Petroleum Industry Data Management: A Spatial Method

by
Norman J. Berls

One major problem facing the petroleum industry is the sheer volume of data it has to contend with. Many diverse types of data are used, including 3D Seismic, 2D Seismic, Well and Fault data. These data yield interpreted information in vast quantities. Major petroleum companies have literally terabytes of digital data. The problem posed by all of this is in finding relevant data at the time it is needed. This problem is fundamentally spatial in nature and so must be the solution.

The problem is compounded by the fact that location information is expressed in a vast array of different systems of measurement including:

- Geodetic Datums
- Prime Meridians
- Map Projection Methods
- Affine Transforms

The fundamental question is:

"Is a particular data parcel relevant to a particular process?"

Global Coordinate Systems

This question is relevant to the people attempting to run a data process and to the process software itself. What is needed is a single system of measurement in which any area on the planet can be expressed.

This is fundamentally a 3D problem. We can begin a solution by adopting the WGS84 geodetic datum as a first step. If the limits of a data parcel are expressed in WGS84 coordinates, it is possible to determine whether the parcel overlaps some area of interest. The problem with this approach is the third dimension. Under WGS84 this dimension is height above an ellipsoid. If our data involves elevation, depth or time, the overlap test is not so easy to perform. Moreover, it would be rather difficult to determine by how much a parcel overlaps... accurately. This last consideration is actually a new requirement. Let us adopt it but, let me defer the rationale for adopting it until later.

The problem can be reduced to a 2D problem by imposing a map projection and ignoring the third dimension. The additional requirement can be met if the map projection is an equal area projection. Thus we are looking for an equal area map projection that will cover the whole planet. Normal Cylindrical Equal Area will do nicely. The projection is made much the same way standard Mercator is done. A cylinder is positioned such that the axis of the cylinder is parallel and coincident with the axis of the earth.

It differs in that it is an equal area projection. Any rectangle in the projection will have XY extents that, when multiplied, yield an area roughly equal to the areal extent of the same rectangle on the earth.

Such a Projected Coordinate Reference System could be set up as follows:

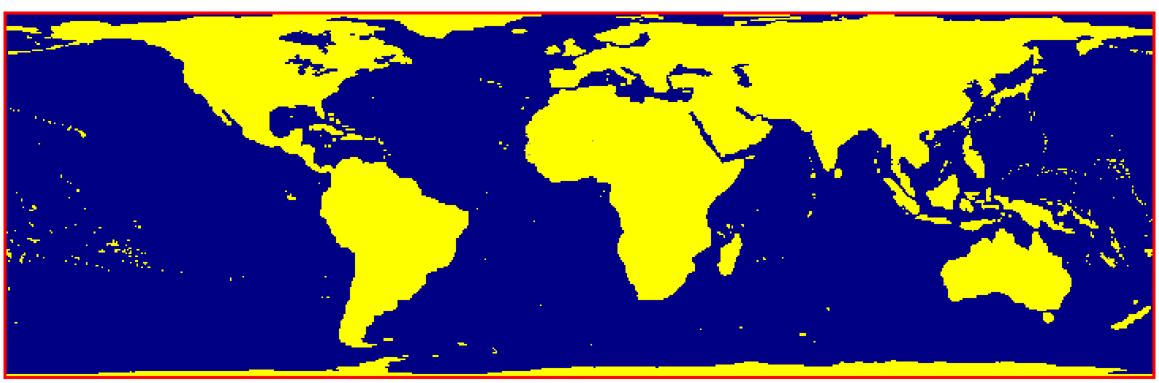
```
          Geodetic Datum: WGS84
          Prime Meridian: Greenwich
        Origin Longitude: 0.0
     True Scale Latitude: 0.0
           False Easting: 0.0 meters
          False Northing: 0.0 meters
```

This system will have extents as follows:

```
     XMIN: -20037508.342 meters
     XMAX:  20037508.342 meters
     YMIN:  -6363885.331 meters
     YMAX:   6363885.331 meters
```

This CRS has some interesting properties. Southern latitudes have negative Y coordinates and western longitudes have negative X coordinates. Also, any area that straddles the 180 meridian must have two boxes to describe its limits, rather than just one. This means that a database design must allow two limits boxes for any data parcel.

