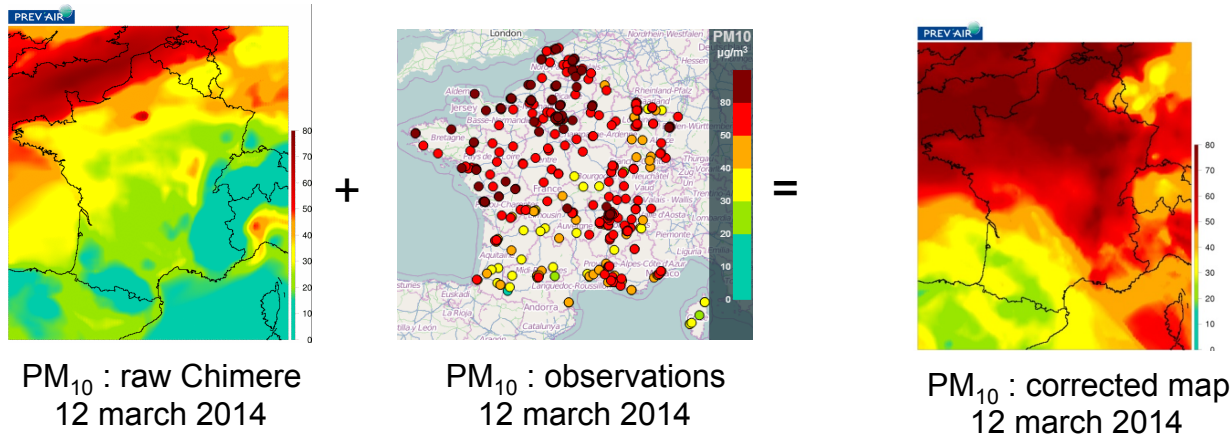
PREV'AIR website

- Every morning, 8 o'clock
 - Daily values (max and mean) for :
 - yesterday, today, d+1, d+2
 - O_3 , NO_2 , Particles (PM_{10} , $PM_{2.5}$ & desert dust)



What is PREV'AIR used for?

- Statistical analyses



- Many end-users :
 - nat/int. research projects,
 - air quality authorities,
 - Medias...

Your data for the RESSTE workshop

- Repertories
 - A CHM repertory that contains data files (daily and hourly) from CHIMERE model for year 2014 (50 km resolution).
 - An OBS repertory containing hourly and daily data from the European database Airbase.
 - One elementary file= (one pollutant, one time, all working locations)
 - A metadata file giving for each station its code, long/lat, etc...
- Focus on France... (for once!)

DATA.TABLE for building dataframes from the raw data

- 1.7 Go of data available from RENATER
- OBS & CHM : Many « rbind »'s, the dataframe gets bigger & bigger; merging large dataframes, work with missing data
- data.table package with powerful « rbindlist ».

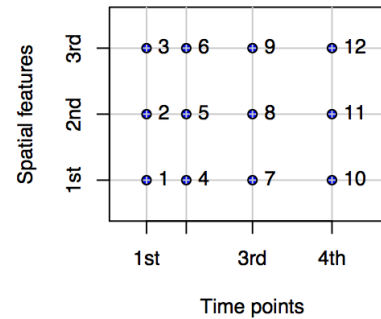
Let's start...

- data/OBS_horaire.Rdata: Dataframe of hourly records over 505 French stations (ID), long, lat, date, PM10, PM25, NO₂, O₃
- data/OBS_jour.Rdata : Dataframe of daily records over 505 French stations
- data/CHM.Rdata : 111x101 gridded values over France
- data/stations.Rdata : Station characteristics

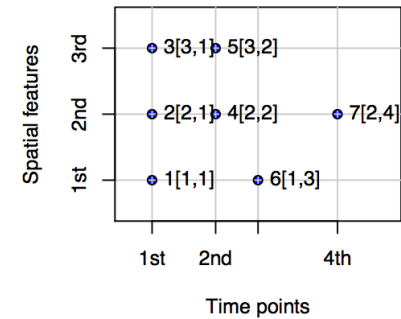
ST formats

- Time-wide format :
row= location,
columns=dates
- Space-wide format:
row= date,
columns=locations
- Long format:
row= record,
columns=location &
date

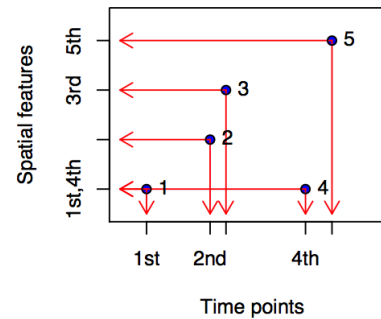
STF: full grid layout



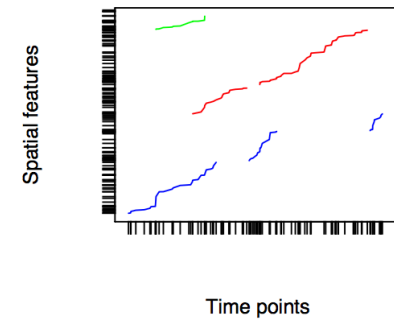
STS: sparse grid layout



STI: irregular layout



STT: trajectory



stConstruct

`(x, space, time, SpatialObj = NULL, TimeObj = NULL,
crs = CRS(as.character(NA)), interval, endTime)`

