BOIVerify
version 2.0

SmartTools, Procedures, and Scripts for maintaining a
gridded verification database and displaying various statistics.

Tim Barker - WFO Boise, ID

This collection of tools, procedures, and utilities can be used to maintain a database of gridded forecasts and analyses for producing verification statistics over a grid. The database is stored 'offline' outside of the GFE databases, so that the normal purging of data by GFE does not purge the verification database. However, the tools access the database and display statistics and grids within GFE, so that the verification data can be used within the GFE program to produce better forecasts.

The gridded data is packed extensively, and only saved for parameters where suitable observation grids are available, which means that verification is possible. Nevertheless, the database of verification data is very large, and cannot be stored indefinitely, even on the large data partitions available on AWIPS. With reasonably sized grid, saved for only a few models, for a few parameters, verification grids can be stored for about 1-2 months. During this period, *grids* of verification statistics can be generated and used to adjust forecasts to remove recent biases, etc.

A separate database of a few error statistics for site-configurable, pre-defined editAreas can also be created. This database is smaller, and thus data can be stored for much longer periods of time. Sites can use this data to see how forecast statistics change over long periods of time – for the pre-defined points or areas.

Table of Contents:

History and Highlights of the current version 2
Basic Overview 4
- **The common Date / Forecaster / Model selectors** 7
- **Grid Displays Tab** 18
- **Grid Stats Tab** 25
- **Distributions Tab** 35
  - **Histograms** 35
  - **Scatterplots** 41
- **Point / Area Stats** 45

Bias Corrected Grids 61

History and Highlights of the current version

This first version of the software contains a bit more refined user interface and better stability than the alpha version (0.1), but still is missing various pieces that will be eventually added. Verification of probabilistic parameters (like PoP) or highly variable parameters (like QPF) is currently not handled. There are still problems with dealing with forecaster numbers and assigning the 'responsible forecaster' to a particular grid. There also needs to be more support for making tables of data, or extracting statistical data for use in other programs. All this is currently under development.

Changes from version 1.0 to version 2.0:

- Added the ability to handle probabilistic parms like Probability of Precipitation (PoP). Only handles the "floating" PoP definition.
- Added the ability to verify a parm with a differently named parm. Thus, for example, we can have a forecast parm named QPF, verified by a parm named "QPE".
- Added the ability to save more than one GFE database as "Observed" grids, and calculate and display statistics using any of the configured "Observed" databases.
- Added several new statistics and displays, many of which are particularly useful for probabilistic parms like PoP, and highly non-normal parms like QPF.
- Added the ability to "smooth" or "average" grids before verifying them, or consider gridpoints within an "neighborhood area" – providing some measure of accuracy at different "scales".
- To save time, added the ability to limit the forecast hours over which statistics are calculated and displayed.
- Added the ability to handle "accumulative" or "rate" parms like QPF, and provided a way to specify the 'accumulation time period' for verification calculations.
- For Vector parameters, added the capability to show statistics calculated on the magnitude of the vector error, often in addition to the ability to show errors in magnitude or direction separately.
- Added the ability to calculate scores and statistics for runs that several models have "in common". Thus, comparing scores between models can be more justified – since they were made for the same periods.
- Changed the Bias Correction algorithm slightly so that in cases where "extrapolation" is needed, the algorithm trends back toward a true "bias correction", rather than using the linear regression results. This helps mitigate unusual results when the forecast values are far outside the range of forecast values that have occurred in the recent past.
- Changed all the tools to be procedures, which removes the need for requiring the special WG1 variable, and removes almost all cases where unexpected failures would lock grids and keep future runs of the tools from being able to proceed.
- Changed the logic of the archiving (or "Save") procedure so that it does not require nearly as many crontab entries. Furthermore, the timing of the scripts is not as critical.
- Separated the configuration information from the Utility, so that future versions can be installed without having to copy and paste configuration information from the old utility to the new utility.
- Documented all the routines in the BOIVerifyUtility, and provided a programmers tutorial for using the routines in other tools and procedures.
- Provided several new sample programs showing some of the features available in the BOIVerifyUtility. One is a basic tool to extract information into a comma-delimited file that could be imported into other programs like Excel. Another is a more fully-featured program that extracts many statistics about daily Max and Min temperatures, and formats a small

"summary" HTML file that can be e-mailed to forecasters, or sent to a local intranet page. Many sites may wish to implement the summary program pretty much 'as is'.

- Added a small GUI program that allows the forecaster 'names' or 'numbers' associated with forecast grids in the archive to be modified (since the GFE history information is often inaccurate). This program also provides rudimentary ways to remove 'bad' forecast grids from the BOIVerify archive.
- Various changes to the logic that often helped to speed up the calculation of statistics.
- Fixed several bugs found in version 1.0, List: ??????

Changes from version 0.1 to version 1.0:

- The internal data file format was changed slightly so that several forecaster numbers could be associated with a single grid (a "team" approach). Also, vector elements were changed to keep the two components within the same file, rather than in separate files.
- Overall stability of the program has been improved quite a bit.
- The user interface has been cleaned up a bit:
    - A dialog was added to show the progress of calculations and allow the forecaster to 'cancel out' of calculations that are taking a long time.
    - The ability to 'shrink' the dialog to a small 'mini-window' was added – so that forecasters do not need to keep moving the dialog 'out of the way' to see the results loaded in the GFE. For certain types of displays where the GFE grid manager is heavily used – the dailog automatically shrinks to this small window when it makes the calculations.
    - For gridded displays, a separate window is created with a list of the number of cases that were actually found for the grids displayed. This can help in interpretation – since sometimes grids from two models or two times will be displayed side-by-side in the grid manager, but represent averages or calculations over a very different set of cases.
- The distribution tab has been expanded to display scatterplots as well as error histograms. The labels and legends of the histograms have been improved. More configuration options to help control the display of histograms have also been added.
- The 'fourth tab' for display of time series of stats for pre-selected edit areas is now supported (as well as the programs running in the background to create these statistics). You can also display average stats over a time period, as a function of forecast time, and overlay them for different edit areas, or different models.
- Displays and statistics of gridpoints within certain 'error thresholds' are now available. Five site-configurable thresholds for each parameter, allow a site to see the percentage of grid points with errors less than, for example, 2 degrees, 5 degrees, 7 degrees, etc.
- Statistics for vector parameters are now working, however more work needs to be done here, since there are certainly different KINDS of statistics needed to get a better idea of problems with vector forecasts. Nevertheless, the simple scalar statistics applied to the wind speed and wind direction can still be instructive.
- Automated scripts/tools to calculate 'bias corrected' model grids are provided. These tools provide a simple way for forecasters to obtain the most commonly desired verification output – current model grids with 'recent biases' removed.

Basic Overview:

The main portion of the software is a GFE utility (called BOIVerifyUtility) that is used by the various tools and procedures to interact with the 'offline' verification database. Cron entries routinely run shell scripts that run the BOIVerifySave procedure to save model grids, official forecast grids, and observation grids, into the verification database.

At the current time, the default parameters available in the Obs database are MaxT, MinT, MaxRH, MinRH, T, Td, RH, and Wind (and optionally TdMrn and TdAft). If you are using another database for the 'truth' grids, and it has other scalar or vector parameters, you can configure the BOIVerifyUtility to save them as well, and the standard statistics and displays will be available. PoP forecast grids are being saved at the current time, even though the tools do not yet support statistics for probabilistic parameters like this. For the grids which are potentially forecast hourly (T, Td, RH and Wind), the forecast grids can be saved at 3-hr, or 6-hr, intervals, rather than every hour – mainly to save space in the verification database. Observation grids of these parameters (T, Td, RH and Wind) are saved hourly – even though forecast grid are only saved at 3-hr intervals.

Model forecast grids for the parameters are stored for all forecast times from every model run. For example, the GFS40 grids are available from 00Z, 06Z, 12Z and 18Z runs, while NGM80 grids are only availble at 00Z and 12Z. Forecasts are saved in the verification database with information that contains the 'basetime' or 'cycle' that they came from, as well as the forecast time that they cover.

Official forecasts, on the other hand, are 'continuously updated' by the WFO forecasters. Rather than save grids every time the official database is 'published', the Official grids are saved only twice a day at 00Z and 12Z. The official grids saved at 00Z are 'marked' as coming from a 'basetime' or 'cycle' of 12Z the previous day, and likewise the official grids saved at 12Z are 'marked' as coming from a 'basetime' or 'cycle' of 00Z. While somewhat arbitrary, this is meant to be consistent with the model forecast data that would be available to forecasters at the same time. For example, forecasts put out in the afternoon (say, around 20-22Z in Mountain and Pacific timezones) are typically made by forecasters who only have the latest 12Z model runs to use as guidance. Thus, when the official grids are saved at 00Z, they are saved in the database as coming from a 'basetime' that is the same as the 12Z model runs. This logic may need to be adjusted as the scripts/tools are used in other time zones.

Some of the GFE 'history' information of the official grids are available when the grids are stored in the verification database. Thus, the GFE username that last touched the official grid is stored with forecast grid. In theory, this can be used to generate statistics for a particular GFE user. However, in practice, this rarely works very well. Often grids are 'touched' by multiple forecasters during the day – so that saving the username that last touched it before 00Z is relatively meaningless. Furthermore, many GFE smartTools mark a grid as having been 'calculated' by a tool or algorithm, or copied from a particular model. In these cases, the history information indicates that NO forecaster has touched the official grid – rather than indicating which user ran the calculation tool, or copied in the model data. In these cases, the forecast grids are saved with a dummy forecaster username. Thus, any statistics calculated for an individual forecaster username should be approached with a great deal of care. Tools to help maintain the forecaster 'association' with each grid are under development.

Basic use of the tool

The BOIVerify procedure provides a very large Graphical User Interface (GUI) to extracts grids and statistics from the verification database, and display them within the GFE system. The procedure is accessed from the **Verify** menu of the GFE menubar

The options for BOIVerify are so numerous that the GUI has been broken into various 'tabs' across the top of the dialog, where different verification information can be accessed: Grid Displays, Grid Stats, Distributions, Point/Area Stats, and Stat vs. Scale:

The tab that is currently "active" is shown "depressed". Each tab provides access to different kinds of displays and statistics:

Grid Displays – This tab allows you to display grids that are in the BOIVerify database. For example, using this tab you can display the GFS40 forecast from "last week" to see what it looked like. For forecasts where observed grids are also available, the error grid (simply the forecast grid minus the observed grid), can also be displayed. The GFE is used for these displays, with potentially large numbers of grids being loaded into the GFE grid manager, so that you can view many different grids quickly.

Grid Stats – This tab allows you to calculate various statistics at every gridpoint – and display the results as a grid. For example, you can calculate the average error (also called the 'bias') over the past two weeks of official forecasts – displayed as a grid. Again, the GFE is used for these displays, so that a large number of statistics grids can be viewed quickly.

Distributions – This tab allows you to display errors or values, combined over time and space, as histograms or scatterplots. For example, you can calculate a histogram of GFS40 errors over the past 30 days for a particular forecast zone, and compare it to the same histogram for the official forecasts. You can also graph a scatterplot of forecast values to observed values, to see biases that might depend on the forecast values. These graphics are drawn in a window that is separate from the GFE, with graphs for various times or models available as overlays.

Point/Area Stats – This tab allows you to display statistics for groups of points, (called editAreas in GFE). You can display a time series of a statistic over time, and compare them for different models, different editAreas, different forecast times, or different parms. You can also display a graph of statistic versus forecast times, and compare them for different models, different editAreas, or different parms. For example, after saving data for several months, you will be able to graph a time series of day-3 model forecast errors over time and overlay the official forecasts to see when the model does better than the official forecasts, and vice versa. In addition, you could compare the average forecast errors over the past few months for various models, perhaps seeing that one model does better for short-term forecasts, but another does better for long-range forecasts. Again, these graphics are drawn in a window that is separate from the GFE, with the ability to overlay graphs from various times, models, or edit areas.

Stat vs. Scale – This tab allows you to display a statistic for a group of points, but calculate it using various 'scales', and graph the statistic value versus scale.

Hiding the large BOIVerify Dialog

At times, the large size of the BOIVerify dialog combined with the large size of GFE display can make it difficult to see the results you would like. Rather than having to drag the BOIVerify dialog back and forth 'off the screen', you can shrink the BOIVerify dialog to a very small window. At the bottom of the BOIVerify dialog are the main buttons:

The Run button starts the calculations that you have currently selected. The Quit button, of course, closes the tool. The Hide button simply shrinks the dialog into a very small window:

By default, this small window is placed in the lower-right corner of the GFE display, but you can place it anywhere you like. Simply clicking on the BOIVerify Options button brings back the large dialog where you can change options and run the tool again.

When you use the "Grid Displays" and "Grid Stats" tab, the results are loaded in the GFE grid manager and displayed in the main GFE window. Thus, for those tabs, after the calculations are performed, the dialog will AUTOMATICALLY hide itself, so that you can see the results.

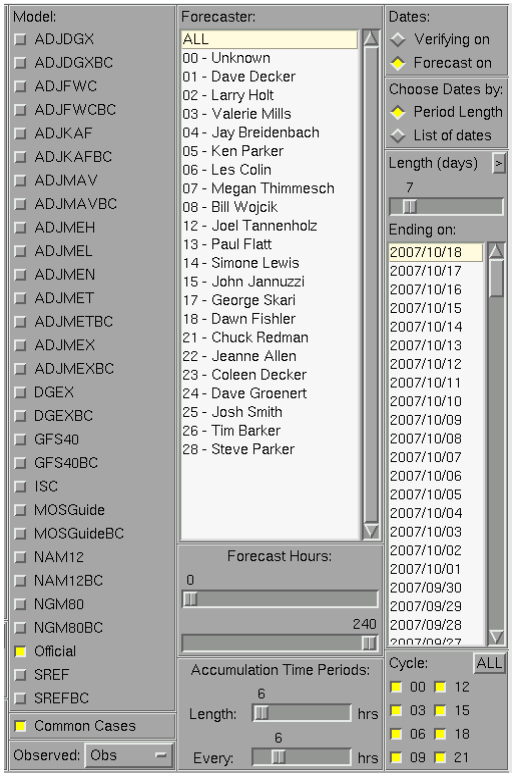
Feedback while working on calculations

At times the BOIVerify calculations can take quite a while to complete. After you click on the "Run" button, a small dialog with "feedback" information will appear:

The display will update as various parts of the calculation are completed. If you click on the "Stop" button, the calculations will stop and the tool will go back to the main BOIVerify options dialog.

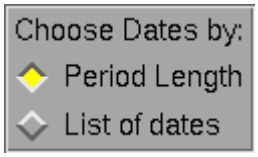
The common Date / Forecaster / Model selectors

Each tab of the BOIVerify tool dialog contains the identical right three columns where you specify the particular dates, cycles, forecasters, and models to use in the calculations:



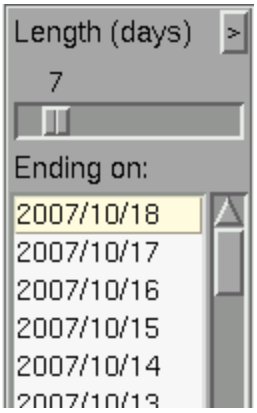
These options are the same, and shared, among each BOIVerify 'tab'. We will discuss these common options first, starting with the rightmost column:

Dates: *Forecast on* or *Verifying on* – Sometimes you want to view graphs or statistics for forecasts that were MADE on a particular set of days. For example, you may be interested in knowing the skill of model forecasts made when you are under a large ridge. In this case, you want to set the "Dates:" section to "Forecast on". Other times you want to view graphs or statistics for forecast that VERIFIED on a particular set of days. For example, you may be interested in knowing the skill of model forecasts for days that turned out to have strong winds in a certain area. For these cases, you want to set the "Dates:" section to "Verifying on".



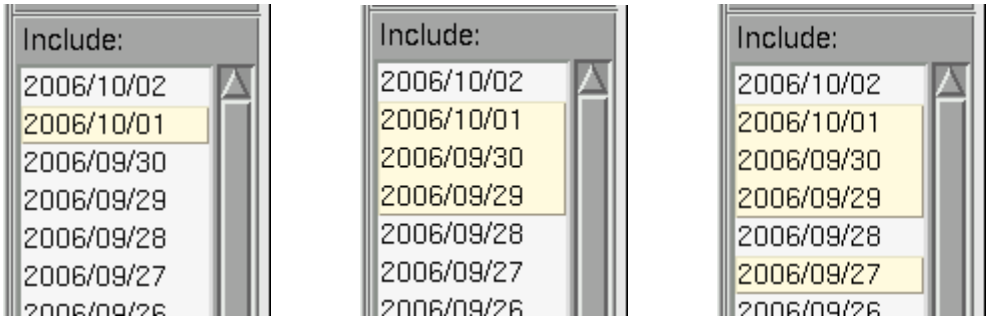
Choose Dates by: Period Length or List of dates – Sometimes you want to calculate statistics over a consecutive period of days. For example, you might to see statistics for the last week, or for the two weeks ending on the first of March. For these cases, you want to set the "Choose Dates by:" section to "Period Length". At other times, you want to calculate statistics over a particular set of dates. For example, you might know that wind events happened on March 3rd, March 7th and 8th, March 23rd, and April 5th. For these cases, you want to set the "Choose Dates by:" section to "List of dates".

Period Length – If you specify to choose dates by period length, then you will see a slider to pick the length of the period (in days), and a scrolled list where you can choose the last day of the period. The small slider can be difficult to work with, so you can change the 'range' of values shown in the slider. The slider can either range between 1 and 50, or between 1 and 500. To switch between these two sliders, click on the button next to the Length label. When you are using the 1 to 50 day slider, this button will show a ">" to indicate that when you click on it, it will expand the slider to range from 1 to 500. On the other hand, when you are using the 1 to 500 day slider, this button will show a "<" to indicate that when you click on it, it will shrink the slider range to 1 to 50. Even with this adjustable slider, it can be difficult to control the mouse accurately enough to set it at exactly 20 days. If you click in the "trough" of the slider (to the left or right of the slider button) it will move the slider in increments of 1 day. Thus, it is often easiest to slide the slider to a value that is "in the ballpark", and then click a few times in the trough to get the exact value that you desire.



