CCG Plugins for Petrel

John G. Manchuk, Chad Neufeld, and Clayton V. Deutsch

An important objective of the CCG is to provide industry with an array of geostatistic related algorithms for different problems. Our tools are often transferred to industry as command-line based programs that are flexible and applicable to most working environments; however, they are not user friendly for users who have become comfortable with a particular piece of commercial software. Software vendors are currently working to make it easier for those in research and development fields to incorporate ideas into their software. A good example is the Petrel Ocean API. This short note describes several additional tools or plugins developed at CCG that can be integrated into Petrel.

Introduction

Transferring ideas and research to industry is an important aspect of any research and development group – it is one of the priorities here at the Centre for Computational Geostatistics. CCG transfers many ideas through papers, source code, and programs in a well known format that was established by the Geostatistical Software Library and User's Guide (Deutsch and Journel, 1998). Recently in 2007 another avenue to deliver usable software became available through Schlumberger's Petrel package. They released an application programming interface (API) called Ocean that enables users to integrate ideas directly into Petrel. These tools, or Petrel plugins, are immediately usable. As more tools are developed and released the requirement to export data from Petrel for use in CCG-developed tools is reduced, increasing the likelihood that CCG tools are actually used.

In 2007 two plugins were developed and released for the annual meeting, those being declustering and debiasing plugins (Manchuk, Neufeld and Deutsch, 2007). Since then, development has continued on other plugins. One workflow that has seen increasing use in reservoir characterization is Bayesian Updating (Deutsch and Zanon, 2007). A related suite of plugins has been developed for the 2008.1 release of Petrel and work is being done to migrate to the latest 2009 version. The following plugins are included: NormalScore, Corrmat, Likelihood, Updating, Postmg, and Correlate. All are named similar to the GSLIB equivalent.

This short note describes plugins and how they are developed at CCG; introduces the Bayesian updating plugins; and points to additional resources for users and developers. This is not a user's guide.

Petrel Plugins

Developing plugins for Petrel requires a special licence for both Petrel and the Ocean API; however, no special licence is required to use the plugins developed at CCG, apart from being a member of the group. Ocean gives users the ability to add their own custom algorithms and processes directly into Petrel. These processes have their own user-interfaces and can access many native Petrel objects such as well logs, polygons, and pillar-grids to name a few. New releases of Ocean have made more built-in data accessible – all CCG plugins are currently built on what was accessible in Petrel 2007.

Ocean was developed in a C# environment and plugins must be developed in this language. However, not all components need to meet this requirement – it is only necessary that code for the graphical user interface and for accessing and creating Petrel objects be in C#. At CCG practically every program is written in Fortran. To use these programs they must be compiled as DLL's and linked into the plugins. For CCG plugins, all Fortran programs are compiled in a common engine: ccgeng.dll. More information on these programming concepts can be found in (Neufeld, 2007).

Basically, a Petrel plugin provides an interface to gather data and user input from Petrel and prepares it for use in the CCG algorithm. The gathered data and inputs are passed to a process, which in the case of the CCG plugins is the common engine DLL, that performs the calculations or modeling and passess the resulting output back to Petrel. Input and output can be any Petrel data that is available through Ocean. Ocean also allows custom data types to be created, used and saved within Petrel. Other types of user inputs include parameters and options available on the interface. Plugins can be tied to various menus including the processes tab and the workflow editor where all CCG plugins are found (Figure 1).

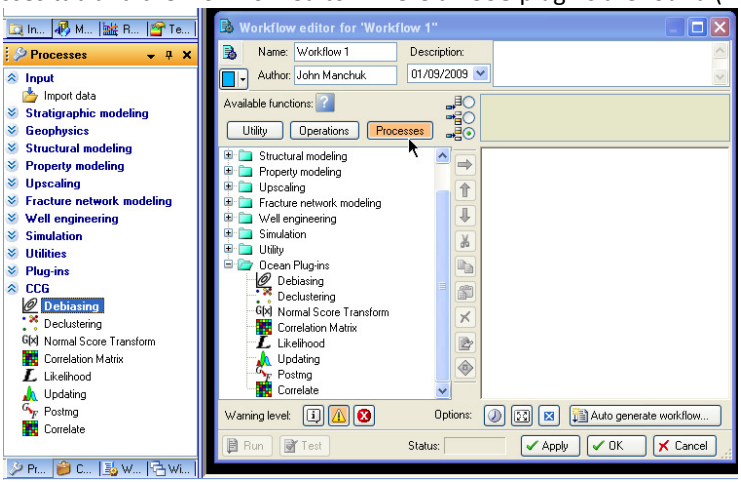

Figure 1: Processes tab and workflow editor in Petrel

PLUGIN DETAILS

The following CCG plugins are available for integration into Petrel 2008.1, with the GSLIB equivalent in parentheses:

- Declustering (declus, Deutsch and Journel, 1998)
- Debiasing (bimodel, Deutsch and Dose, 2007)
- Normal Score Transform (nscore, Deutsch and Journel, 1998)
- Correlation Matrix (corrmat*)
- Likelihood (likelihood*)
- Updating (update*)
- Postmg (postmg, Lyster and Deutsch, 2004)
- Correlate (correlate*)

* described in Neufeld and Deutsch, 2004 and Deutsch, Ren and Leuangthong (2005)

All plugins have a custom built user interface to accept inputs. The type of input data accepted is common to all of them: pillar grid properties, which for some plugins must contain upscaled properties and for others be fully populated. Attempting to pass any other type of data results in a warning as shown in Figure 2. Interfaces were also designed to be similar to existing Petrel tools, such as the typical drop targets shown in Figure 2 for passing data to the interface.

Figure 2: Warning for incorrect data

Plugins also share a common set of components such as drop targets. These are highlighted in Figure 3 using the interfaces from the normal score and postmg plugins. Some of the plugins involve custom domain objects such as transformation tables, declustering objects, and correlation matrices. The main purposes behind these objects are to store information for other CCG processes and display the information graphically. Custom objects are created when a process is executed and they are displayed in the input tab (Figure 4).

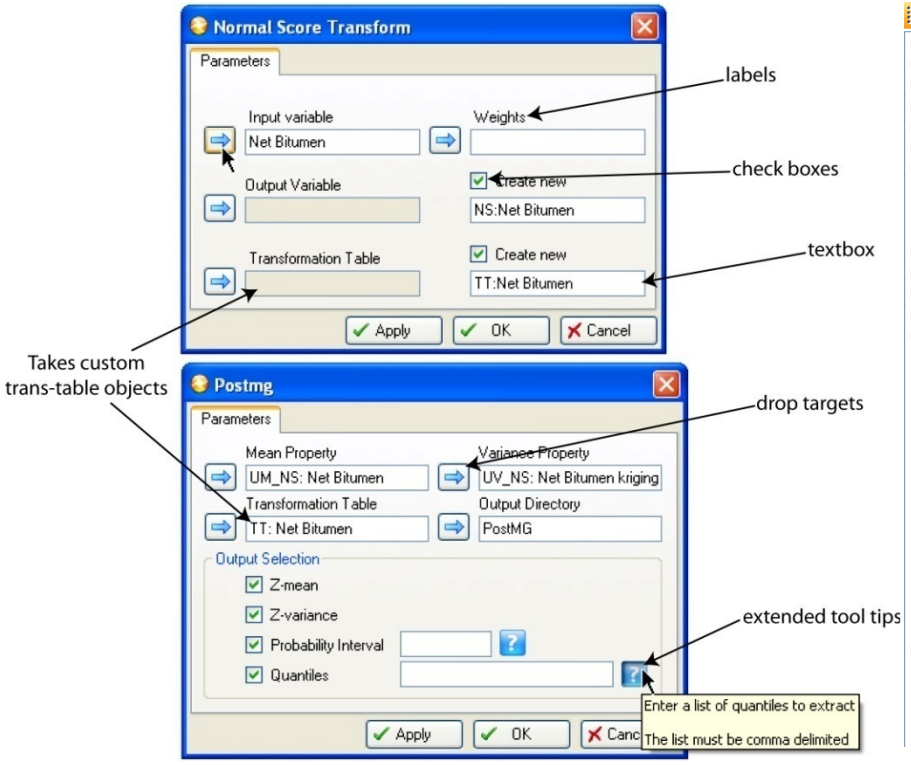

Figure 3: User interface components

Figure 4: Custom domain objects

There is one complication with the declustering and debiasing plugins due to data access restriction with the 2007 Ocean API, which the plugins were original built with. The Ocean API does not allow access to Petrel's function data type. This means that we cannot directly create a function that can be used as part of the petrophysical modeling workflow. However, there is a way that the results can be used. Here are the steps that are required to use the declustering or debiasing results in simulation within Petrel:

1. Run the declustering or debiasing inside of Petrel
2. Right click on one of the declustering/debiasing objects
3. Choose export
4. Choose the file name and where you want to save the file
5. Import the file you just exported as a Function (X/Y)
6. Use the imported function as a General Distribution in the petrophysical modeling workflow.