For the sake of brevity, I will refer to this CRS as the Global Coordinate System or just GCS for short. The world, expressed in the GCS appears as follows:



The GCS can be used to filter potential input data parcels. If one knows the GCS coordinates of an area where one plans to do some kind of data process, then it is possible to limit the selection list for input data to just those parcels that overlap the process area. In this manner, thousands of choices can be reduced to maybe a few dozen. Such a filtering makes the task of processing data much easier.

Environments

Our goal of determining whether or not a data parcel is relevant to a process can only be achieved if we know the limits of both the process and the parcel in the GCS. The notion of a known system of measurement together with limits is so fundamental that it deserves to be broken out as a separate concept. Let us call such a bundle of information an environment.

The POSC notion of a standardized grid definition is simply an environment combined with a grid lattice. A specification for a map is simply an environment combined with a drawing scale. For the sake of communication between applications and to make it easy to use a common pool of information, environments can and should be standardized as a separate library of resources.
Some environments could include a grid lattice, but every environment should include a drawing scale. This leads to the general rule that every environment should be drawable and some environments ought to also be griddable. To distinguish between environments, every environment should have a name.

While every environment should have two limits boxes that are implicitly expressed in the GCS, each environment is inevitably defined in some coordinate system other than the GCS. While the GCS is ideal for data management purposes, it is largely useless for conducting actual oil industry data processes.

In the end, an environment should include the following:

1. Name
2. Coordinate Reference System
3. Limits in the CRS
4. Limits in the GCS
5. Drawing Scale
6. Grid Lattice (Optional)

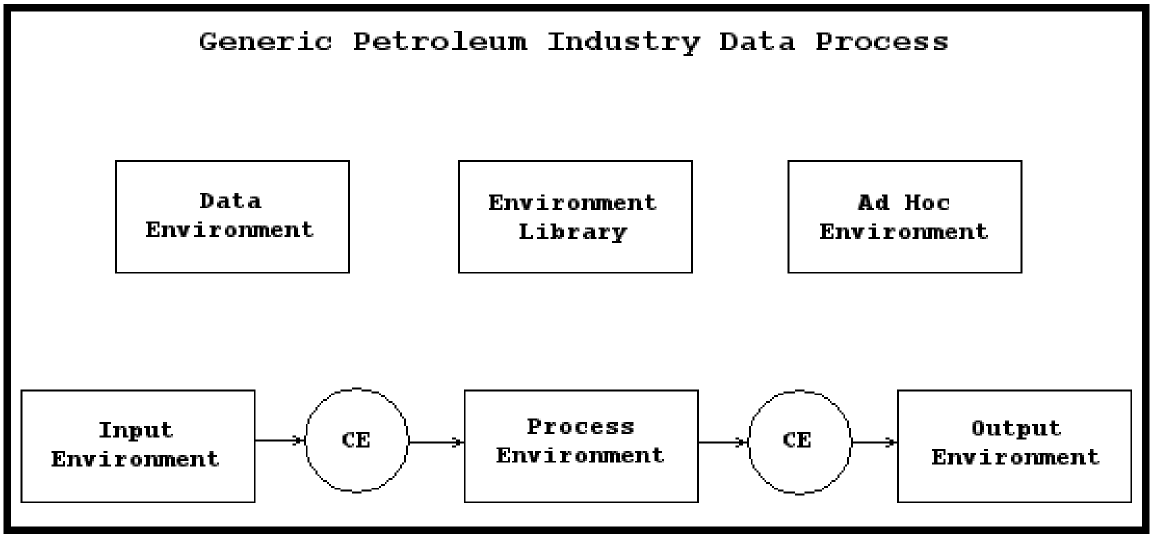
Each data parcel should include the following, slightly diminished, form of an environment:

1. Coordinate Reference System
2. Limits in the CRS
3. Limits in the GCS
4. Drawing Scale (if the parcel is a map)
5. Grid Lattice (if the parcel is a grid)

Such an environment can be plausibly called a data environment, as its meaning and significance derive from the fact that it is an attribute of a data parcel.