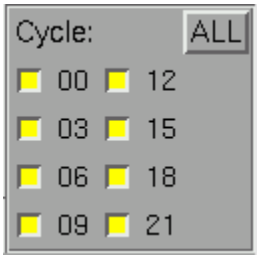
The dates shown in the "Ending On" list are in Year/Month/Day format – and shown in reverse order, with the most recent days at the top and older days at the bottom. In the example shown above, we have chosen a period 7 days long, that ends on October 18, 2007.

List of Dates – If you specify to choose dates by a list of dates, then you will see only a scrolled list where you can pick dates. As with picking the ending dates of a period, the dates are in Year/Month/Day format, and in reverse order with most recent dates at the top of the list and older days at the bottom. Unlike the list for choosing the ending day of a period, this list allows you to choose multiple entries using the standard GUI conventions. A regular "left click" clears the current selection and selects the entry under the mouse pointer. A "shift click" extends the selection from the last selected item to the entry under the mouse pointer, and a "control click" adds or deletes an element from the selection.



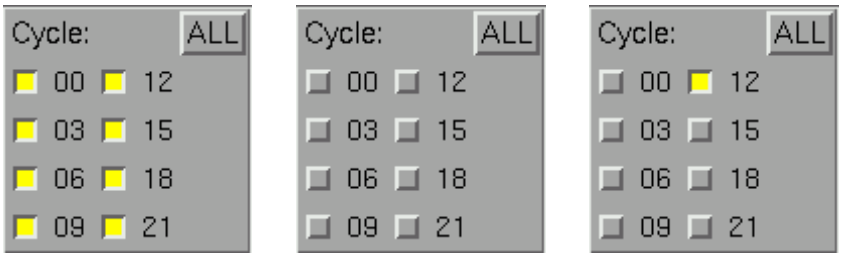
For example, in the sequence shown above, the forecaster clicked on 10/01, which cleared the selection and a then set 10/01 as the selected element. Then the forecaster held down the shift key while clicking on 09/29, which extended the selection between 10/01 and 09/29. Finally the forecaster held down the control key while clicking on 09/27, which added only 09/27 to the current selection. Using the shift and control keys, you can quickly pick many dates.

Cycle - Sometimes you want to calculate statistics only for a particular run of the model during the day, while other times you want to calculate statistics for any run during the day. The "Cycle:" section allows you to choose particular runs that you want to include in statistics or displays. Not all cycles are available for all models. If you specify to only show data from

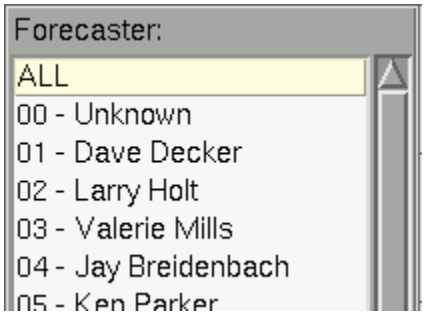


00 UTC runs of the DGEX model, you will not get anything, since the DGEX model is only produced at 06 and 18 UTC. The default is for all model runs to be considered or displayed.

Since there can be a large number of cycle buttons, the ALL button is provided so that you can toggle all the cycle buttons at the same time. This can be a quicker way to choose one, or only a few cycles.

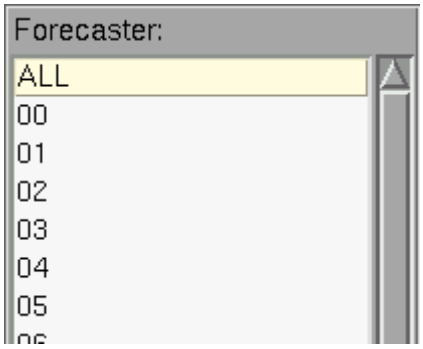


For example, in the sequence shown above, we started out with all the cycles selected, then we clicked on the ALL button to toggle off all the cycles, and then clicked on just the 12Z cycle to choose only that cycle. Without using the ALL button, we would have had to click OFF seven other cycle times – which would have been very tedious.

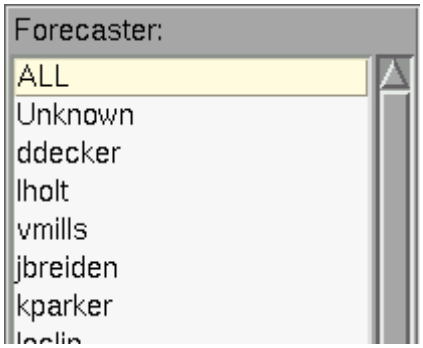


Forecaster – Sometimes you want to calculate statistics for forecasts made by a certain individual. Of course, this choice has no impact on statistics for model forecasts – since they are not made by anybody. The "Forecaster:" selection only affects statistics and display of Official forecasts. The list is similar to the "list of dates" above, because you can choose several forecasters, and the statistics and displays will be for forecasts made by ANY of the forecasters you specify. The default is the shortcut "ALL" entry, which means that ANY and ALL forecasts are included in the statistics and displays. You could select a particular forecaster IN ADDITION to the ALL entry, but it wouldn't matter – since one you have ALL chosen, any additional forecasters are already included.

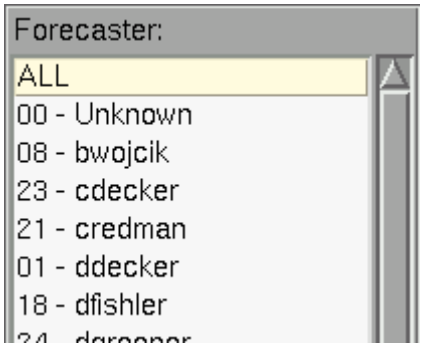
The labels for forecasters can be shown in a variety of ways: either as full names, usernames, forecaster numbers, or both. The forecaster list can also be sorted by forecaster number, or username, or full name. For example, here is the forecaster list shown with forecaster number, and sorted by the forecaster number:



Here is the same list, but shown with the forecaster username, and sorted by the forecaster number:



And here is the same list, but shown with both the forecaster number and forecaster username, and sorted by the forecaster username:

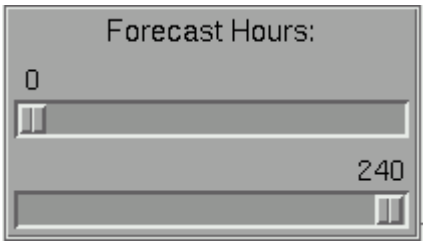


See the configuration documentation for details of changing this display.

One problem with statistics calculated for a particular individual is related to how GFE tracks who has 'touched' a grid. When the Official forecasters are saved, the only forecaster data available is the ***last*** GFE user that modified that grid. Some sites use 'generic' GFE users of 'long term', or 'short term', or something similar. In these cases, those will be the GFE users associated with those grids – not a particular forecaster. Some sites run a 'finalize' procedure before publishing grids, which fixes minor grid consistency and integrity issues. But if that procedure modifies a grid, the user that runs the procedure gets 'associated' with the grids because they were the last one to touch them – even though another forecaster may have done most of the editing work. Even worse, many tools and procedures run in such a way that they modify several grids at once – which GFE interprets as 'a tool' modified the grids – and NOBODY is associated with the grid (it is considered 'calculated'). The BOIVerifyInfo procedure provides a GUI that allows you to change the forecasters associated with each Official grid in the BOIVerify database.

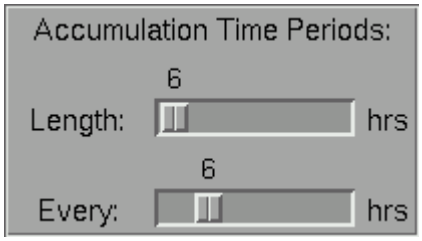
However, consider that forecasters never work in isolation. We always inherit grids from the previous forecast shift, and rely on the expertise of others, office neighbors, etc. The most important issue should be the quality of OUR grids, as a whole, rather than individual "finger pointing" at people that make 'good' or 'bad' forecasts. On the other hand, finding personal biases can help each of us become better forecasters. The bottom line is that calculating personal statistics can be both enlightening and misleading, so a certain amount of care should be exercised when interpreting the results.

Forecast Hours – By default, the system will calculate and display statistics for every available forecast hour. However, since some of the calculations can take quite a bit of time, you can use the sliders in the "Forecast Hours:" section to set the minimum and maximum forecast hours to calculate and display.



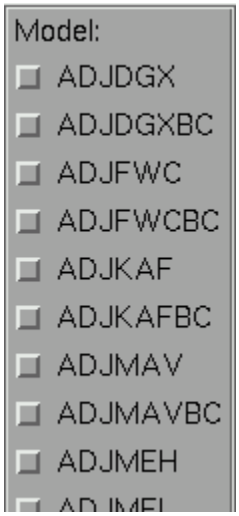
The forecast hour is always relative to the ***start*** of the forecast grid. For example, a 12-hour long MaxT grid, which runs from 16 to 28 hours after the start of the forecast, is referred to as a 16-hr forecast. If you set the Forecast Hours sliders both to 16, you would get statistics for this period. However, if you set both the Forecast Hours sliders to 17, you would not. The starting time of the forecast grid must fall within the range of hours set by the sliders.

Accumulation Time Periods – The "Accumulation Time Periods:" section allows us to specify more details about time periods, when dealing with accumulative parameters like QPF, or probabilistic periods like PoP. For example, we might want to calculate verification statistics for 6-hour long QPF periods, or for 24-hour long QPF periods. To modify the length of the periods, use the "Length" slider.



In addition, we may want to calculate statistics over long, but overlapping, time periods! For example, we may want to calculate QPF statistics for the 12-hour periods from 0 to 12 hours, 3-15 hours, 6-18 hours, etc. We refer to this as the ***frequency*** of the accumulation time periods, and you can specify it using the "Every" slider. In this example, we would set the length to 12 hours, and the frequency (via the "Every" slider) to 3 hours. If you set the frequency equal to the length, then you will not get overlapping periods. The default time period length and frequency, as well as the available resolution of the time period length, are all configurable, as discussed in the configuration documentation.

Model – Sometimes you want to view statistics or grids from a single model, while at other times you want to view or load the information from several models at the same time so that you can compare them. The "Model:" section allows you to choose one, or many, models for statistics and displays. The example shown at the right has many more models than you probably have available. It is easy to add a new model into the BOIVerify system. For details, see the configuration documentation.



The models are listed in alphabetical order. The Official forecasts are listed here as just another "model" – and it is often the default model that is selected when the BOIVerify tool starts. Remember that the "Forecaster:" section of the GUI is also used when selecting data to be used in calculations and displays. The ISC data (if your office configures BOIVerify to save it) is also shown as a "model". Just like within GFE itself, the ISC data contains each of the neighboring office's forecast grids in their respective grid area.

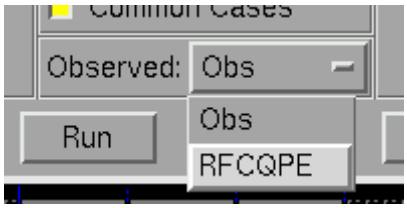
Common Cases – When you choose several models, you most often want to calculate statistics ***ONLY*** for those cases that the models have in common. Version 1.0 and earlier of BOIVerify did not have this ability. For example, if you only had one run of the GFS40 model available during a month, and thirty runs of the NAM12 model, old versions of BOIVerify would calculate the statistics independently for each model, and display them side-by-side. This could be quite misleading since the statistics for the NAM12 came from thirty cases, but the statistics for the GFS40 came from just one case!

As of version 2.0, the default is to calculate statistics for the cases that the models have 'in common'. So, for our example above where the GFS40 had only one run during the month, and the NAM12 had thirty, the statistics displayed would now be for only one case for both models (as long as the one run from the GFS40 was from the same time as one of the runs from the NAM12!). This is particularly important when showing statistics for particular forecasters. Without considering common cases, old versions of BOIVerify would display the statistics for the few cases that a particular forecaster made, right next to the statistics from ALL available model runs. Such comparisons are very unfair. Now, BOIVerify compares the statistics for the forecaster to statistics for the model, *from the same cases*.

It is worth pointing out that another configuration element comes into play when calculating 'common cases'. For example, if the NAM12 model comes from 0, 6, 12, and 18Z runs, while the SREF model comes from 3, 9, 15, and 21Z runs, then there would be no way for these models to have any runs 'in common'. Technically that is correct, but in practice, we usually want to allow 'related' runs to be compared, even if they do not start from exactly the same time. This is controlled by the BASE_OFFSET configuration element, which can be set for each model. In the case described above, we might set the BASE_OFFSET for the SREF model to -3. The BASE_OFFSET (in hours) is added to the model runtime when making decisions about 'common cases'. With it set to -3 hours for the SREF, a 3Z run of the SREF would be 'in common' with the 00Z run from the NAM12 (as long as the BASE_OFFSET for the NAM12 is left at 0 hours – the default). We could just as easily set the BASE_OFFSET for the SREF to +3, so that the 21Z runs of the SREF would be 'in common' with the 00Z runs. Typically sites set the BASE_OFFSET configuration so that runs available to forecasters at about the same time, appear to have the same basetime, or runtime. For example, in the Mountain timezone, when dayshift forecasters are making grids, they typically have the 12Z NAM12 model grids at their disposal, as well as 15Z SREF model grids. Thus, we set the BASE_OFFSET for SREF to -3 hours.

Without carefully considering the implications of all the settings, you can sometimes come up with confusing instances where no 'common cases' exist. For example, if you set the Cycle to only allow 00Z cycles, but ask for both the GFS40 and SREF models, it will NOT find any common cases because the SREF has no 00Z cycles. You must include both the 00Z and 03Z cycles to compare the 00Z GFS40 to the 03Z SREF. The 03Z SREF will appear to be a '00Z run' in terms of common cases (if the BASE_OFFSET is set to -3), but will not be considered at all unless the appropriate Cycle option is chosen. Also, the Forecast Hours, are relative to the 'common case' time, not the real model runtime.

Observed Model – Starting with BOIVerify 2.0, you have the ability to save 'observed' grids from any number of sources. For example, you may want to save grids from nationally supported analysis systems like MAPS, or LAPS, in addition to grids from locally run analysis systems like MatchObsAll. You may want to verify against QPF grids from your RFC, in addition to QPF grids derived only from radar data, etc. The configuration documentation specifies how to save the grids for these various database. The Observed section allows you to specify which 'observed' database you want to use for the calculations. Simply 'pull down' the menu, and choose the desired observed database. For example, here we have saved 'observed' grids from both a database named "Obs", and one named "RFCQPE", and we are about to choose the RFCQPE database:

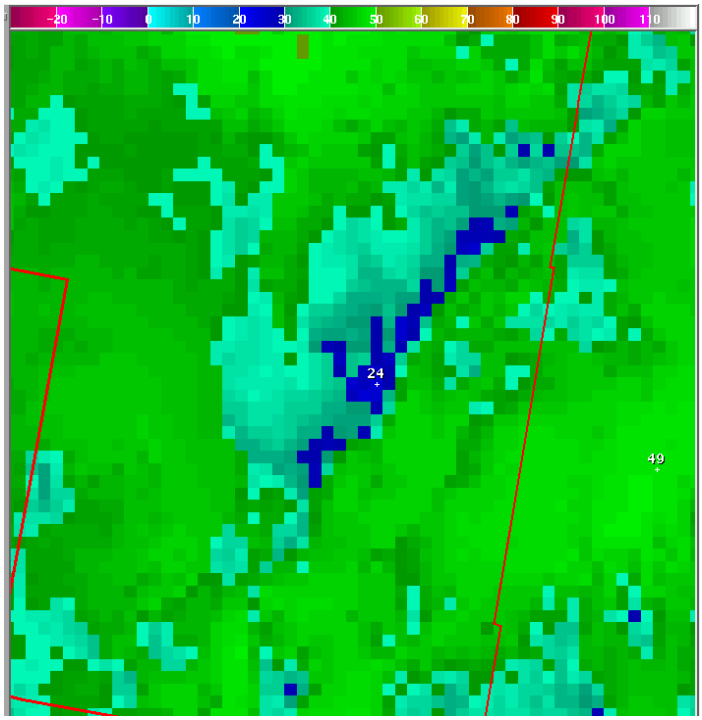


Scale (or smoothing) selector – With most of the BOIVerify tabs, you also have the ability to specify a 'scale' or 'smoothing' value. This typically shows up as a button in the "Display:" section of the left-most column:

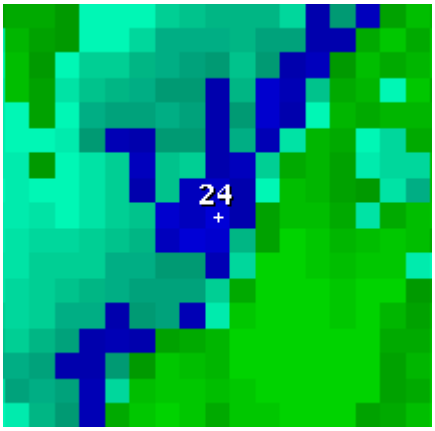
This 'scale selector' is also 'shared' among all the BOIVerify tabs, meaning that when you change it on one tab, it also changes on the others.

This option allows you to specify an area around each gridpoint to be used when calculating statistics. In most cases, this simply means that the forecast and observed grids are 'smoothed' by averaging over all gridpoints within the 'square' of the specified size, centered over each gridpoint. By default, this is set to the resolution of your GFE grid – which means that no smoothing is performed. However, if you specify larger values, by selecting from the menu, smoothing takes place.