Distribution

The CCG plugins are distributed to members in a self extracting installation package, which has been included as part of this years meeting. When it is executed (Figure 5), checks are done to ensure the correct version of Petrel is installed (2008.1) and that other requirements are available including Microsoft .NET and windows installation components. If everything is present, the installation will extract all the DLL's into the "Schlumberger\Petrel 2008\Extensions" directory, if that is where Petrel was installed. Installation also adds the appropriate integration information into the petrel.exe.config file located in the Petrel 2008 directory. After installation the ocean modules section of the file should look something like Figure 6.

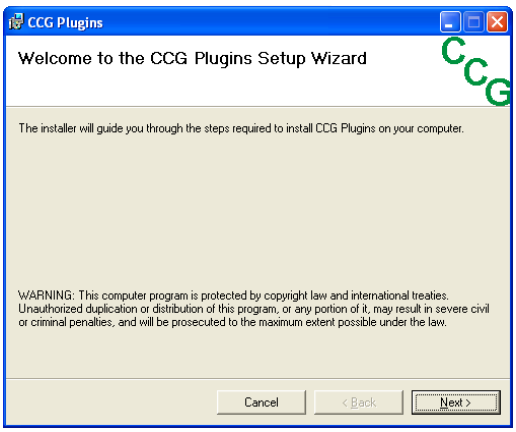

Figure 5: CCG plugin installation wizard

```
<moduleGroups>
  <!-- Add Ocean API based modules here, follow the syntax in the example below - moduleType="TypeImplementingIModule,
  <add trustName="Default">
    <modules>
      <!--<add moduleType="Slb.Ocean.Petrel.Sandbox.Circle.Module, Slb.Ocean.Petrel.Sandbox" />-->
      <!--<add moduleType="Slb.VR.TDxReader.Module, Slb.VR.TDxReader" />-->
      <add moduleType="Slb.Ocean.Petrel.DomainObjectExtensions.UI.Module,Slb.Ocean.Petrel.DomainObjectExtensions.UI,Ver
      <add moduleType="Slb.VolumeInterpretation.VolumeInterpretationModule, Slb.VolumeInterpretation" />
      <add name="RealTime" moduleType="Slb.Ocean.Petrel.RealTime.Core.Module,Slb.Ocean.Petrel.RealTime.2008.1" />
      <add name="LogChannelCatalogModule" moduleType="Slb.Ocean.Catalogs.LogChannelCatalogModule,Slb.Ocean.Catalogs" />
      <add moduleType="Slb.VR.TDxReader.Module, Slb.VR.TDxReader, Version=1.8.1.0, Culture=neutral" />
      <add moduleType="Slb.Ocean.CCG.Geostats.Module, CCGeostats" />
      <add moduleType="Slb.Ocean.CCG.Geostats.DeclusPI.DeclusIntegration, DeclusPI" />
      <add moduleType="Slb.Ocean.CCG.Geostats.DebiasPI.Module, DebiasPI" />
    </modules>
  </add>
```

Figure 6: Integrating plugins into the petrel.exe.config file. CCG plugins are highlig**hted**

Much of the information on the programs underlying the plugins is available in the annual CCG reports. Anyone interested in developing plugins will find the developers guide useful. For 2008, the guide comes in three parts: Ocean core and services, petrel data access, and petrel customization (Schlumberger, 2008). Support documentation is also part of the Ocean installation providing useful reference material.

References

Deutsch C.V. and Dose T. (2005) Programs for debiasing and cloud transformation: bimodal and cltrans. In Centre for Computational Geostatistics, Paper 404, Annual Report 7

Deutsch C.V. and Journel A.G. (1998) GSLIB: geostatistical software library and user's guide. Oxford University Press, 384 pages

Deutsch C.V., Ren W., and Leuangthong O. (2005) Joint uncertainty assessment with a combined Bayesian updating/LU/p-field approach. In Centre for Computational Geostatistics, Paper 203, Annual Report 7

Deutsch C.V. and Zanon S.D. (2007) Direct prediction of reservoir performance with Bayesian updating. Journal of Canadian Petroleum Technology, 46(2): 5 pages

Manchuk J.G., Neufeld C., and Deutsch C.V. (2007) Petrel plugins for declustering and debiasing. In Centre for Computational Geostatistics, Paper 403, Annual Report 9

Neufeld C. (2007) Programming tips for plugins. In Centre for Computational Geostatistics, Paper 402, Annual Report 9

Neufeld C. and Deutsch C.V. (2004) Incorporating secondary data in the prediction of reservoir properties using Bayesian updating. In Centre for Computational Geostatistics, Paper 114, Annual Report 6

Lyster S. and Deutsch C.V. (2004) PostMG: a postprocessing program for multigaussian kriging output. In Centre for Computational Geostatistics, Paper 406, Annual Report 6

Schlumberger Information Systems (2008) Ocean* application development framework version 2008.1, Volumes 1, 2 and 3. Schlumberger