In actual usage environments turn out to be fairly fluid things that get moved about, swapped, substituted and varied on an as-needed basis. This flexibility is used to drive a piece of software that is charged with conflating coordinate systems. All location information has to be conflated (homogenized) into a common coordinate system for a data process to be valid. The software that does this work may fairly be called a

Conflation Engine or just a CE for short. Thus a generic petroleum industry data process can be diagramed as follows:



In the simplest possible terms a data process consists of

1. Some input data
2. One or more operations that are applied to the data
3. Output data.

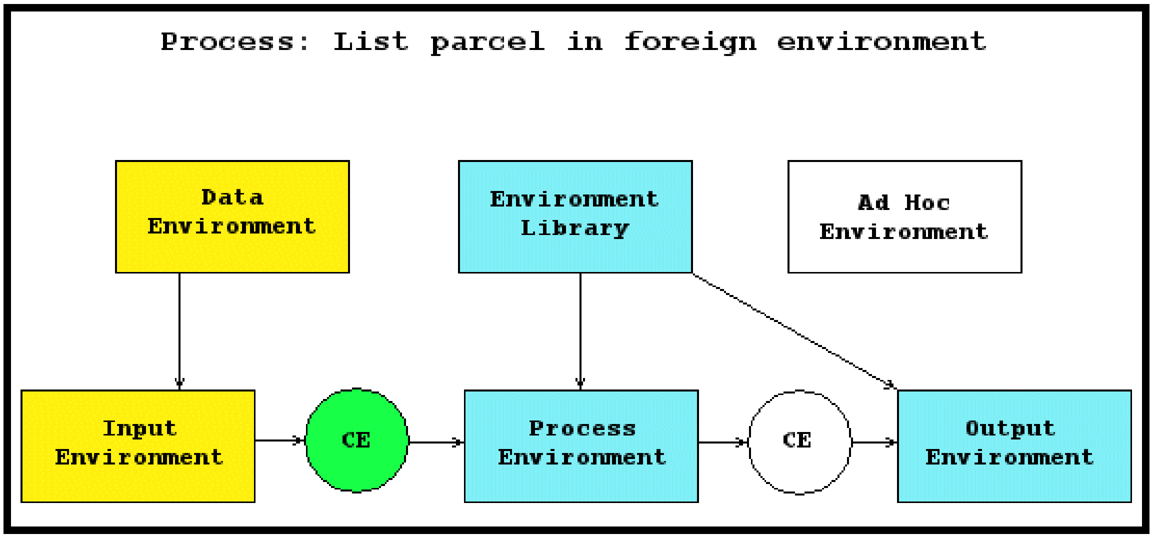
Each of these elements is spatial in nature in that some CRS describes the spatial context of each element. We will be referring to these spatial contexts repeatedly, and so it makes sense to give each of them a formal name:

1. Input Environment
2. Process Environment
3. Output Environment

Because there can and often are differences in these environments, the CE may be called upon to do conflation chores. Of course, this is a substantial oversimplification. To understand the full implications of this model, we need to look at some specific usage cases. In all of the following usage cases, the GCS has been used to filter the input data parcels exposed to the user. The user sees only data parcels that overlap his area-of-interest. This is made possible by the fact that every data parcel has limits expressed in the GCS, and the area-of-interest is expressed in the GCS..

Case 1:

A technician needs to inspect the location information in a certain file. He needs to inspect it to see if any locations are inside the boundaries of a map he has been asked to make. Unfortunately, the data parcel is expressed in one CRS and the map has to be made in a different CRS. An application that performs this task will have to do the following:



1. The environment of the input data parcel will have to be loaded into the Input Environment.

2. The environment of the map will have to be loaded into the Process Environment.

3. Because these two environments are different, the CE will have to perform conflation chores as data is imported.

4. The environment of the map will have to be loaded to the Output Environment..

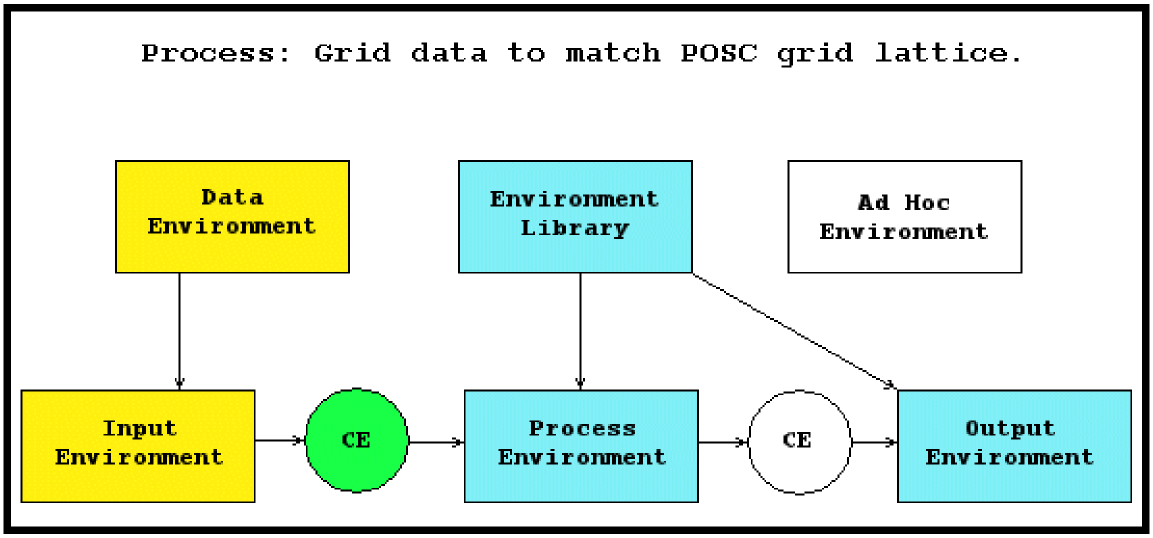
5. Because the Process Environment and the Output Environment are the same, no conflation chores will be needed on output.