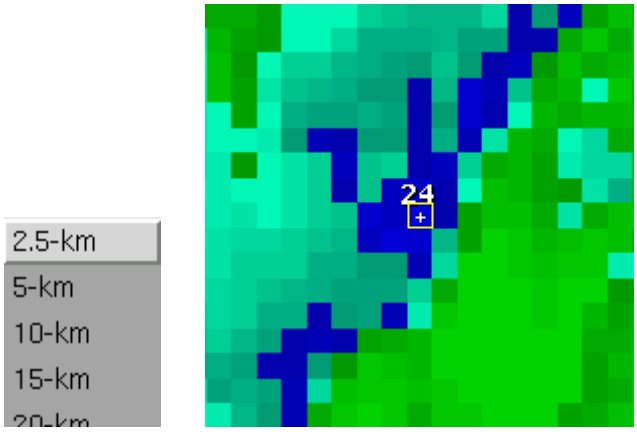
Let's examine this 'smoothing' or 'scale' option more closely. Say we have an observed grid MaxT grid that looks like this:



We are going to concentrate on the gridpoint with the 24 degree sample near the middle of the display. Let's look at the area around that gridpoint in more detail:

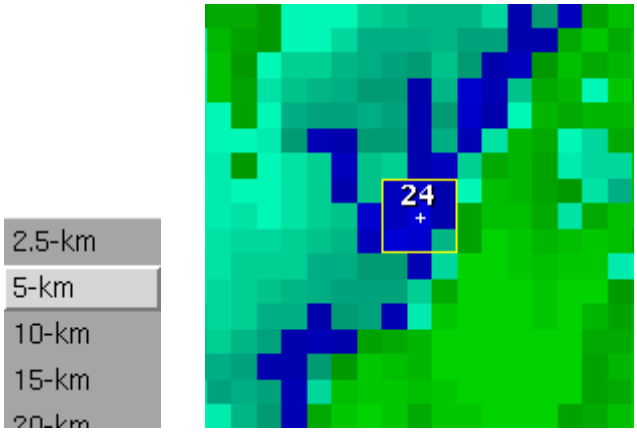


When we perform normal statistics calculations at the gridpoint, we are really considering the whole 'gridbox' that encompasses the 24 degree value, which is shown here with the yellow box:



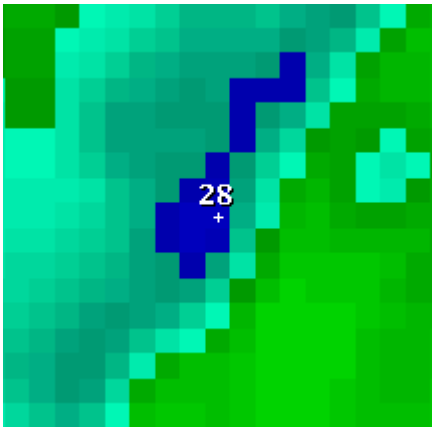
In our case, our grid resolution is approximately 2.5km, so the box is approximately 2.5km on a side, when the ‘scale selector’ is set at 2.5km (the default).

The next choice in the ‘scale selector’ is 5km, which is a box centered over the gridpoint, but considering one more gridbox in each direction:



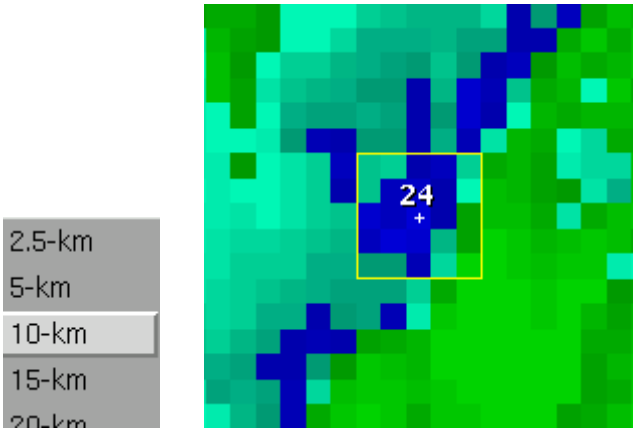
Now, it is true that the yellow box depicted here is 3 gridboxes wide, by 3 gridboxes high, and is technically 7.5km on a side. However, it is the usual convention to refer to this as a 5km box, since it covers 5km from the center of the leftmost gridpoint to the center of the rightmost gridpoint. If you feel we should change the naming convention, please let me know – I am truly open to suggestions on this issue!

For this gridpoint, we can smooth simply by averaging all nine gridpoint values inside the box, and plotting that value at the original gridpoint. If we do that for *each* gridpoint, we get a smoothed grid that looks like this:

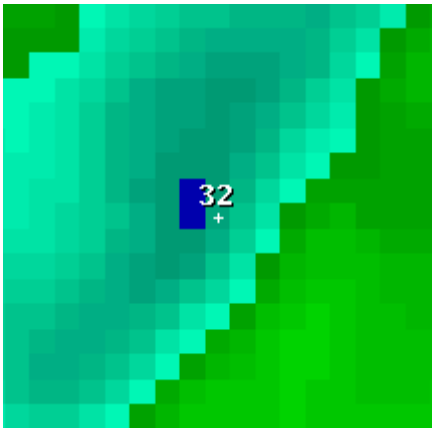


Note that the value for our gridbox changed from 24 to 28. The warmer temperatures at the surrounding gridpoints created a smoothed value that was warmer.

The next choice in the 'scale selector' is 10km, which simply considers all the gridboxes in the square within 2 gridboxes of this point:



Note that there are now 25 gridboxes within this smoothing area! The box is technically 12.5km on a side, but, again, we will refer to it as a 10-km box. Smoothing over these boxes at each gridpoint, yields a grid that looks like this:



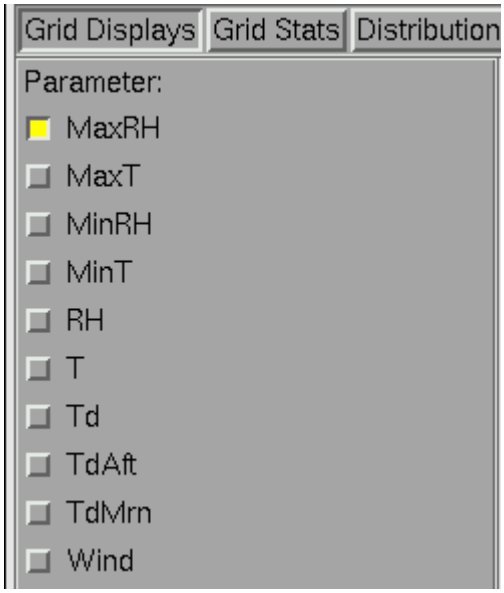
Note again, that instead of our original value of 24, we now have a value of 32 at this gridbox. The smoothing operation tends to 'flatten out' areas of minima and maxima. If you smooth over extremely large areas, you will eventually get the average value over the entire grid.

For some statistics, rather than smoothing over the square ‘scale box’, a condition is ‘searched for’ within the ‘scale box’. For example, we can calculate an “Areal Threat Score” for QPF greater than 1.0 inches. Rather than a QPF forecast having to be forecast and observed at precisely the same gridpoint, we would search inside the ‘scale box’ to see if the condition was forecast or observed anywhere within the ‘scale box’. Such techniques are often called ‘fuzzy’ or ‘neighborhood’ verification scores, and are often appropriate for very small scale and difficult to forecast phenomena, such as precipitation. More details on the “neighborhood” verification scores are found in the statistics section, under the Grid Stats tab.

Grid Displays Tab

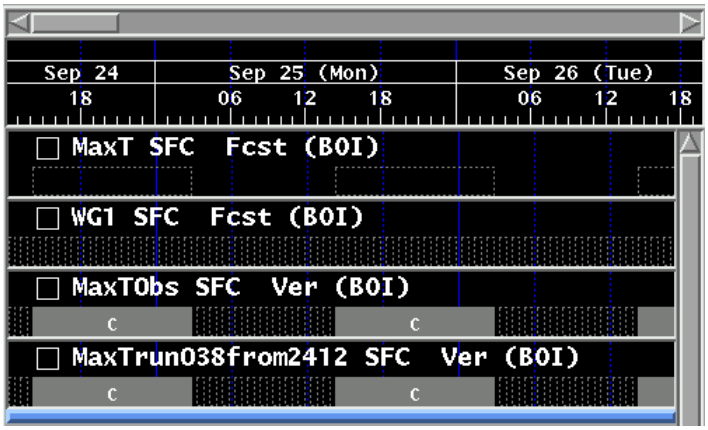
The Grid Displays tab is used to display the individual grids kept in the verification database.

Parameter – Sometimes you would like to see grids from several different parameters at the same time. These are chosen in the "Parameter:" section. Although multiple parameters can be displayed at the same time, it can become very time consuming to display many different parameters from many different models.

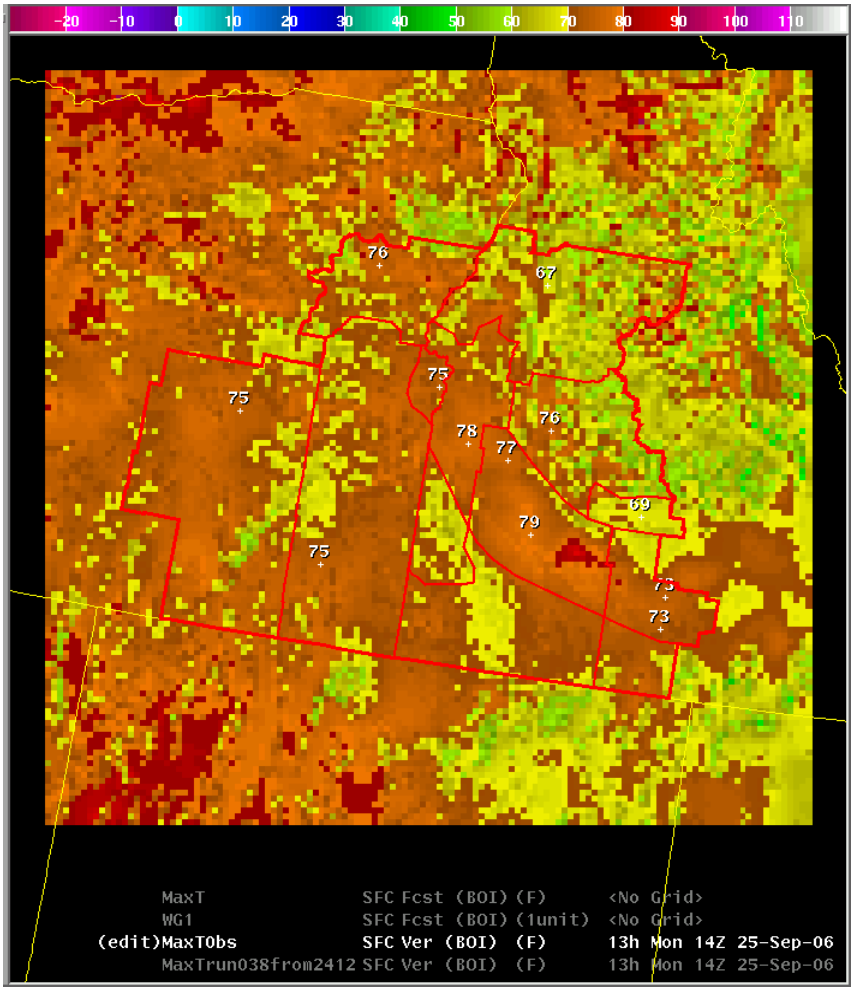


Again, the list of parameters available at your site may vary. Configuration of the software to add a new parameter is relatively easy. In the example shown here, the TdAft and TdMrn parameters are available, which are typically only created by Western Region offices.

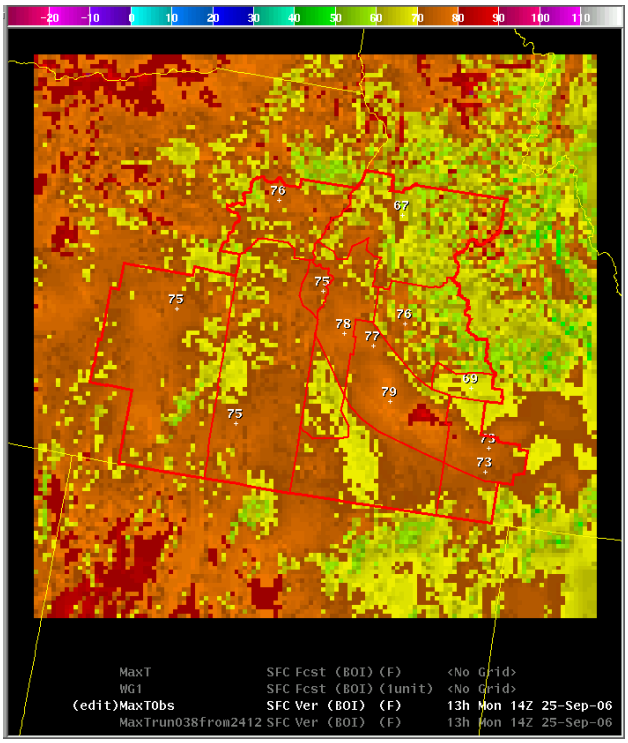
Display – Sometimes you would like to see what the archived forecast grids themselves look like, but at other times you want to see what the forecast errors look like. The "Display:" section allows you to choose which of these you want displayed. When you choose "Forecasts", the forecast and observed grids are loaded into the grid manager at the times the forecast were valid for, and shown using the default range and color curve used for that parameter in the Fcst database. For example, here we chose to show the MaxT temperature forecasts from the 12Z Official forecasts on September 24th:

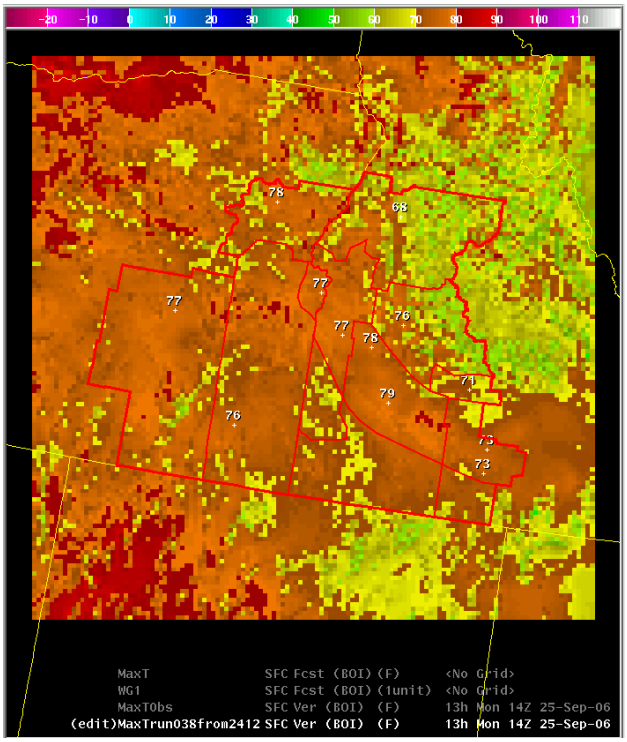


The tool removes any other grids from the grid manager, other than the Fcst MaxT grids and the temporary WG1 element (if you have it defined). Then it loads the Observed MaxT grids, along with the forecast temperature grids below the observed grids. If we were to click on the gridblock for the MaxTObs grid for Sep 25th, we would see a display like:

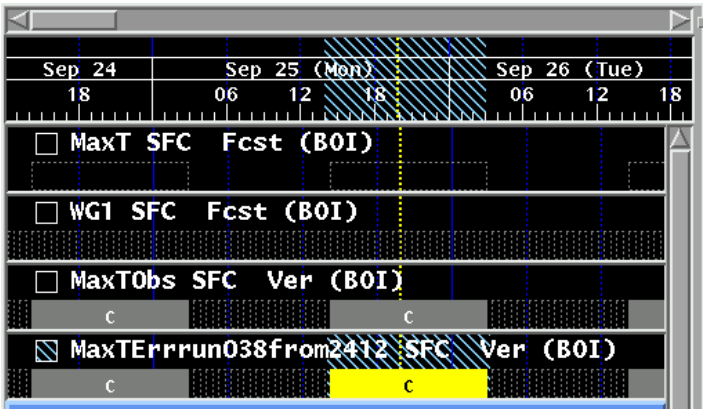


The color curve used here is the default one that we use at BOI for temperatures, and our default temperature range is from -30 to 120 degrees. The forecast grid for this time period is loaded in the grid manager directly below the observed grid. Just clicking on that gridblock in the grid manager allows the forecaster to switch quickly back and forth between the two displays. Here the observed is shown on the right and the forecast on the left:

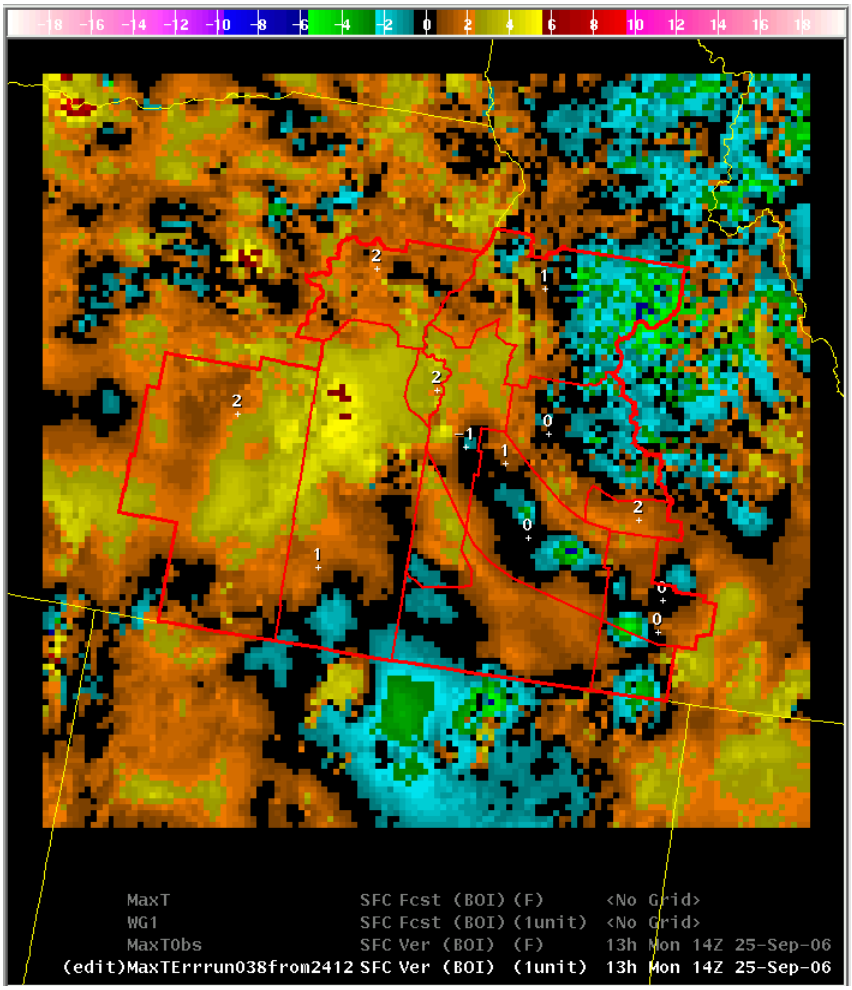




If we had selected errors in the “Display:” section, the grid manager would be loaded in a very similar way (only the label of the forecast grids is different):



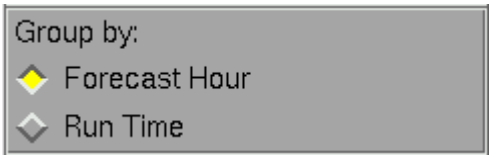
The observed grid will be displayed using the same range and color curve shown above. However, the forecast grids are now shown as forecast errors, which is simply the forecast grid minus the observed grid:



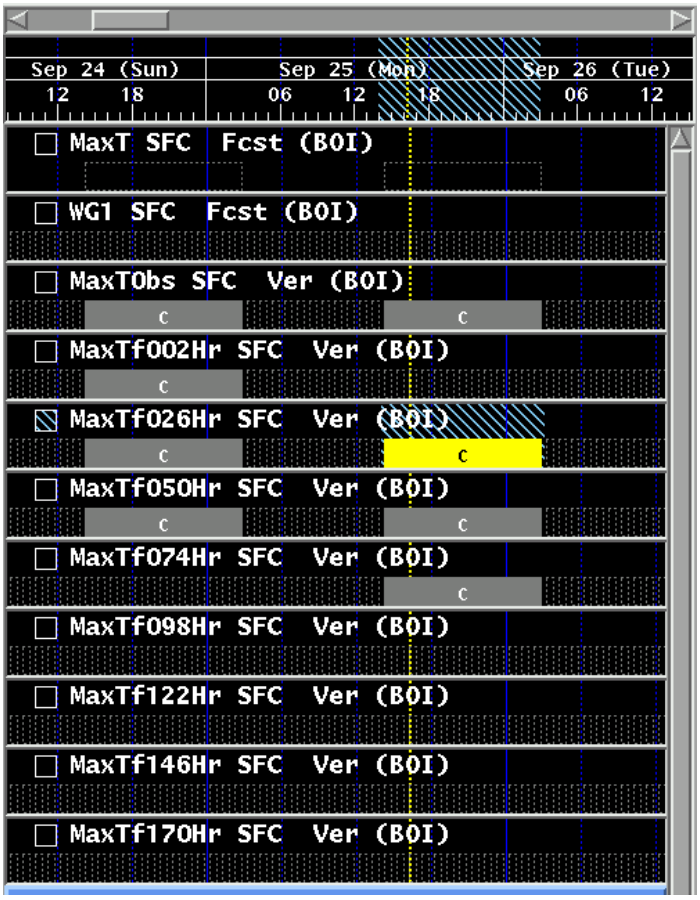
The default color curve used is intended to highlight the forecast errors. Warm colors of orange, yellow, and red, indicate where forecasts were higher than observed. Similarly cool colors of cyan, green, and blue indicate where forecasts were lower than observed. The range of errors displayed here is from -20 degrees to +20 degrees. Errors larger than these values will simply be 'clipped' for display. The error limits, and color curve, are all configurable at each site. For example, some sites may find that errors of +/- 20 degrees are very large and most of their errors fall between +/- 10 degrees. In that case, it would probably be better to display errors between +/- 10 degrees so that the colors are more spread out across the error distribution, making it easier to interpret the displays.

The default error color curve and error ranges can be different for each parameter. For example, by default, temperature errors are shown with "warm colors" used for positive forecast errors. However, for dewpoint, the default is to use "warm colors" for negative forecast errors, since "warm colors" seem to associate better with "too dry", while "cool colors" are used for positive errors, since those associate better with "too wet". Similarly, while the default range of errors displayed for temperatures is from -20 to +20, the default range of errors displayed for RH is from -40 to +40, due to the more difficult nature of these forecasts.

Group by – Sometimes you want to compare forecasts at similar forecast hours, while at other times you want to compare forecasts made from the same run. The "Group by:" section allows you to change the way that the grids are loaded into the grid manager for easier interpretation.

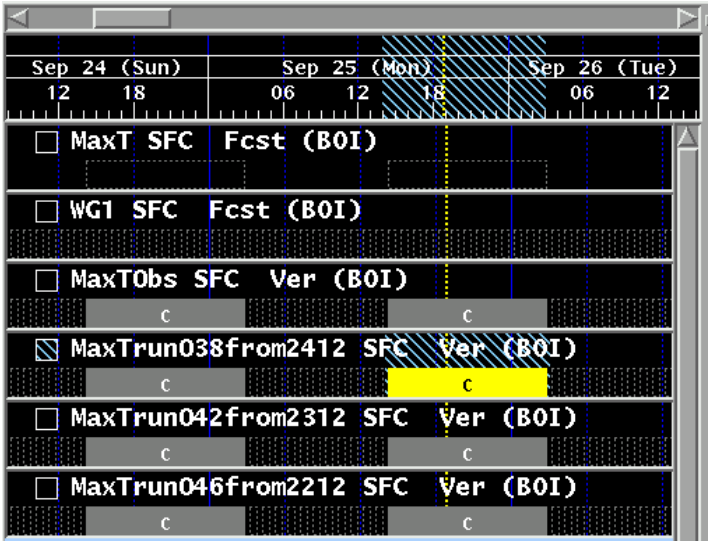


When you choose the “Forecast Hour” option, grids are loaded into the grid manager with similar forecast times running horizontally and labeled with their forecast time. For example, here we chose to display the 12 UTC Official MaxT forecasts from the 3 days ending on Sept 24th:



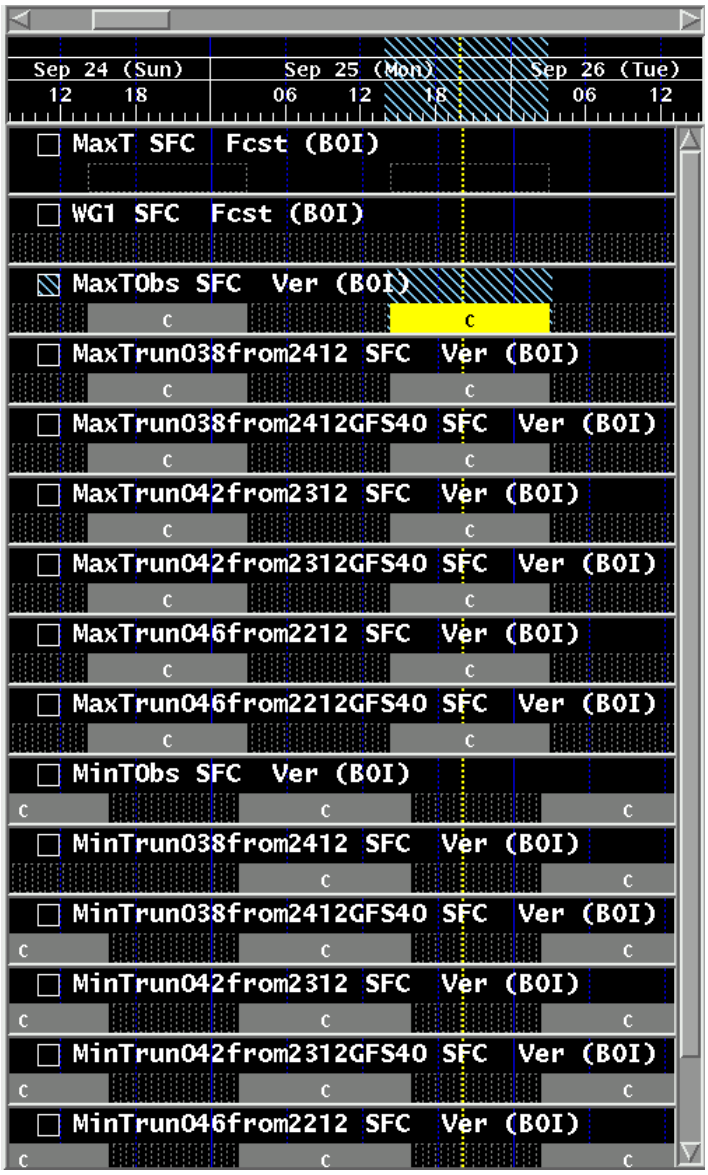
Note that the highlighted grid is valid on the afternoon of Sept 25th, and is labeled as MaxTf026Hr, indicating that it is a 26hr forecast. The forecast hour is taken as the BEGINNING of the grid, so the beginning of this grid is 26 hours after the “run time” of 12 UTC on Sept 24th. The grid located “directly below” this in the grid manager is also valid on the afternoon of Sept. 25th, but is labeled as Maxt050Hr. This is the 50hr forecast from the “run time” of 12 UTC on Sept 23rd (recall that we asked for 12 UTC forecasts for the three days ending on Sept. 24th). Similarly, the grid directly below this is the 74hr forecast from 12 UTC on Sept. 22nd. Note that all these forecasts are valid on the afternoon of Sept. 25th and are lined up under the Observed MaxT grid for that day.

Note that grids from the same run are located “diagonally” when using this “Forecast Hour” grouping method. On the other hand, when you choose the “Run Time” grouping method, grids from the same run are located “horizontally” across the grid manager. For example, here we have again chosen to display the 12 UTC MaxT forecasts from the 3 days ending on Sept 24th using “Run Time” grouping:



The highlighted grid is once again the 26hr forecast for the afternoon of Sept. 25th. But it is now labeled as MaxTrun038from2412. The number "038" is simply a sequence number chosen so that the three runs will sort correctly in the grid manager – with the most recent forecast at the top. Once again, all three forecasts of the MaxT for the afternoon of Sept. 25th are lined up under the Observed grid for that time, but grids from the same "run time" are now loaded horizontally across the grid manager.

Multiple parms or models – You can choose to display grids from multiple models and multiple parms, as well as from many different forecast runs from many days. Quickly you will be loading THOUSANDS of grids into your GFE grid manager – and you'll probably crash the program – so it pays to think a bit about what you really want to display. If you choose to load multiple parameters and multiple models, the program will group by parameter first, then by "forecast hour" or "run time" depending on what you have specified in the "Group by" section, and finally by model.



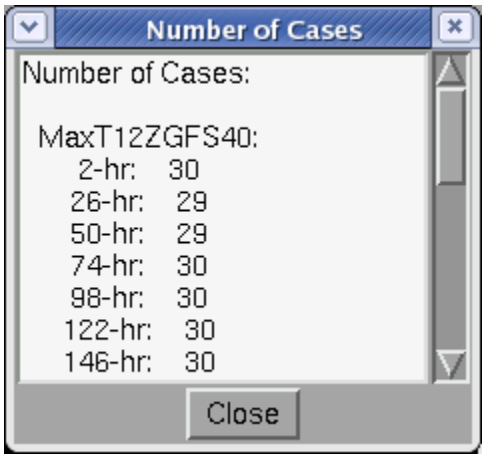
For example, at the right we show the grid manager after asking for displays of 12 UTC MaxT and MinT forecasts from the GFS40 as well as the Official, for the three days ending on Sept. 24th and group by "Run Time". First all the MaxT grids are shown at the top, with the MinT grids shown at the bottom. The MaxT grids are then grouped with all the grids from the 12 UTC runs from the 24th shown at the top, and grids from the 12 UTC runs from the 22nd at the bottom. Finally within those, the grids from the Official grids are shown above the grids from the GFS40 model. Though the labels in the grid manager can get very long, they give you all the information you need to understand the source of the grid that is displayed.

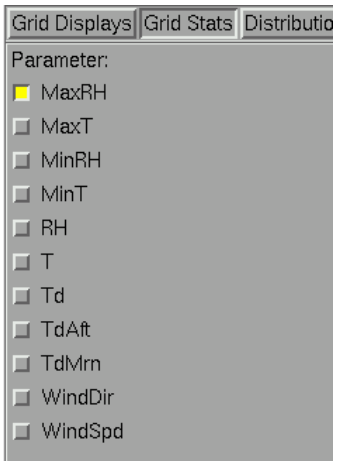
Grid Stats Tab

The Grid Stats tab is used to calculate and display statistics as a grid. Essentially the statistic is calculated at every gridpoint independently – and just displayed as a grid. The statistics are calculated over the time period or dates that are specified in the rest of the GUI. Unlike the Grid Displays tab, these statistics are not valid at a PARTICULAR time, but were calculated from many times. Thus there is some question of 'where' to load these grids into the grid manager. We have decided to load the statistics in locations that line up with forecast time from the current models. Examples below will explain this further, but it is thought that this is usually "what you want". For example, when you display the 24hr temperature forecast bias, it will be displayed at the time that corresponds with the current 24hr temperature forecast. If you see significant biases, you will likely want to modify the forecast at that time by the amount shown in the statistics grid – and it makes sense to have it located at the same time in the grid manager.

One of the difficulties with loading grids into the grid manager in this way is that there is no way to indicate how many grids were actually used in calculating the statistic. For example, if you are not using the 'Common Cases' option, the system may have found 20 grids for one average, but only 2 grids for another. They would look exactly the same in the grid manager, and in the displays.

To help with this problem, the BOIVerify system creates a small window with case information when it makes Grid Stats displays. By default it places this window in the upper-right portion of the screen, but you can move it to wherever you want. If you click on the "Number of Cases" button, the window expands into a display of the number of cases for each displayed grid:



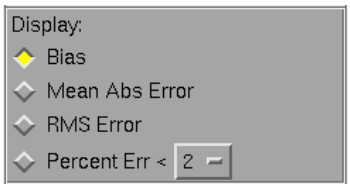


Parameter – This is very similar to the choice of parameter in the Grid Displays tab. You can choose to load statistics from multiple parameters at the same time, to speed up your ability to interpret the displays. Note, however, that the vector element of Wind has been split up into WindDir and WindSpd for the statistics.

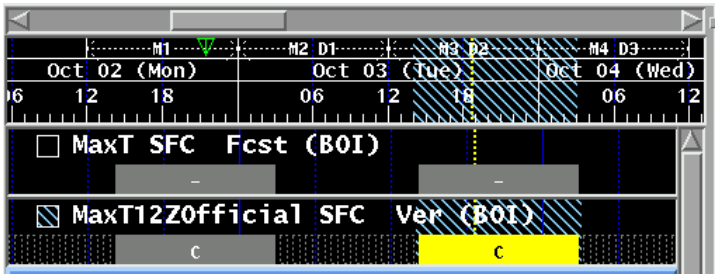
As with the Grid Displays tab, you can specify many different statistics from many different models and you might get more grids than you would like. It pays to think a bit about what you want to

display before you start asking for everything!

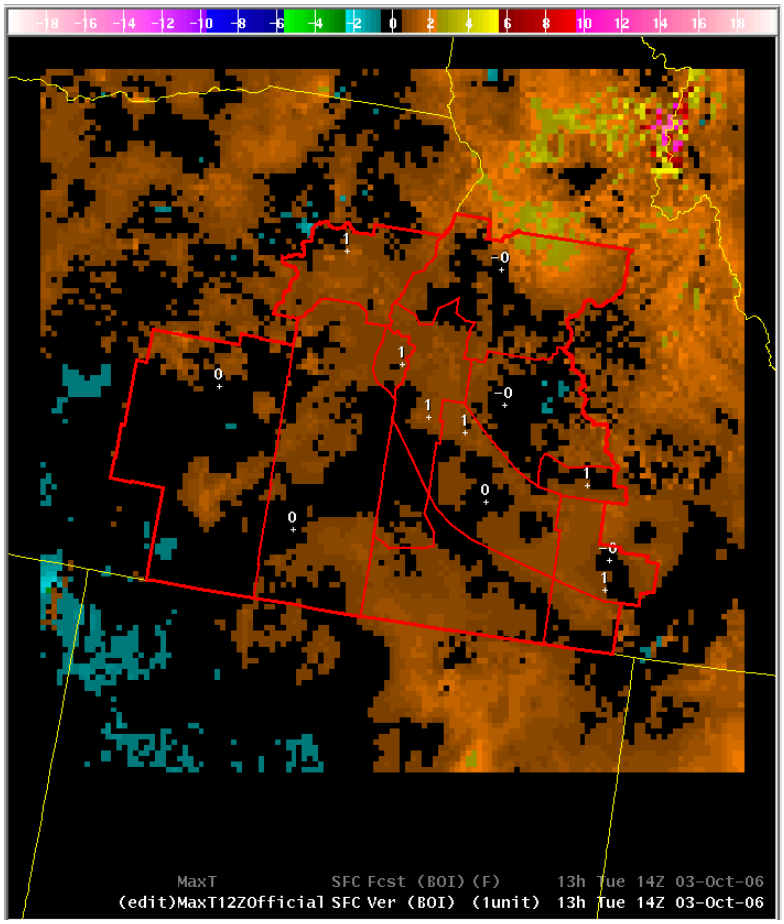
Display – This is where you specify the statistic to display.