- `X`: object of class `matrix` or `data.frame`, holding the long, space-wide or time-wide table
- `Space`: in case `x` is a long table, character or integer holding the column index in `x` where the spatial coordinates are (if `length(space)==2`) or where the ID of the spatial location is (if `length(space)==1`). If `x` is a space-wide table, a list with each (named) list element a set of columns that together form a variable
- `Time`: in case `x` is a long table, character or integer indicating the column in `x` with times;
- `SpatialObj`: object of class `Spatial-class`, containing the locations of a time-wide table, or the locations of a long table
- `TimeObj`: in case of space-wide table, object of class `xts`, containing the times for each of the columns in a list element of `space`
- `Crs`: object of class `CRS-class`; only used when coordinates are in `x` and no CRS can be taken from `SpatialObj`

stConstruct to build STIDF, STFDF & STSDF

```
STFDF_day <- stConstruct(OBS_jour,  
space=c('long','lat'),  
time='date',  
SpatialObj=SpatialPoints(OBS_jour[,c('long','lat')]))
```

#23 Mbytes

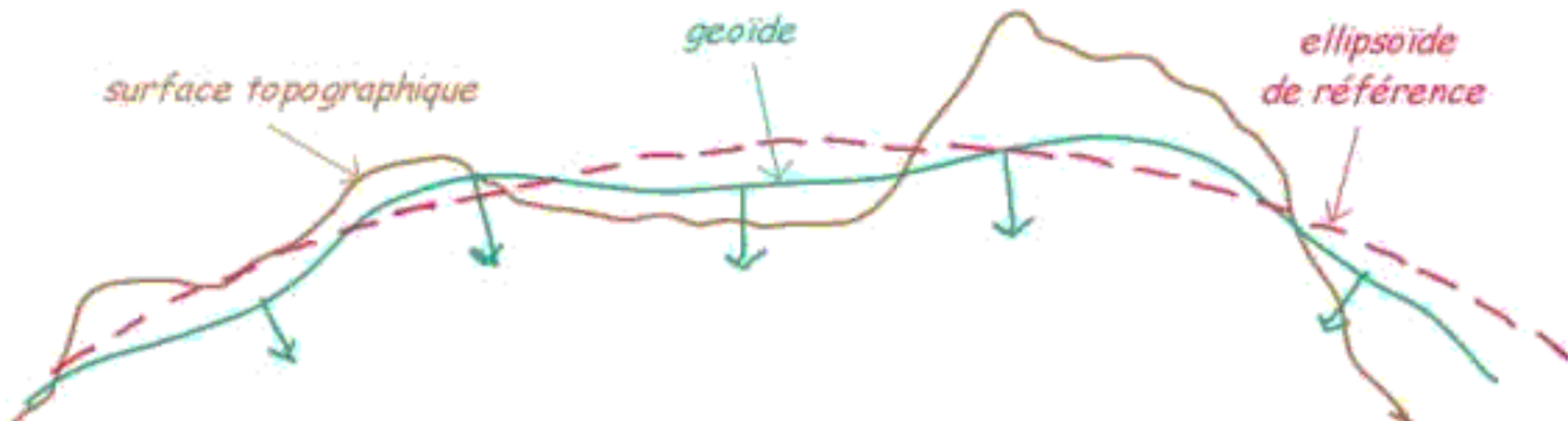
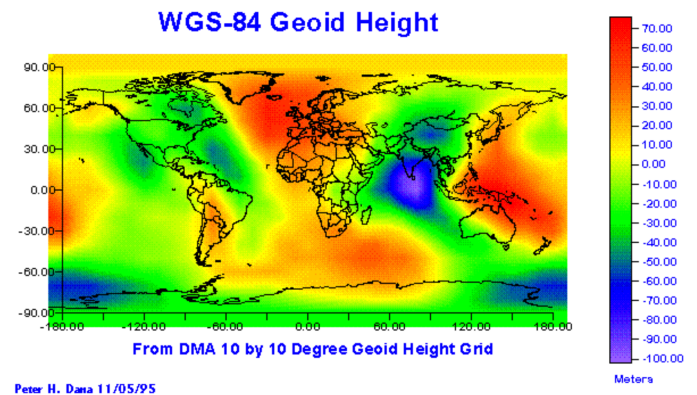
```
STFDF_day <- as(STFDF_day, "STFDF") #6 Mbytes
```