Case 2

A technician has been asked to create a surface model in the form of a grid from:

1.One interpreted 3D survey
2. Twenty five interpreted 2D seismic lines
3.Four sets of fault information
4.One polygon to limit the gridding area.

The process environment contains a POSC grid lattice definition that describes the desired output. The CRS underlying the POSC grid lattice is different from the respective CRS's of the input data.



1. The environment of the POSC grid will have to be loaded to the Process Environment.

2. The environment for each input data parcel will have to be loaded to the Input Environment in turn.

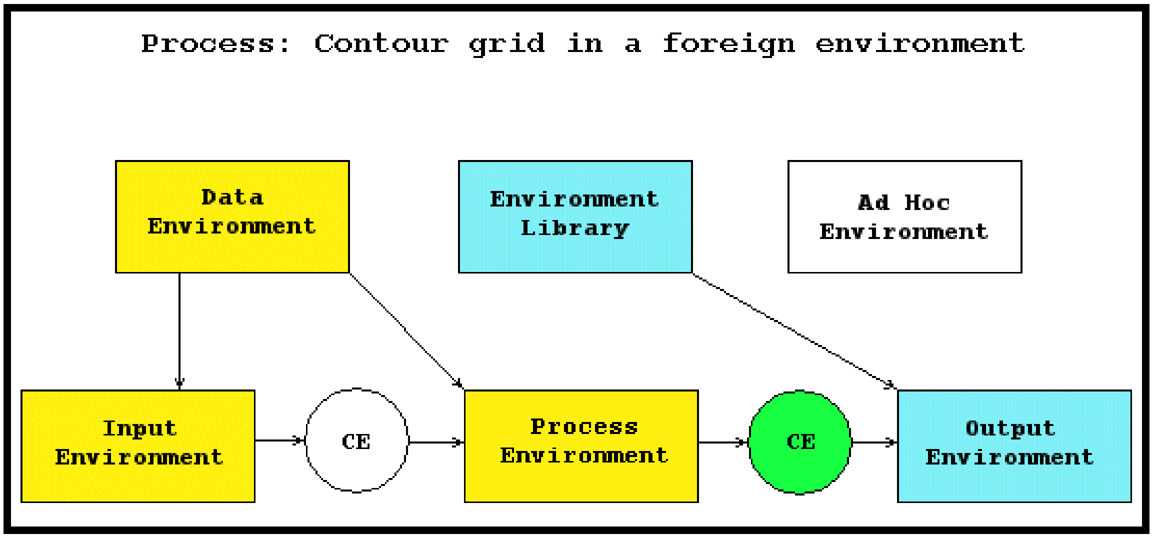
3. Because the input and process environments are different, the CE will have to perform conflation chores as each parcel is imported.

4. The environment of the POSC grid will have to be loaded to the Output Environment.

5. Because the Process Environment and the Output Environment are the same, no conflation chores will be needed on output.

Case 3

A technician has been asked to create a contour map. The grid for generating the contours uses a different CRS from the requested map.



1. The environment of the grid will have to be loaded to the Input Environment.

2. The environment of the grid will also have to be loaded to the Process Environment. Contouring algorithms often depend on having a strictly orthogonal grid lattice to operate in. The only way the grid lattice can be guaranteed to be orthogonal is by dealing with it in the CRS in which it is defined.