The **Bias** is simply the "average error" or "mean error" over the time period specified in the rest of the GUI. For example, here is the grid manager after calculating the Bias errors from 12 UTC Official forecasts for the 30 days ending on Sept. 30th:



Note that this was run at about 21 UTC on Oct. 2nd. The Fcst MaxT grid was already in the grid manager and was kept, but other grids were removed, and the Bias Error from 12 UTC Official forecasts were loaded on the line labeled "MaxT12Zofficial Ver". The highlighted bias error grid looks like:

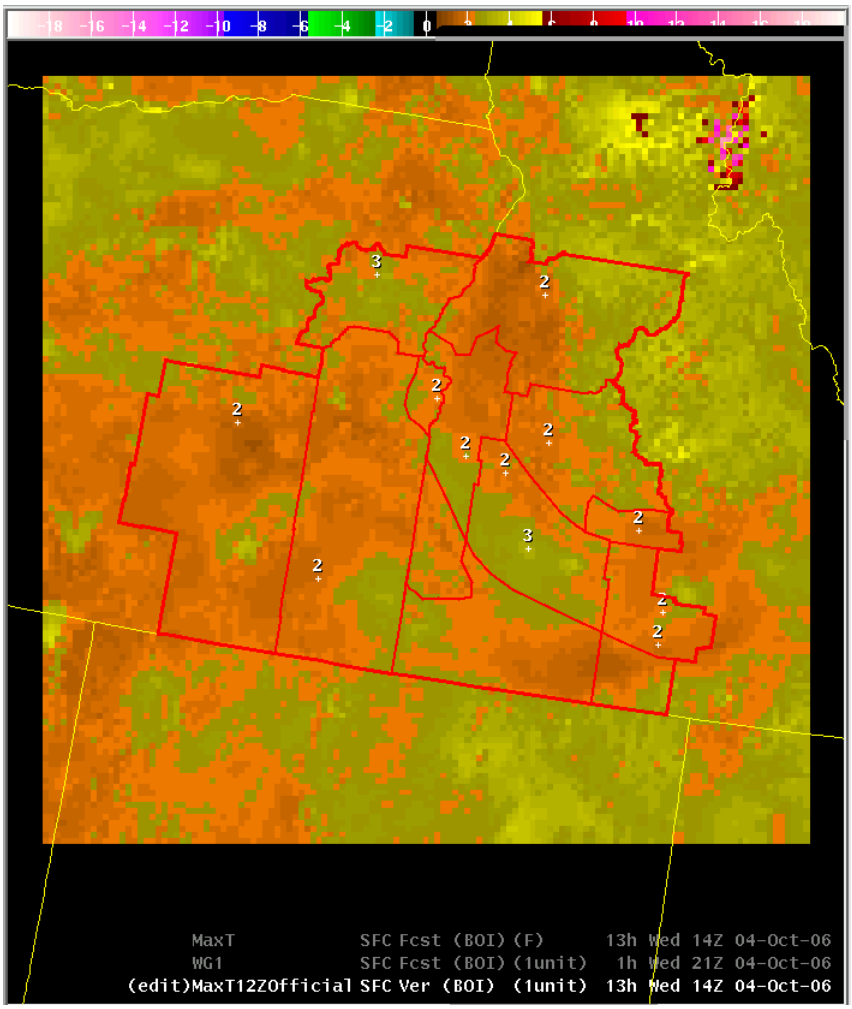


This is the Bias error of 26hr MaxT forecasts that verified during the month of September

(verifying the 30 days ending on the 30th). Many forecasters would refer to this as the “day 1” forecast error, since the MaxT forecast for the “next day” is the first MaxT forecast made from the 12 UTC cycle. Note that it was loaded in the grid manager right under the time where the current “day 1” MaxT forecast grid is located.

As with the Grid Displays of forecast errors, the default color curve is set to use “warm colors” of oranges, yellows, and reds, for positive forecast errors, or where forecasts have been, on average “too warm”. You’ll see in the display above that average errors inside our forecast area are less than about 1 or 2 degrees, except for a small area along the northern border where errors tend to be about 3 degrees too warm. The large errors to the north, outside of our forecast area, are probably because of a bad observation in that area, that we do not take the time to remove during our quality control procedures. You’ll also notice that over nearly the entire forecast area the errors were small, but were almost all positive. If the forecaster believes that this bias error over the past 30 days is likely to occur again in the current forecast, they can remove it by subtracting this bias grid from the current “day 1” MaxT forecast grid.

The **Mean Abs Error** (or MAE) is simply the average of the absolute value of the error. Averaging an error of +5 degrees with -5 degrees would produce a bias of 0 degrees, but the Mean Abs Error would be 5 degrees. In this way, the Mean Abs Error is a measure of the ‘size’ of the errors, rather than whether they tend to be positive or negative. Here is a display of the Mean Abs Error for 26-hr Official MaxT forecasts from 12 UTC that verified in September (the same time period shown above for the bias):

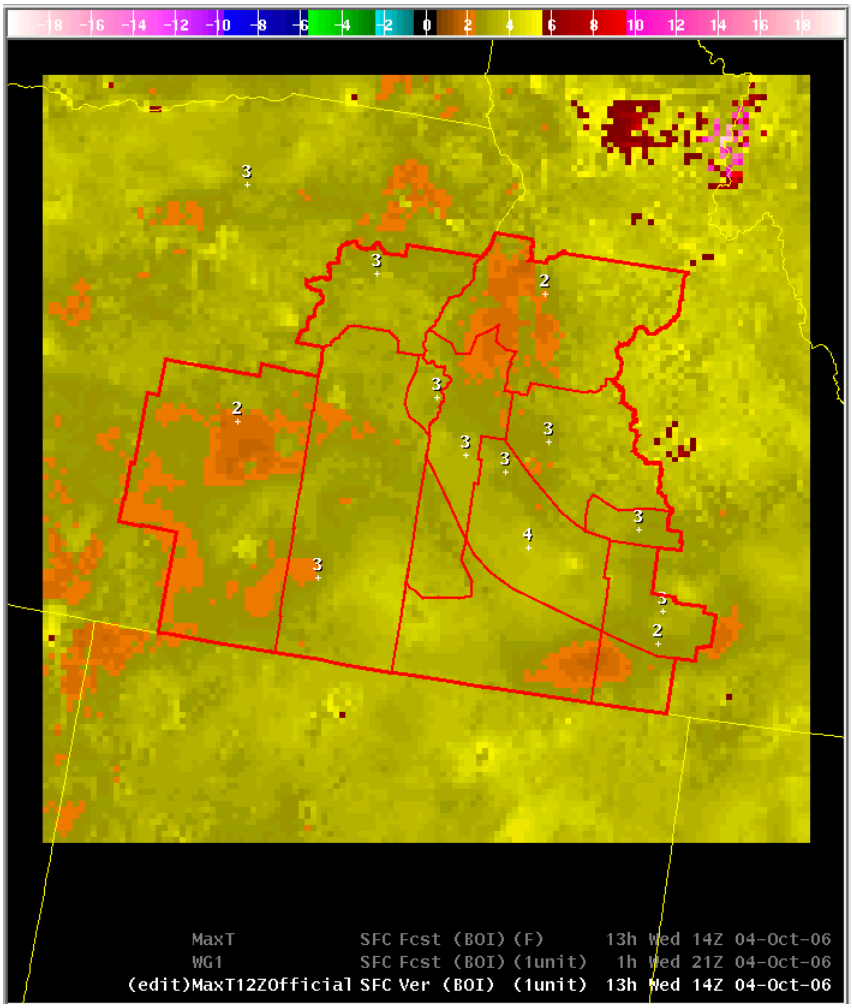


To simplify the displays, the color table used is the same as for the bias errors, and the range of values is the same, even though negative values of MAE are not possible. Comparing this display to the Bias above, you'll see that there are many locations where the bias is near zero, but the MAE is around 2 degrees, this just means that errors are typically around 2 degrees, but are just as likely to be too high, as they are to be too low.

The **Mean Squared Error** (or MSE) is simply the average of the square of the error. Like the MAE, this is always positive, and is a measure of the 'size' of the errors, not whether they tend to be positive or negative. However, because the errors are squared before being averaged, the large errors, or 'big busts', tend to have a much larger influence over the final value. This is often desired, since perception often seems to emphasize a single large error more than lots of small errors.

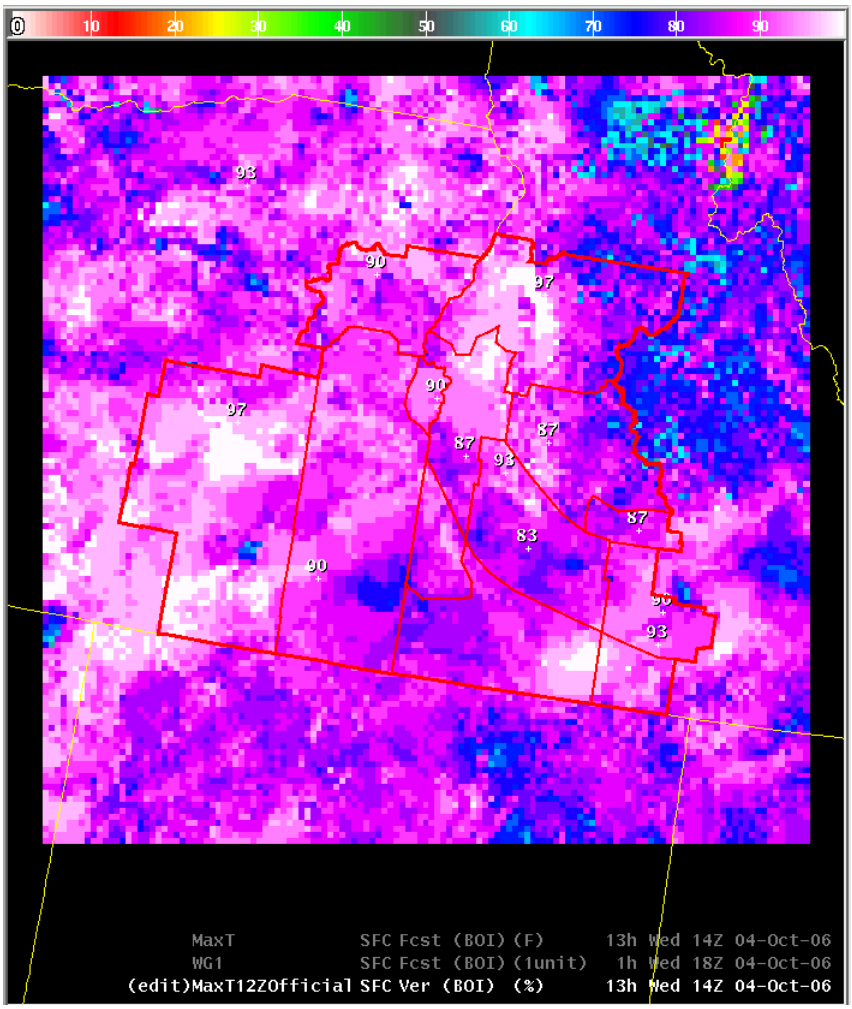
When MSE is calculated for probabilistic parameters, it is often called the Brier Score. The Brier Score can be shown to be 'unbiased' and 'proper' for probability forecasts. Other scores are not always proper, in that scores can be artificially inflated with no forecast knowledge. For example, if you just looked at the MAE for probability forecasts, and the climatological frequency of the event was less than 50%, then it can be shown that blindly forecasting 0% probability all the time will give you a better MAE score, rather than forecasting the climatological frequency of the event. The MSE, or Brier Score, does not suffer from this deficiency.

The **RMS Error** is simply the square root of the mean squared error. Like the MSE, it tends to emphasize large errors, but the units are the same as the forecasts, and the values tend to make more 'sense' when thinking about making new forecasts. Here is the RMS error of 26-hr MaxT Official forecasts from 12 UTC that verified in September (the same time period shown above for the MAE and the bias):



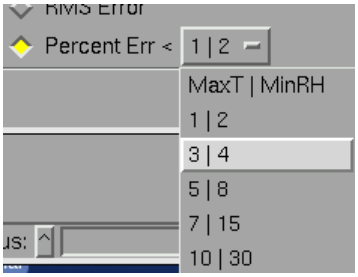
Note that the RMS error is larger than the MAE, but has a similar pattern. Again, for simplicity, the color table and range of values used is the same as for the Bias, MAE, and MSE, even though negative values are not possible.

The **Threshold** statistic, allows you to see the percentage of forecasts when the absolute value of the error is below certain thresholds. You choose the threshold in the "Percent Err <" selector. Each parameter has five site-configurable thresholds. For example, here is the percent of 26-hr MaxT Official forecasts from 12 UTC that had errors less than 5 degrees and verified in September (the same period shown above for bias, MAE and RMS statistics):



Note that the range of values is from 0 to 100%. The default color table for these percentage displays is “Warm to Cold”, but is also configurable. Note that the MAE, RMS, and Thresholds, all give similar information – but the differences can help you figure out where you are having particular problems.

Choosing the threshold can get a little confusing if you have more than one parameter selected. Since each parameter can have a different threshold, you might be choosing a different threshold for different parameters. For example, we have thresholds for MaxT set to 1, 3, 5, 7, and 10, while for MinRH I have them set to 2, 4, 8, 15, and 30. When we have both MaxT and MinRH selected under the Parameter section, the “Percent Err <” switches to showing the different thresholds with a “|” character between them. When you pull down the selector, the top entry is a label, showing you which threshold value corresponds to each parameter.



Many **Two Category Scores** are now available in BOIVerify. Almost all forecasts can be broken down into a simple two-category, or yes/no forecast. For example, a temperature forecast can be broken down into whether the forecast temperature is above or below 32 degrees. The observations can be broken down in the same way

The results from all the gridpoints can be expressed in a simple yes/no matrix. If a condition is forecast to occur, and it does, this is often called a **hit**. If the condition is forecast, but it does occur in the observations, this is often called a **false alarm**. Similarly, if the condition is *not* forecast, but it does occur, this is called a **miss**. And, if the condition is not forecast, and it does not occur, this is called a **correct negative**. The results are often plotted in a 2 x 2 matrix, or contingency table:

		Observed	
		Yes	No
Forecast	Yes	Hits	False Alarms
	No	Misses	Correct Negatives

Then, various scores are computed from the values in the contingency table. These scores are well documented elsewhere, and we'll only briefly describe them below. A nice website with discussion of these scores, in addition to many other verification issues, can be found at:

http://www.bom.gov.au/bmrc/wefor/staff/eee/verif/verif_web_page.html

and I have shamefully copied some of their nomenclature and descriptions here.

For almost every score, BOIVerify also includes an 'areal' alternative. When considering the 'scale box', normally the forecast or observed value at each gridpoint, is calculated by averaging over all the gridpoints within the 'scale box'. When the scale box is set to it's smallest value, then there is only one gridpoint to consider inside that box, and the score is made using just the simple gridpoint-by-gridpoint calculation of the hits, misses, false alarms, etc. However, when the scale box is set larger, the forecast or observed value at each gridpoint will be the average over a 3x3 box of gridpoints, or a 5x5 box, or a 7x7 box, etc. This averaging procedure tends to smooth the forecast and observed grids, and can give some information about the larger scale skill of the forecast.

However, the smoothing also lowers the extreme values on the grid. For example, if you had a QPF forecast with a maximum of 1.25 inches at one gridpoint, and you wanted to use one of the Two Category scores at a larger scale, the smoothed forecast grid would ***not*** have the value of 1.25 inches at any gridpoint. Depending on the size of the 'scale box', and the spatial distribution of the QPF forecast gridpoints, the smoothed value could be substantially less.

Thus, if we calculated Two Category scores for QPF values greater than 1.25 inches, there would be no forecasts of that amount!

Thus, for almost every Two Category score, BOIVerify also includes an 'areal' alternative. For these scores, rather than averaging over the larger 'scale box', the scale box is 'searched' to see if the condition exists ***anywhere*** within the scale box. This search is performed both on the forecast grid, and on the observed grid. In our example above, if we are using a scale box which means that, say, a 7x7 box of gridpoints is being considered, then gridpoints near our 1.25 inch gridpoint will also have a *forecast* of 1.25 inches. And if there are observed gridpoints nearby, *but not necessarily at the same gridpoint*, with observed values of 1.25 inches, then we will still get a *hit* for those gridpoints. The results give 'smoother' results spatially, but still relate to the 'real' forecast and observed values, rather than smoothed values.

This averaging, or searching, over a 'scale box', is sometimes referred to as 'neighborhood' scoring, or 'fuzzy' scoring. There are many variations, where more sophisticated measurements of the distribution of values inside the 'neighborhood' or 'scale box' are considered, and where 'scale circles' are used, rather than boxes. Here, we use boxes because the calculations are much faster. There are certainly other, more sophisticated ways to measure how forecasts perform at different scales, but this 'neighborhood' approach is rather simple and easy-to-understand, yet effective.

The current 'scores' available in BOIVerify include:

Hits – This isn't a score at all, but just displays the number of hits found in the contingency table for each gridpoint.

Misses – This just displays the number of misses found the contingency table for each gridpoint.

False Alarms – This just displays the number of false alarms found in the contingency table for each gridpoint.

Correct Negatives – This just displays the number of correct negatives found in the contingency table for each gridpoint.

