**Wireline Code Summary**

Here are the components to the Wireline application as they sit today and their specific function. All of these code modules are located within this release package.

* **MD1.exe** - (Header file generator, written in C#) – This code is located in the “md1 data header logging” folder within this package. This generates a small header file (txt comma delimited) that is fed into the DITCO application prior to Wireline measurements, and is separate form the rest of the wireline functionality.
* **Wireline Load Workflow(s) -**  (This is a sharepoint workflow specifically written in C# , *there was an older parser written in C++ but when we integrated this into sharepoint as part of the workflow the old one was depreciated*. (The Sharepoint “feed” and the parser are effectively now one) This application is designed to accept a DITCO output file via a Sharepoint workflow (Email Enabled as well as manual), parse it, insert it into the Database, and then run all of the post processing procedures which will later affect what is shown on the map. Note that there are two versions of this application to accommodate the two different versions of the DITCO application that were being used.

Included in this package are the workflows for Ditco v1010 and v1012. They are located within the “wireline\_parsers (Sharepoint Workflows)” folder in this package. Either would need to be compiled and be registered properly within a sharepoint site for use.

* **SQL database** - (This is your standard SQL Server Database SQL Server 2005- We have the create scripts to generate the DB, tables, views, stored procs as well the current data to date is still intact within the existing DB). The scripts are located within the “database scripts” folder in this software package. The DB create Scripts would need to be run first, and the table create second within a working SQL Server 2005 environment.
* **Crystal Reports -** (wireline\_MD1\_summary.rpt) –Despite its complexity there is only one Crystal report, however we do have two versions to accommodate the fact that were measuring drills that had two different versions of the DITCO application installed. This report is simply designed to display wireline measurements in an organized format. (Crystal XI) In this package I have included both the 1010 and 1012 versions of the report and labelled them accordingly.
* **Google API -**  The Google maps API is tied into the main Wireline Sharepoint site using JavaScript, this is a an active server page that renders the Map using the Google API and places specific icons on the map depending on the drill performance, and allows a user to access the crystal report via the map.  (To use the Google Maps API one needs to acquire a valid Key through Google) The asp page containing this functionality is located in the “map\_front\_end\_asp (Sharepoint asp)” folder within this package, and would need to exist as its own page within the sharepoint site.
* **Optimal Speed Force Interface –** This is an active server page linked into the main Wireline site that allows a user to set expected speed force targets for the drill.  From the pervious point the Google Map asp will interact with the **table** in which this data is saved in order to determine which icon is displayed per drill based on performance. (ASP) Located within the “optimal speed force interface” folder
* **Web server setup –** There is a setup utility that we need to use to set up the Web server. Setup code is located within the “CrystalRpt11 webserver setup” folder.
* **Misc Front End Code –** Once the Web server is set up there is additional front end code that is required in order to link to the crystal report.  (This is written in PHP and ASP). This code is located in the “CrystalReport\_Viewer” folder
* **Depreciated code (Old version - pre sharepoint)** – although this code is considered an older version, that was used prior to implementing the sharepoint solution, I have still included it as part of this release package. Contained are a windows based parser written in C++, the old front end code that was used for loading KML files, and the Loader which is essentially a BAT file. These would have only conformed to an older version of Ditco release 1010, possibly even earlier than 1010.