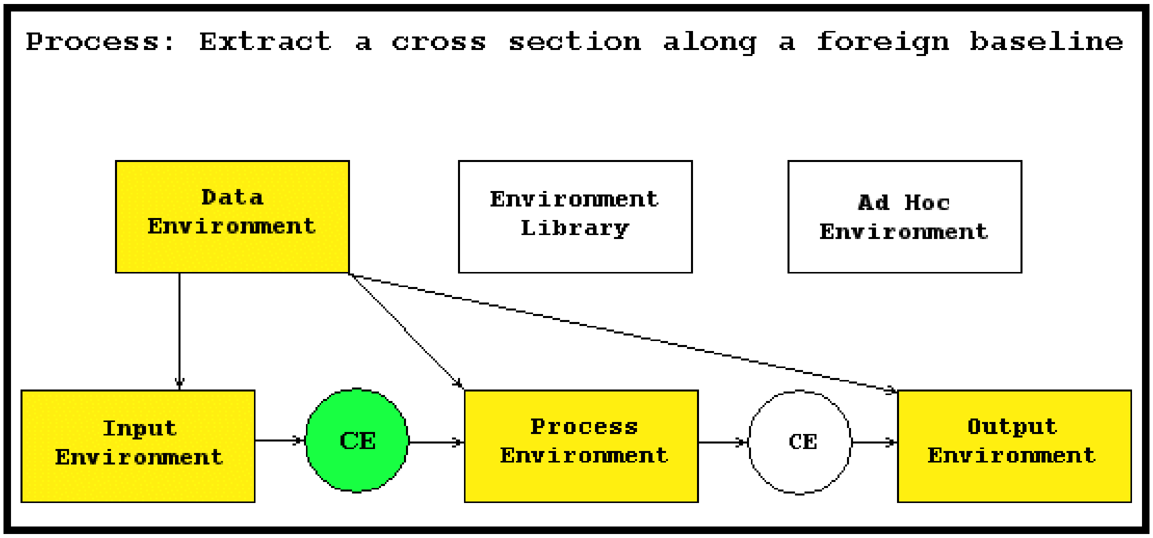
3. Because the input environment and the process environment are identical, the CE is not required to do any conflation chores during input.

4. Load the environment of the map to the Output Environment.

5. Because the Output Environment is different from the Process Environment, the CE will have to perform conflation chores as the contours are written to the output map. Contours end up being long strings of discrete points, so the conflation process is only a matter of conflating a large set of discrete points.

Case 4

A technician has been asked to create a cross section display from 5 grids defined in 5 different CRS's. The baseline that defines the cross section is a polyline (i.e. several different straight line segments chained together). The baseline is in a different CRS than the grids.



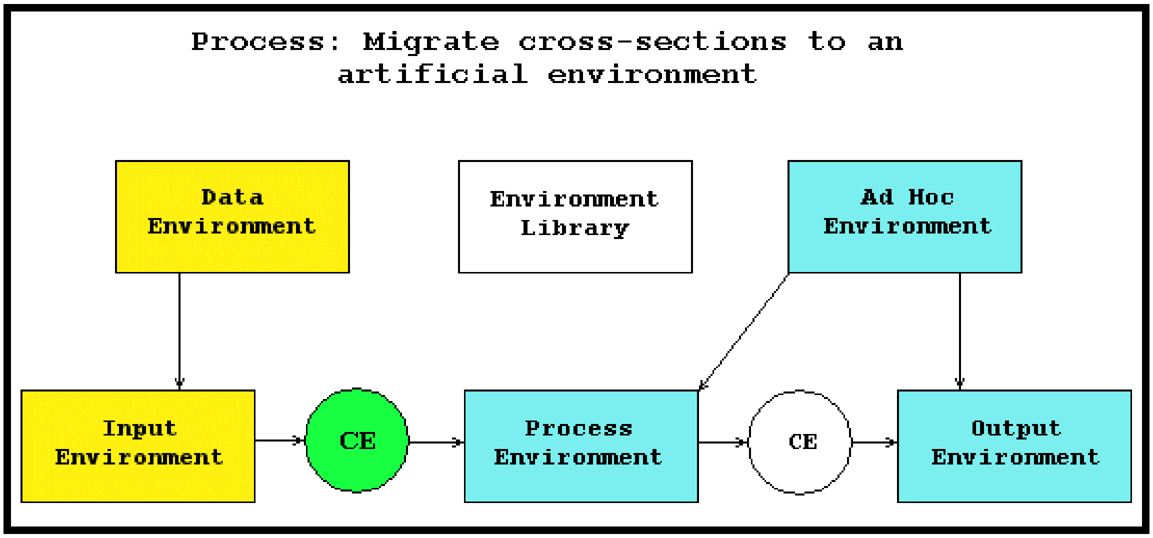
1. For each grid load its environment into the Input Environment, Process Environment and Output Environment. Import the grid.