Coordinates Reference System

- `proj4string(CHM_day) <- "+proj=longlat
+ellps=WGS84 +datum=WGS84 +no_defs"`
- ???
- `rgdal` : geospatial abstract library
 1. Projection `projInfo(type='proj')`
 2. Datum `projInfo(type='datum')`
 3. Ellipsoid `projInfo(type='ellps')`

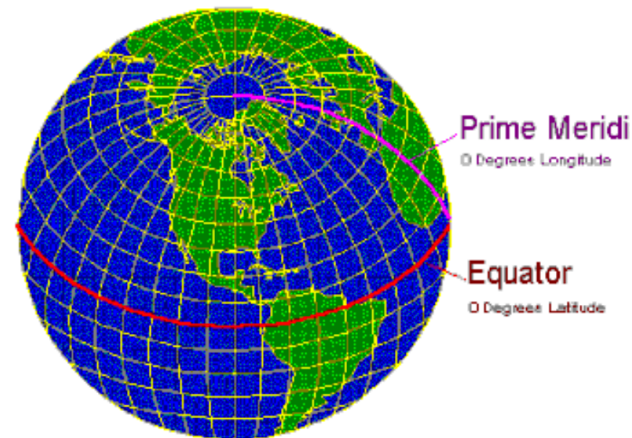
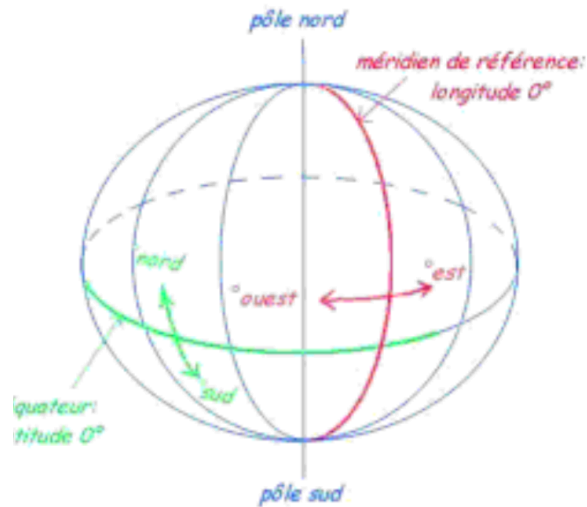
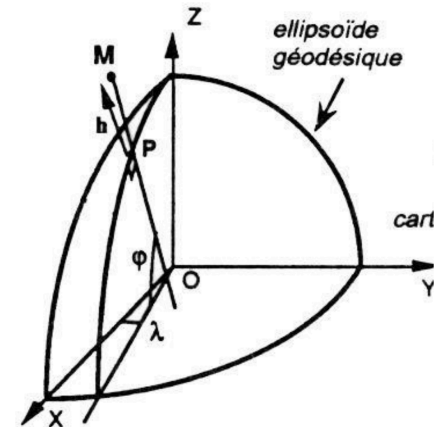
Datum

- Topological terrestrial surface
- Reference Geoid
- Mathematical Ellipsoid



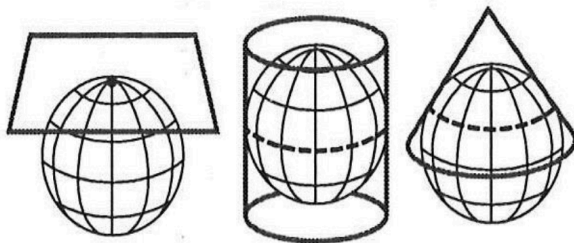
« Natural » polar coordinates

latlong: unprojecting

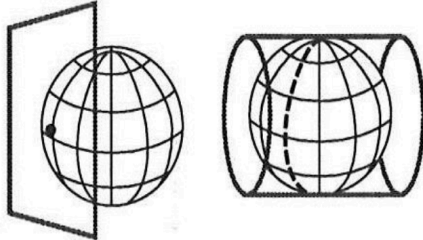


WGS84 (Google Earth)

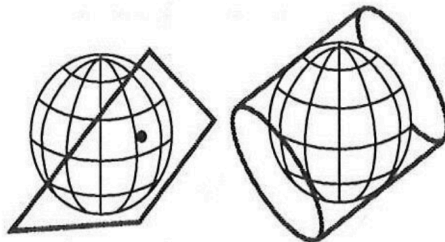
Projecting



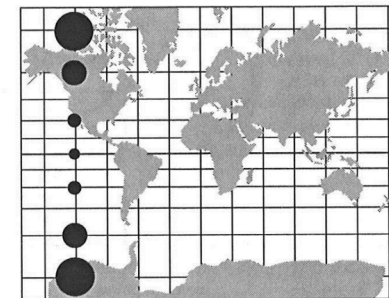
La projection est **directe** :
la surface de projection est centrée sur un pôle (projection azimutale), sur l'équateur (projection cylindrique) ou sur un parallèle (projection conique).