Fraction Correct = Answers the question, what fraction of the forecasts were 'correct'. Ranges from 0 to 1, with a perfect score being 1.

$$\text{fraction correct} = \frac{\text{hits} + \text{correct negatives}}{\text{total}}$$

Frequency Bias – Answers the question, how did the frequency of "yes" forecasts compare to the frequency of "yes" events. Ranges from 0 to infinity, with a perfect score being 1. Bias greater than 1 indicates 'over forecasting'. Bias less than 1 indicates 'under forecasting'

$$\text{frequency bias} = \frac{\text{hits} + \text{false alarms}}{\text{hits} + \text{misses}}$$

Probability of Detection (POD) – Answers the question: what fraction of the observed "yes" events were correctly forecast? Ranges from 0 to 1, with a perfect score being 1.

$$\text{POD} = \frac{\text{hits}}{\text{hits} + \text{misses}}$$

False Alarm Ratio (FAR) - Answers the question: what fraction of the predicted "yes" events actually did not occur? Ranges from 0 to 1, with a perfect score being 0.

$$\text{FAR} = \frac{\text{false alarms}}{\text{hits} + \text{false alarms}}$$

Probability of False Detection – Answers the question: What fraction of the observed "no" events were incorrectly forecast as "yes"? Ranges from 0 to 1, with a perfect score being 0.

$$\text{probability of false detection} = \frac{\text{false alarms}}{\text{correct negatives} + \text{false alarms}}$$

Threat Score (also called the Critical Success Index, or CSI) – Answers the question: How well did the forecast "yes" events correspond to the observed "yes" events? Ranges from 0 to 1, with a perfect score being 1.

$$\text{threat score} = \frac{\text{hits}}{\text{hits} + \text{misses} + \text{false alarms}}$$

Equitable Threat Score - Answers the question: How well did the forecast "yes" events correspond to the observed "yes" events (accounting for hits due to chance)? Ranges from -1/3 to 1, with a perfect score being 1.

$$\text{ETS} = \frac{\text{hits} - \text{hits}_{\text{random}}}{\text{hits} + \text{misses} + \text{false alarms} - \text{hits}_{\text{random}}}$$

where

$$\text{hits}_{\text{random}} = \frac{(\text{hits} + \text{misses})(\text{hits} + \text{false alarms})}{\text{total}}$$

True Skill Score - Answers the question: How well did the forecast separate the "yes" events from the "no" events? Ranges from -1 to 1, with a perfect score being 1.

$$\text{TSS} = \frac{\text{hits}}{\text{hits} + \text{misses}} - \frac{\text{false alarms}}{\text{false alarms} + \text{correct negatives}}$$

Heidke Skill Score - Answers the question: What was the accuracy of the forecast relative to that of random chance? Ranges from minus infinity to 1, with a perfect score being 1.

$$\mathrm{HSS} = \frac{(\text{hits} + \text{correct negatives}) - (\text{expected correct})_{\text{random}}}{\mathrm{N} - (\text{expected correct})_{\text{random}}}$$

where

$$(\text{expected correct})_{\text{random}} = \frac{1}{\mathrm{N}}\left[\begin{array}{l}(\text{hits} + \text{misses})(\text{hits} + \text{false alarms}) + \\ (\text{correct negatives} + \text{misses})(\text{correct negatives} + \text{false alarms})\end{array}\right]$$

Odds Ratio - Answers the question: What is the ratio of the odds of a "yes" forecast being correct, to the odds of a "yes" forecast being wrong? Ranges from 0 to infinity, with a perfect score being infinity.

$$\mathrm{OR} = \frac{\text{hits} * \text{correct negatives}}{\text{misses} * \text{false alarms}}$$

In BOIVerify, you pick the score you want calculate from the list of scores. Then you pick whether the condition is less than (<), less than or equal to (<=), greater than (>), or greater than or equal to (>=) by selecting from the condition menu. You then enter in the threshold value in the box.

Distributions Tab

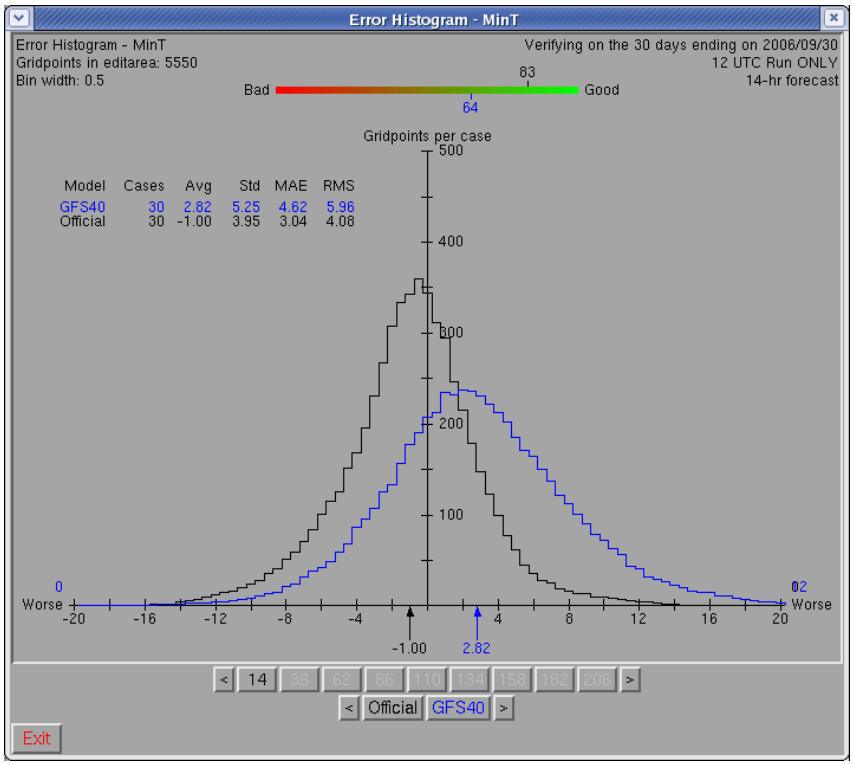
The distributions tab is used to calculate and display histograms of errors or scatterplots over a set of gridpoints (the currently active editArea in GFE, or ALL gridpoints if there is no active editArea selected). The histograms and scatterplots are displayed in a window separate from the GFE display. It is worth noting that these distributions and scatterplots combine values over space AND time. They tell you nothing about WHERE the errors occur spatially, just the size and distribution of the errors.

Parameter – This is very similar to the choice of parameter in the Grid Displays and Grid Stats tabs. However, here you can only choose a single parameter at once. Thus, there is currently no way to overlay a histogram or scatterplot from one parameter on top of another. Also note that, like with the Grid Stats tab, the vector parameters are broken into the direction and speed components.

Display – There are four different display types, which you choose from the Display section. Error Histograms display errors in a histogram. Value Histograms display both forecast and observed *values* as a histogram. The Expected Value displays show the min, max, and average observed values for various forecast values. For probabilistic parms, the Expected Value display is also called a Reliability diagram. Scatterplots plot forecast/observed values on the x/y axis.

Error Histograms

When you choose the Error Histogram display, the distribution of errors are displayed as a histogram. For example, here is the histogram display of MinT forecast errors made from 12 UTC, verifying during the month of September, over a particular edit area (in this case, all the gridpoints within our CWA) for both the Official forecasts, and the GFS40 model:

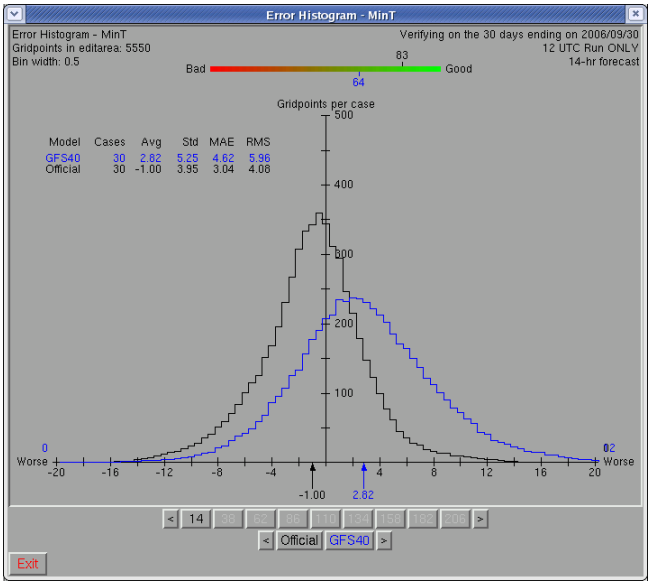


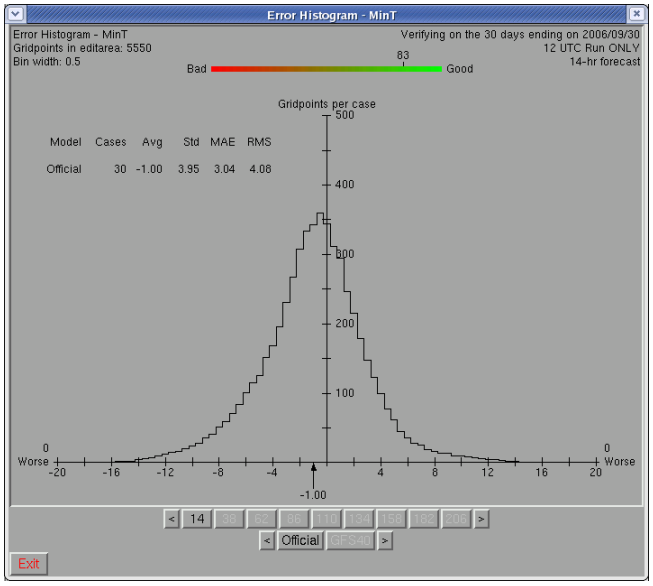
The histograms are drawn in the top portion of the display, while the buttons at the bottom control the graphs that are overlaid.

First, let's look at the buttons at the bottom of the display that control the overlays. The buttons are split into two sections: one for the forecast times, and one for the models. If there are too many buttons to fit on a single row, they are broken up into multiple rows, such as in this display for temperature errors:

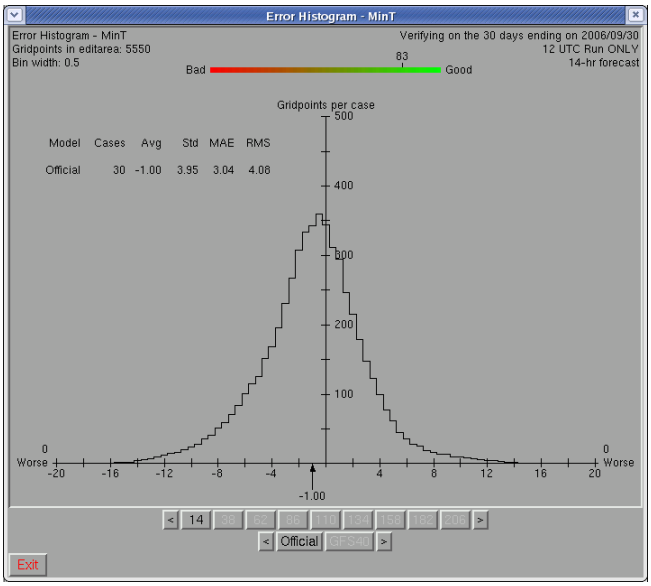
Each of the buttons is a "toggle" that allows you to turn on and off a graph for that time/model. The buttons for graphs that are turned off are shown in a dull grey color.

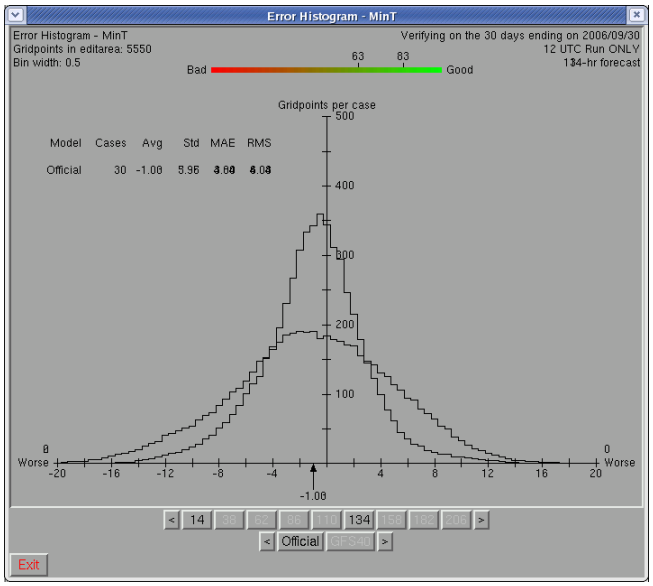
For example, we could toggle on and off the GFS40 model histogram at 14 hours by clicking on the GFS40 button:





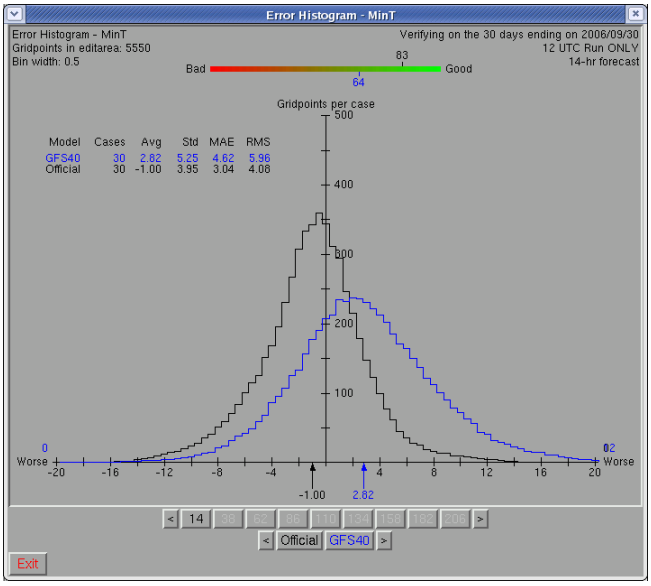
We can also use the forecast hour buttons as toggles. For example here we show the temperature histograms from the Official forecast for both the 14hr forecast and the 134hr forecast:

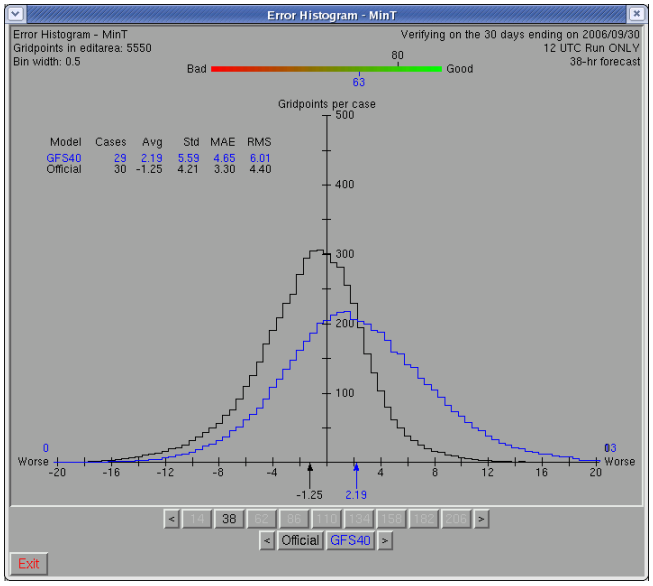




Note that some of the legends are displayed on top of each other, and the graphs are drawn in the same color, which makes this a fairly difficult to interpret display. Same model, multiple time displays aren't used very much.

However, rather than toggling individual times off and on, the "arrow" buttons at left and right of the forecast hour buttons are very helpful for 'looping' or 'stepping' through the graphs for various forecast times. For example, here we show the difference between two displays of Offical and GFS40 histograms, one for the 14hr forecast and one for the 38hr forecast. All we clicked on to switch these displays was the "arrow" button to the right of the forecast time buttons:



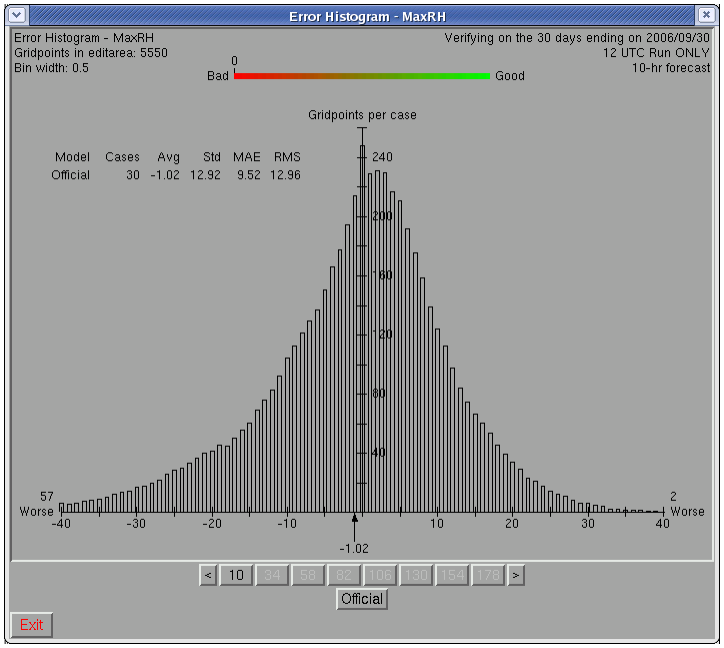


You will find that these “stepping” buttons are a very efficient way to look at a lot of histograms very quickly.

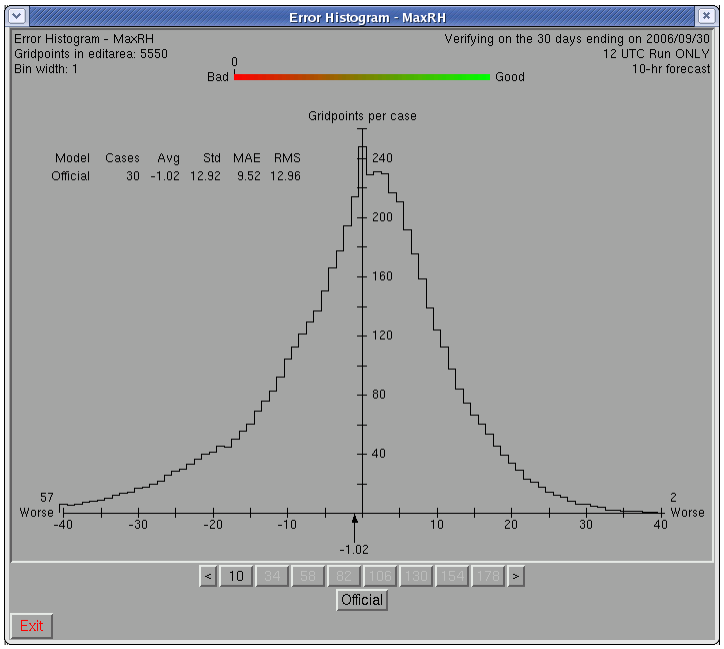
Now let’s look at the various parts of the histogram display:

In the **upper-left corner** there is information about the graph and edit area. The top line tells you the parameter that is diplayed. The total number of points in the edit area is listed on the second line. The size of the bins for the histogram is listed on the third line. The histogram bins are centered about zero. For the example shown here, where the bin width is 0.5 degrees, the bin around zero goes from -0.25 to +0.25 degrees. The next bin on the positive side is +0.25 to +0.75. The next positive bin is from +0.75 to +1.25, etc.