2. Load the baseline environment to the Input Environment.

3. Because the Input Environment derived from the baseline and the Process Environment derived from the grid are different, the CE will set up to do a transform when the baseline is imported. Import the baseline.

4. This leaves us holding 5 cross-sections in memory. Each cross-section is measured at the same locations along the baseline, although each cross-section is expressed in the environment of the grid from which it was extracted.

A different set of procedures will have to be followed to get the cross-sections into a form for drawing.



5. Create an artificial (Ad Hoc) environment by stretching out the baseline into a straight line that maintains the distances between sampling points in the cross section. This is purely an impromptu action that will be stored only as the environment of the output graphic. It is worth noting that this environment is not a CRS.

6. Load this same environment to the Output Environment.

7. Because the held cross-sections are in the CRS's of the original grids, the locations associated with each cross-section sample will have to be conflated into the Process Environment.

8. The cross-section is then drawn in the Process Environment and output. The CE does not have to do anything on output because the Process Environment and the Output Environment are the same.

CRS Libraries

EPSG offers a list of projected CRS's on the EPSG database. These CRS's make a nice set of initial offerings for anyone wanting to produce maps for the petroleum industry or engage in the processing of petroleum industry data. The only problem is that there are so many of them. It would be very convenient if the list could be filtered so that users would be offered CRS's that were germane to a particular area-of-interest.

Projected coordinate systems can be assigned two limits boxes in the GCS:

1. A Quality Box that specifies a geographic area over which the projected coordinate system will produce small enough distortions that the system will be generally usable for the purposes of the petroleum industry.

2. A Fail Safe Box that specifies a geographic area over which the projection algorithm will run without involving mathematical contradictions. The distortion involved in such an area will often be so extreme that the coordinate system will be unusable. However, the goal here is merely to insure mathematical viability.

Quality boxes can be designed by starting at the natural origin of the map projection. An initial box can be formed for any map projection method by extending the point by three degrees in all directions. This box will usually need to be further expanded depending on the map projection method. For example Transverse Mercator can be expanded to latitudes 84 N and 80 S. Lambert Conformal Conic can be expanded significantly in the east-west direction. Miller Cylindrical can be reasonably expanded to include the whole planet. Special provisions will have to be made for projections like Oblique Mercator where the natural origin is not necessarily inside the area of prime usability. Thus a quality box expressed in the GCS can be produced for any instantiated map projection method.

Doing an overlap test between the area-of-interest and a projection quality box allows instantiated projections to be filtered. Let us call this measurement the %Coverage. Systems that don't overlap at all can be excluded. Systems that cover less than 50% of the area-of-interest could be likewise excluded.

This leaves a problem with large-area projections. Because Miller Cylindrical can be used to map the whole planet, it inevitably will appear on the list of feasible projections, it will have a 100 %Coverage, and it will generally be a poor choice. This can be remedied by doing another calculation for consumption. The question is how much of the quality box is consumed by the area-of-interest? The more of a quality box that is consumed, the more useful an instantiated projected CRS is likely to be. Let us call this measurement the %Consumption.

Filtering should involve the product of the %Coverage and the %Consumption. Filtered instantiated CRS's should be ordered from highest product to lowest product. In this manner, real intelligence can be brought to the process of selecting a usable projected CRS from a list such as the one provided by EPSG.

The thing that makes the %Coverage and %Consumption measurements valid is the fact that the GCS is based on an equal area projection. This is the ultimate justification for using Normal Cylindrical Equal Area projection rather than some other projection for the GCS.

One more bit of filtering can be achieved by asking the user which goal is most important to him:

1. The preservation of the shape of geographic objects
2. The preservation of geographic distances
3. The preservation of the areas of geographic objects

His answer results in the instantiated map projections being limited to Conformal, Equidistant or Equal Area projections.

Thus it is possible to take the EPSG projected CRS offering and build it into a library where each CRS has at least a Quality Box. Such a library would serve as a foundation for an expert cartography service.

Conclusions

Software can go a long way in making petroleum industry data easier to find and use. Spatial technology in the form of:

1. Global Coordinate Systems
2. Environments

can be used to filter input data and other resources so that usable information is brought readily to hand.