La projection est **transverse** :
la surface de projection est centrée sur un point de l'équateur (projection azimutale) ou sur un méridien (projection cylindrique).



La projection est **oblique** :
la surface de projection est centrée sur un point quelconque de la sphère.

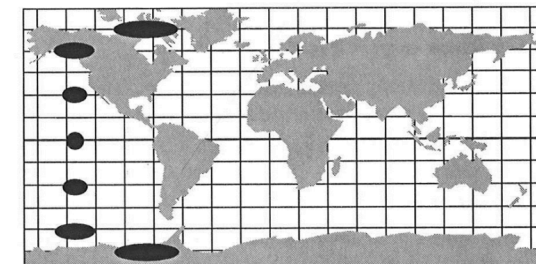
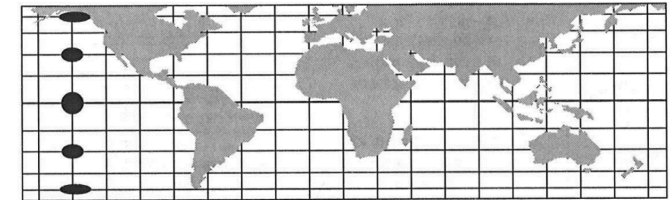


Projection conforme
L'indicatrice reste un cercle mais sa surface varie

Les angles, donc les formes sont préservées

Projection équivalente
L'indicatrice s'aplatit, sa surface reste constante

Les superficies sont préservées



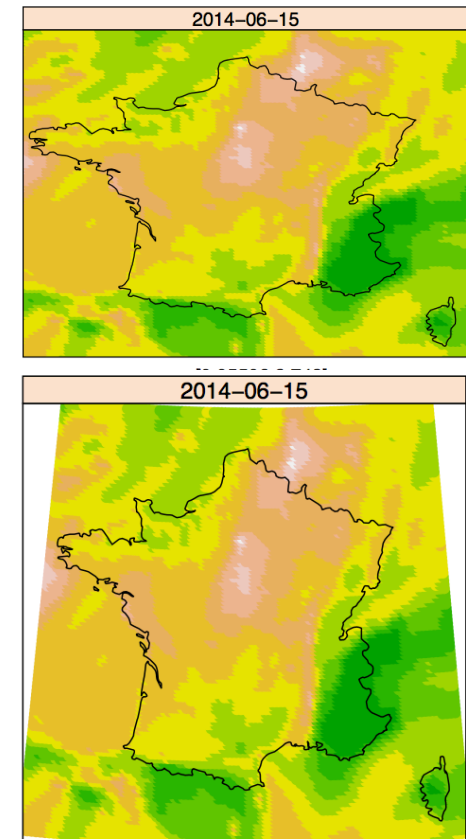
Projection aphylactique
L'indicatrice devient une ellipse et sa taille varie

Compromis

European Petroleum Survey Group codes

All-in-one encoding using « weird » tags

- epsg:4326 (Google unprojected)
- epsg:4269 (most US federal agencies)
- epsg:2154 (French Lambert 93)
- To retrieve CRS: `proj4string(xsp)`
- To assign CRS: `proj4string(xsp)=CRS('+init=epsg:2154')`
- To retrieve CRS: `new=spTransform(old,CRS('+init=epsg:2154'))`



***spacetime**: Spatio-Temporal Data in R*

E Pebezma J Stat Softw. Nov. 2012, 51(7).

method	what it does
stConstruct	Creates STFDF or STIDF objects from single or multiple tables
[[, \$, \$<- [Select or replace data values Select spatial and/or temporal subsets, and/or data variables
as	coerce to other spatio-temporal objects, xts , Spatial , matrix , or data.frame
stplot	create spatio-temporal plots, see Section 5
over	overlay: retrieve index or data values of one object at the locations and times of another
aggregate	aggregate data values over particular spatial, temporal, or spatio-temporal domains

Table 1: Methods for spatio-temporal data in package **spacetime**.

Plotting ST objects

- Ordinary base plots using plot, image, etc.
- Lattice (treillis) plots using spplot
- ggplot2 for nice graphs of time series
- New capabilities of **stplot**
 1. multi-panel plots
 2. space-time plots (Hovmöller diagrams)
 3. Animated plots
 4. Time series plots

Practical: Do it yourself

- Find the AirBase stations located in Paris (av des champs Elisées) , in Lyon (St Just) , in Rennes (les Halles)
- Make a daily time series plot of PM10 & NO2 for these 3 locations for year 2014
- Compare your series to the time series of the nearest Chimere pixel