Displays can be difficult to overlay if you have a bin width that is smaller than the smallest potential error differences. For example, here is a display of MaxRH errors with a bin width of 0.5:

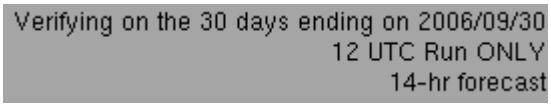


Notice the “picket fence” appearance of the histogram. The reason for this is that MaxRH values are stored as integer numbers of percent. Thus, the only possible error values are 0, +/-1, +/-2, etc. You cannot get an error of 0.5 percent. Note however, that with a bin width of 0.5, the bin around zero goes from -0.25 to +0.25 and has lots of gridpoints that an error value of 0. However, the next bin on the positive side goes from +0.25 to +0.75, and no gridpoints can have such an error. Thus, that bin has ZERO, and the reason for the “picket fence” appearance. The bin width is configurable for each parameter, so that you can change it if your site has decided to save different parameters with different levels of precision. For example, here is the same histogram as above, but using a bin width of 1.0:



This display is much easier to interpret, and looks much better when different models are overlaid.

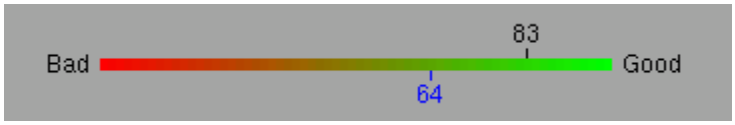
In the **upper-right corner** of the display is



information about the dates and times for the displayed graphs. The top line describes the dates. If you have chosen a time period, it will list the length of the period and the last day, as shown here. If you have chosen specific days, the software will try to list them. However, if there are many days, there may not be enough room to list them all, and the display will simply say “various dates” instead. The second line shows the cycles chosen for display. The third line shows the forecast hour of the currently displayed histogram line (see below how you can change which forecast hour is shown).

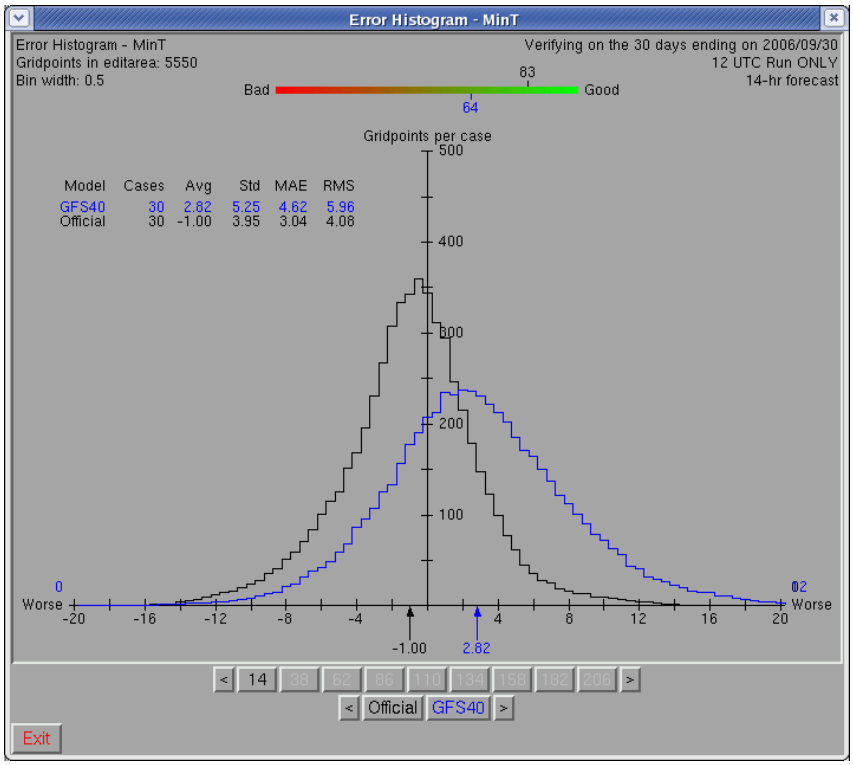
Model	Cases	Avg	Std	MAE	RMS
GFS40	30	2.82	5.25	4.62	5.96
Official	30	-1.00	3.95	3.04	4.08

Various statistics on the errors are displayed in a table at the top-left of the graph. For each model, it shows the number of cases or grids, the average error or bias (Avg), the standard deviation of the errors (Std), the Mean Absolute Error (MAE) and the Root Mean Squared error (RMS). Keep in mind that for the Grid Stats displays we calculated some of these statistics ***at each gridpoint*** and displayed them as a grid. Here, we are calculating the statistics over ***many gridpoints and many times***, and showing them in this table (as well as showing the distribution of errors in the histogram). The standard deviation is the typical statistical measure of the “spread” of a distribution. It is worth pointing out that if the average error was zero, then the standard deviation would be the same as the RMS error. Thus, in some ways, the RMS error is a combination of the measure of the “spread” of the distribution, as well as the location of the distribution about zero (its bias). The MAE is also a combined measure in this sense, but, as discussed above, it does not give as much weight to ‘big busts’.



Simple Score: Since many people want only a single measure of “goodness”, a “score” is highlighted at the top of the graph along a line labeled “Bad” to “Good”. The score shown here is 100 minus the RMS error squared, or 100 minus the Mean Squared Error. A perfect forecast would have a “score” of 100 and be plotted at the right edge labeled “Good”. If the RMS error is 10 or larger, the “score” would be zero and would be plotted at the left edge labeled “Bad”. Anyone familiar with statistical distributions should understand that no single score fully describes the error distribution. This “score” is simply ***one*** that combines ***a*** measure of spread and bias in a reasonable way. The scores are labeled in the same color that is used for their histogram display. The score from the Official forecasts is always shown above the line, while other models are shown below the line.

The range of errors shown on the histogram is the same as the range of errors shown in the Grid Displays and Grid Stats displays. For example, in this distribution of MinT, the errors are shown between -20 and +20 degrees:



If there are errors larger or smaller than these limits, they are shown above the “Worse” labels at the edge of the graph. The average error for each model is highlighted at the bottom of the graph axis by the colored arrow. The height of the graph is determined by the number of errors for the edit area that is chosen. The number of errors in each bin is actually divided by the number of cases (or number of grids) so that the height of the histogram can be interpreted as the “number of gridpoints per case” in each bin. The largest value among all forecast times is used to set the height of the graph, so that as you move to other forecast times, the height of the graph stays constant.

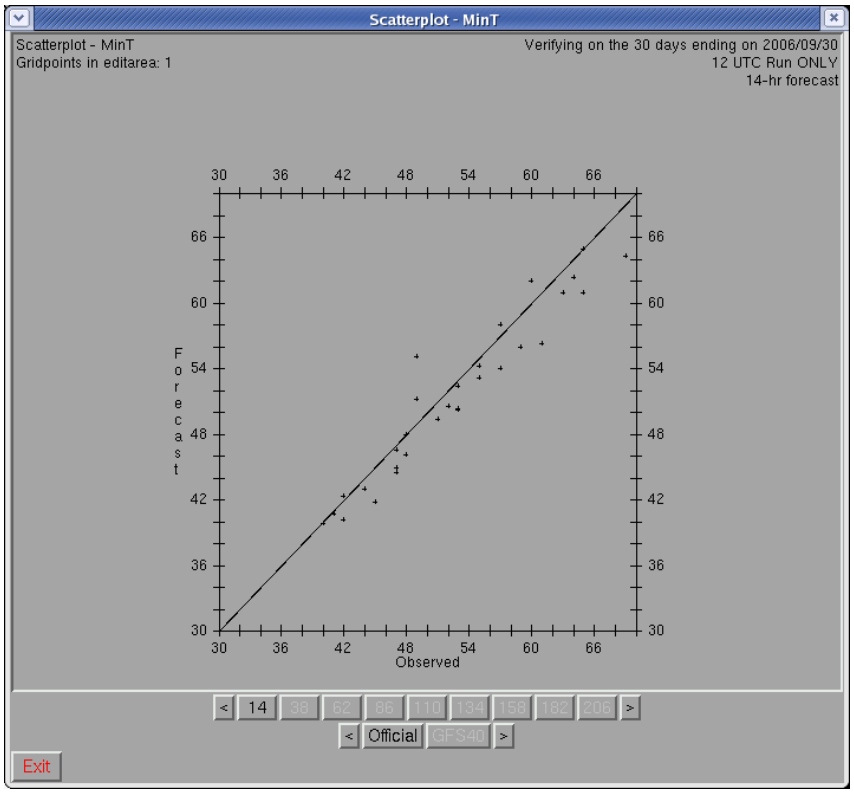
Value Histograms

Expected Value

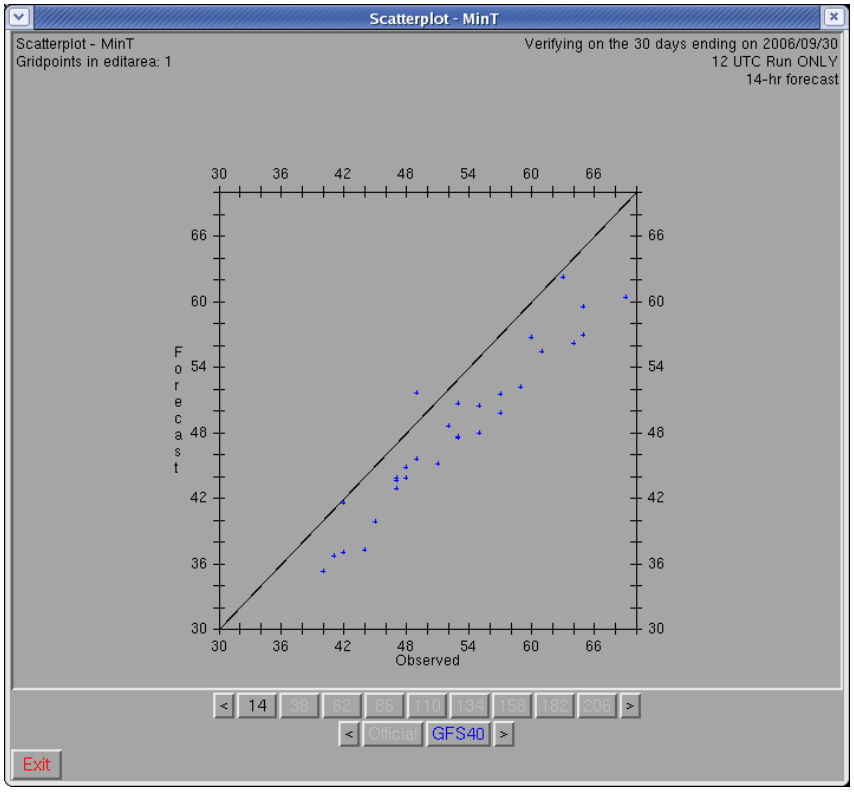
Scatterplots

A scatterplot is simply an x-y plot of points where the observed value is used for the x-coordinate and the forecast value is used for the y-coordinate. Perfect forecasts would line up along a diagonal line from lower-left to upper-right where the forecast and observed values are equal. Forecasts that are “too high” show up “above” the diagonal line, and forecasts that are “too low” show up “below” the diagonal line.

Here, as an example, we have chosen a single gridpoint, and are showing the scatterplot of 14-hr Official Forecasts from 12 UTC that verified within September:



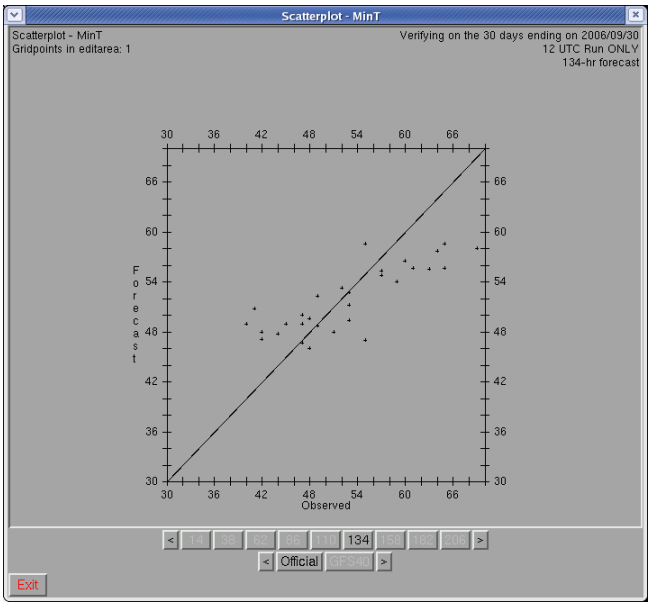
The forecasts look pretty good. They line up nearly along the diagonal. There appears to be a slight low bias at this particular point because more of the points are below the line than above it. We can use the toggle buttons to display the same display for the GFS40 model:

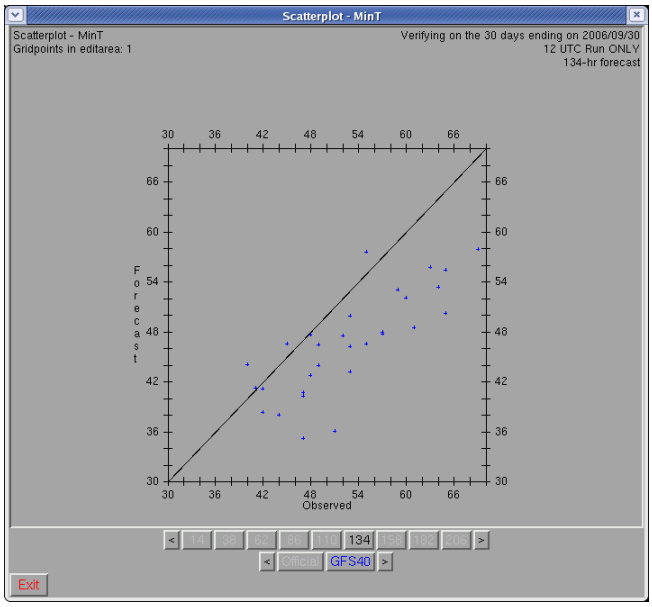


Here a very pronounced cold bias is evident, so we would need to be very careful when using the model forecast for this particular point. However, if we could account for the bias, the

model is well behaved with the points laying nearly along a diagonal.

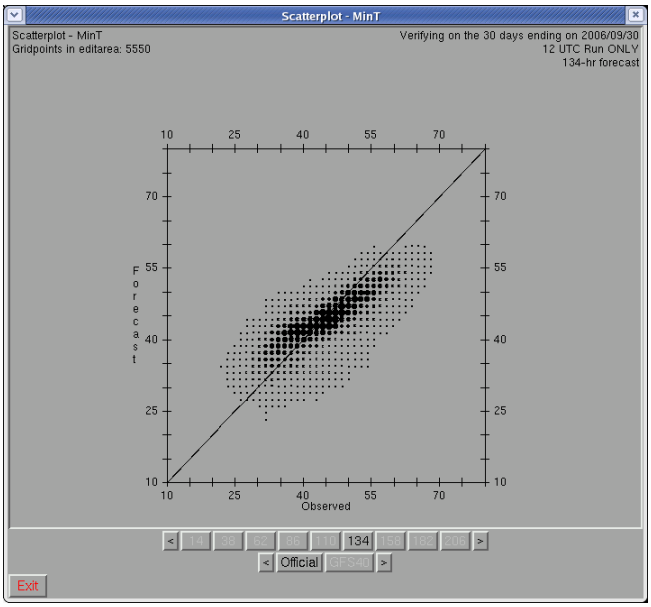
Contrast the scatterplots above with the same scatterplots for 134 hours (the “day 6” MinT):

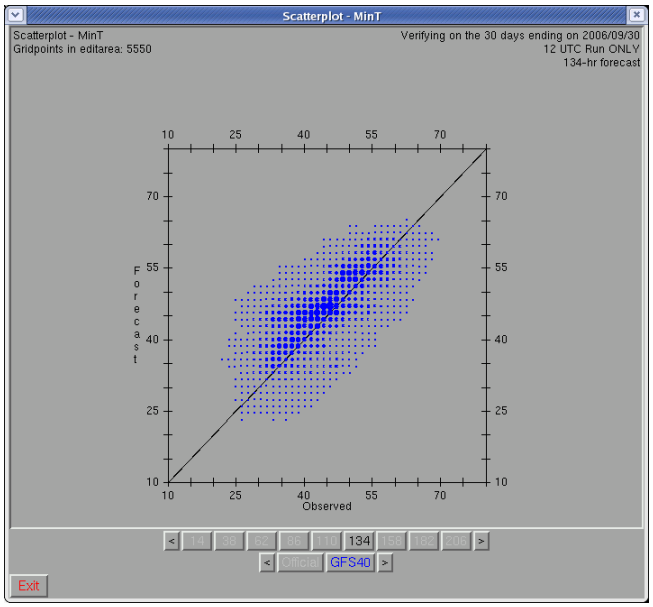




Interestingly, the Official forecasts appear pretty “conservative”. When the forecasts were relatively cool during this period, they tended to be on the warm side. In other words, they weren’t “cool enough” on the cool days. On the other hand, when the forecasts were relatively warm, they tended to be on the cool side – they weren’t “warm enough!” Such ‘conservative’ forecasts are not all that unreasonable at day 6, but it is interesting to compare to the GFS40 in this case. The model forecasts, while still exhibiting a cool bias, do not appear to be as ‘conservative’ as the Official forecasts. This may help the forecasters understand that we can be a bit more ‘risky’ in our day 6 MinT forecasts at this point, and that other than the cool bias, the model forecasts are a bit more realistic in their depiction.

The scatterplot displays shown so far have been for an edit area that was only a single gridpoint – so that we could see the individual forecast values. If you use an edit area with thousands of points, over many forecast times, the display could easily have millions of points to draw, which would be very time consuming. Thus, when the number of points to display gets above 1000, the scatterplot displays change into “thinned” scatterplots, where circles are used and the SIZE of the circle indicates the relative number of points at those coordinates. For example, here is the same comparison as above, but for the entire CWA:





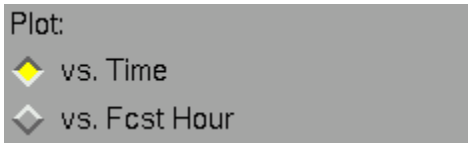
It is interesting that the same behavior is evident for the Official forecasts, even when now looking at 5550 gridpoints. The relatively cool forecasts tended to work out “too warm”, while the relatively warm forecasts tended to work out “too cold”. The Model on the other hand was slightly better, though it had a slight overall warm bias (which is very interesting, since it had a cool bias at the one individual point we looked at above). Such displays can be very helpful for determining how to modify forecasts in different situations.

Point / Area Stats

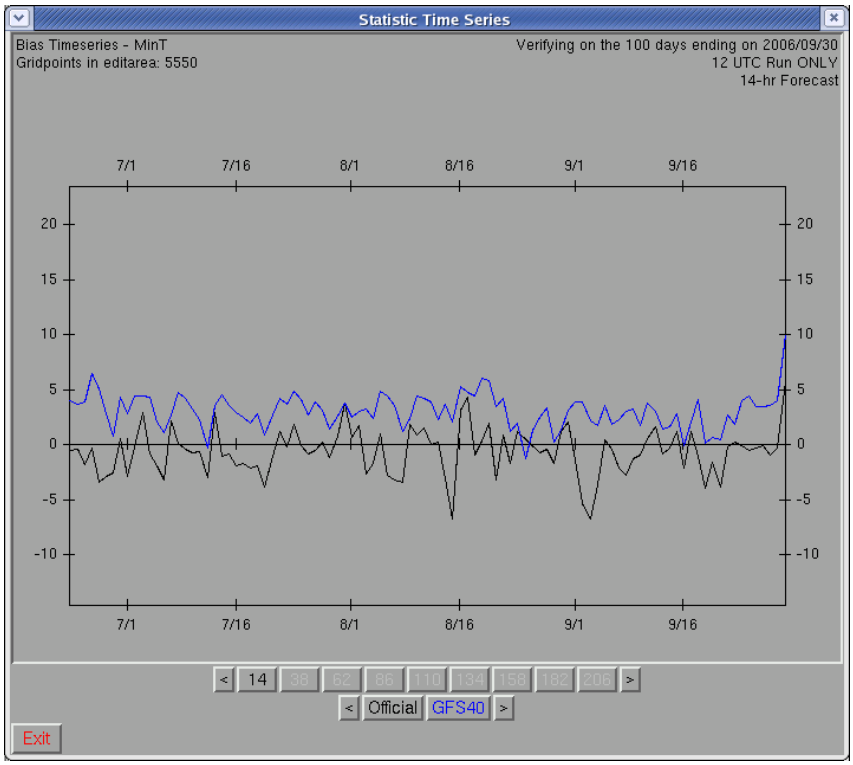
The “Point / Area Stats” tab displays time series of statistics that are automatically calculated for pre-selected edit areas. These statistics can be saved for a much longer time period than the grids themselves, which allows you to track long-term trends of various statistics. A program runs in the background calculating these statistics for all models, all times, and all configured edit areas. This program is new to version 1.0 of the BOIVerify software, so you may not find these graphs very useful until they have run for a few days or weeks, to build up enough points to make meaningful graphs.

Parameter - Once again, you choose the parameter or parameters that you would like to display in this section. Like the “Grid Displays” and “Grid Stats” tab, you can choose multiple parameters to display at the same time (although that may not always be possible – as discussed below). Also, like the other statistics displays, vector elements are broken up into their speed and direction components.

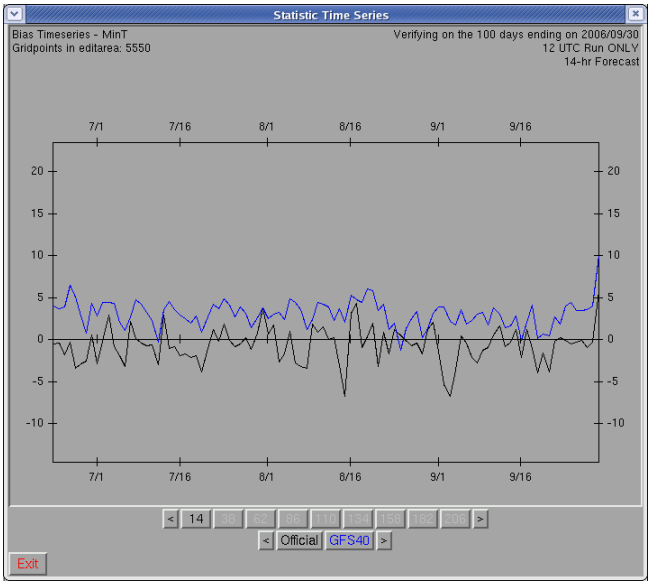
Plot vs. Time or Plot vs. Fcst Hour – To plot time series of statistics, choose “vs. Time”. This will display the time series over the selected time, and the toggle buttons will allow you to display the different time series for different forecast hours. On the other hand, if you choose “vs. Fcst Hour”, the statistics are averaged over the selected time, and displayed on a graph versus forecast time. In this case, the toggle buttons will allow you to display the different time series for different edit areas or parameters.

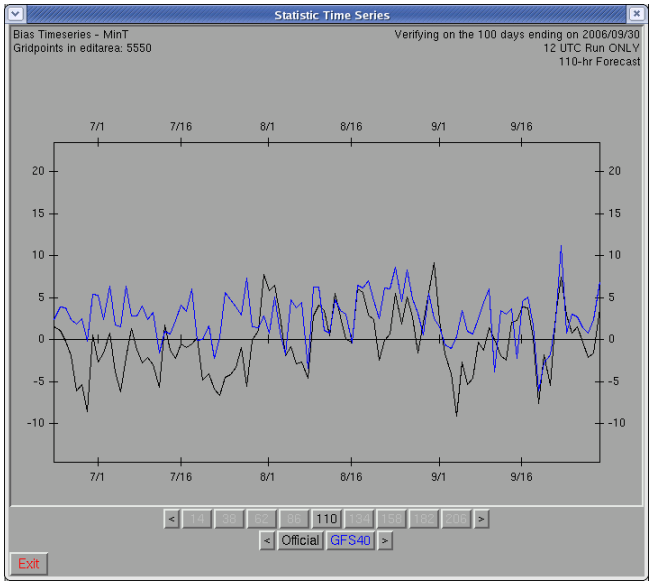


For example, here is a display of the time series of MinT mean forecast errors (bias) for the 100 days ending Sept. 30th, for both the Official and GFS40 models from the 12 UTC runs:



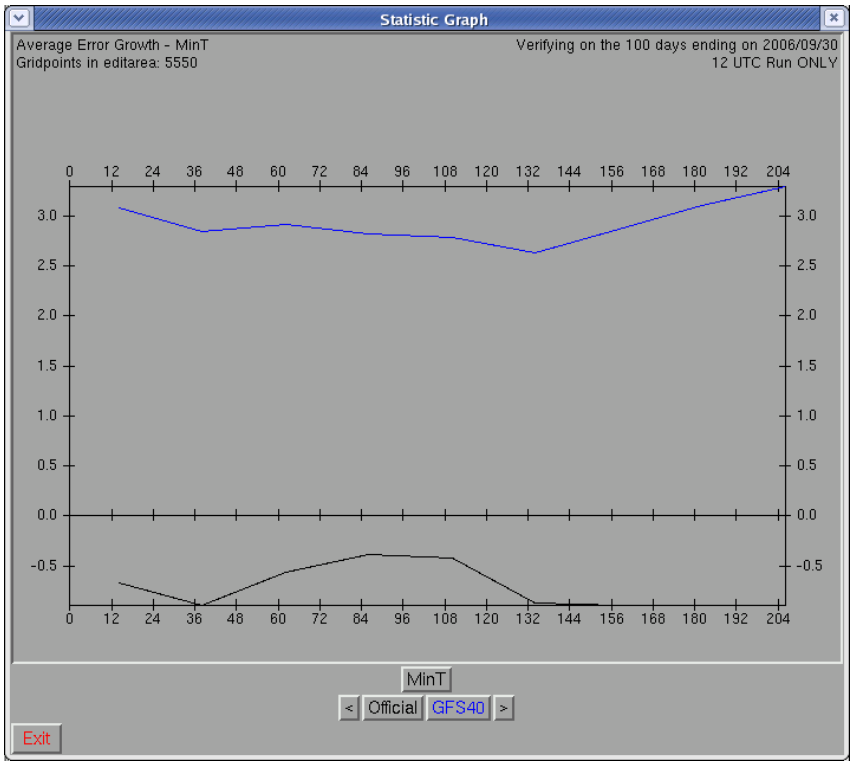
In this case, the toggle buttons allow us to see the same time series at other forecast hours, to see how the errors grow at longer forecast hours. Here, we compare the time series of MinT at 14hrs with 110hrs:





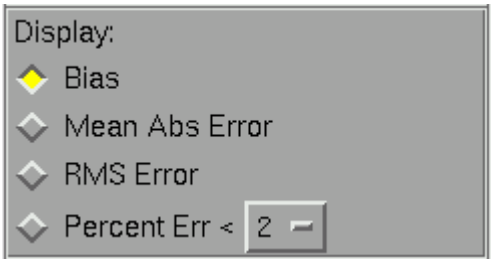
We see that the errors are growing, but the overall bias does not appear to be changing much for the Official forecasts. On the other hand, for the GFS40, there appears to be a positive bias at 14hrs that appears similar at 110hrs.

Choosing to “Plot vs. Fcst Hour” allows you to quantify the trends that we think we see above. Here is the corresponding graph:

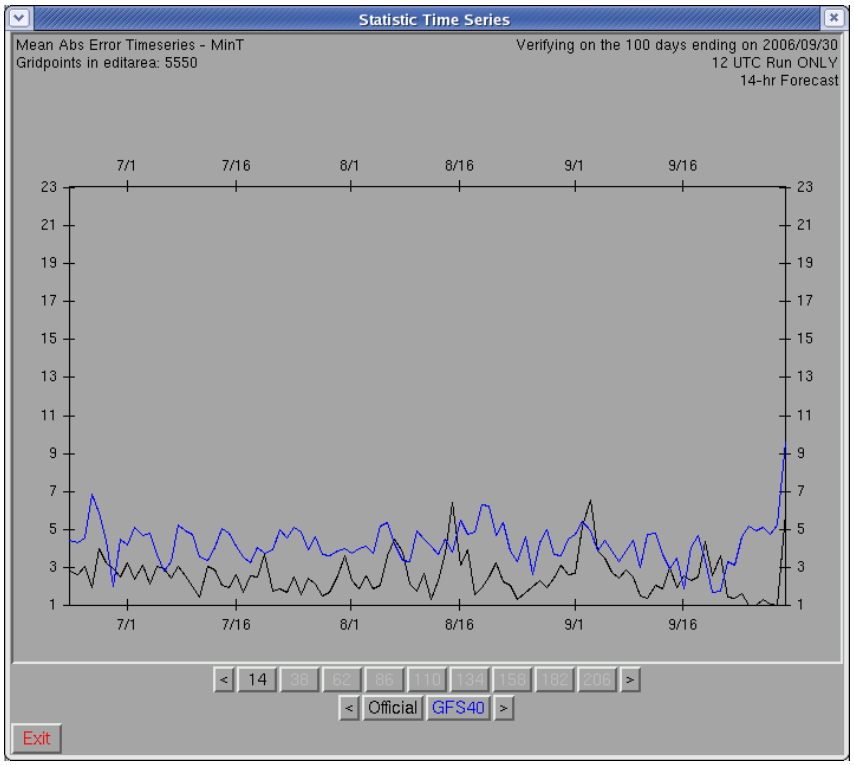


Note that the Official forecasts of MinT over this 100 day period tended to be slightly on the cool side, with an overall bias of between -0.25 and -0.75 degrees. On the other hand, the GFS40 had a bias of around +2.50 to +3.25 degrees.

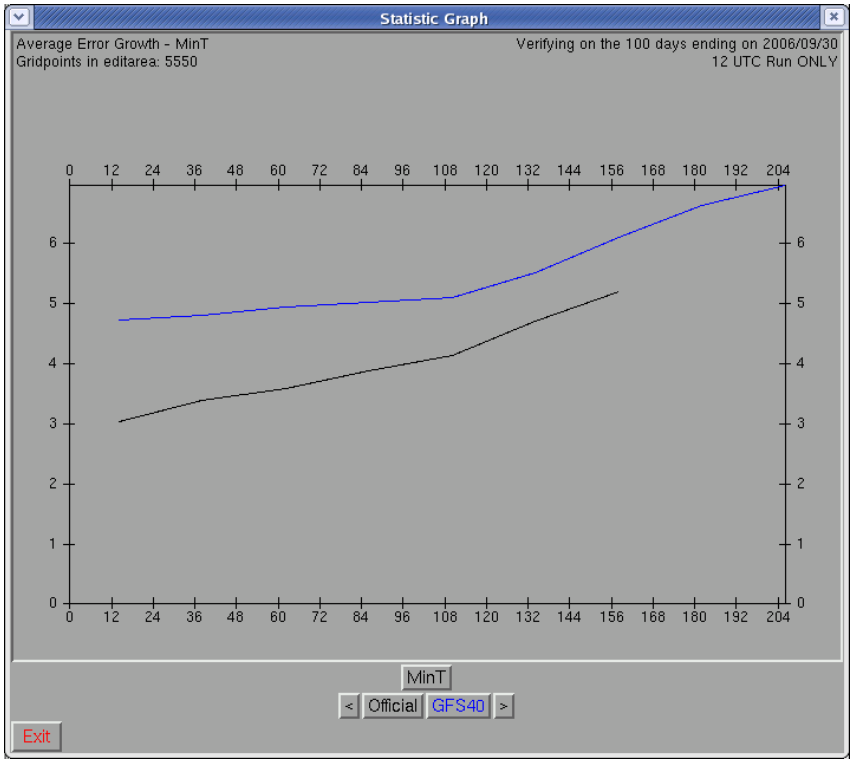
Display – This is where you specify the statistic to display. The same statistics described in the Grid Stats section are also available here.



Above we compared the time series, and time/forecast hour plot of Bias for MinT forecasts from 12 UTC from both the Official and GFS40 forecasts. Here, we show the time series for Mean Absolute Error (MAE) for the same period (at 14hrs):



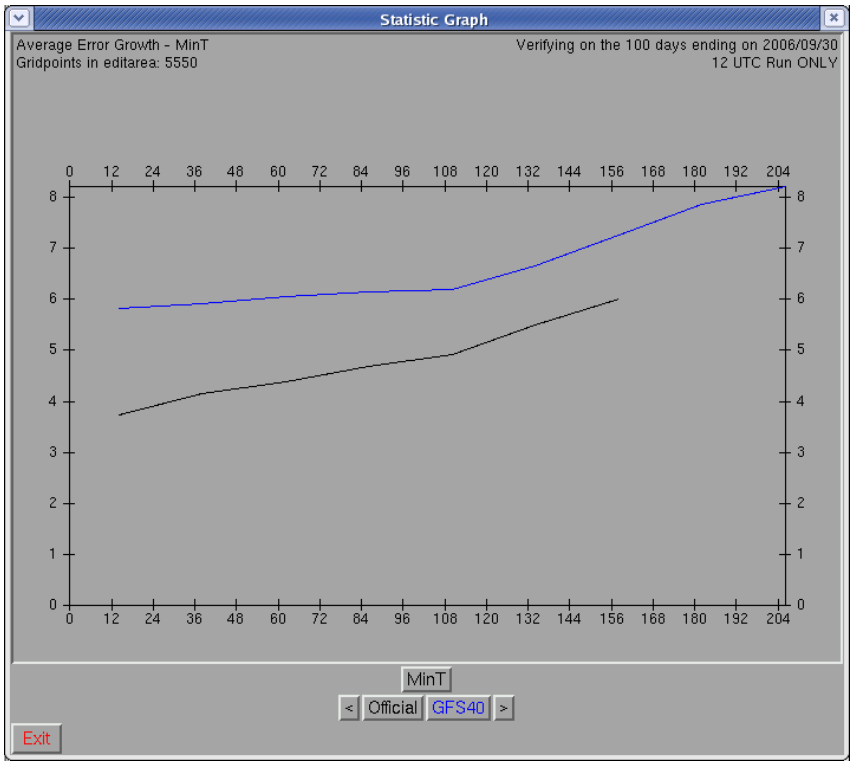
Thankfully, on most days, the Official forecast has smaller overall MAE values at 14hrs than the GFS40. This means that the forecasters are adding useful information to the model forecasts. Remember that this is over the entire CWA edit area, with 5550 points (as shown in the upper-left corner). If we choose “Plot vs Fcst Hour” we get a display like:



This shows that, on average, over these 100 days, the forecasters consistently made smaller

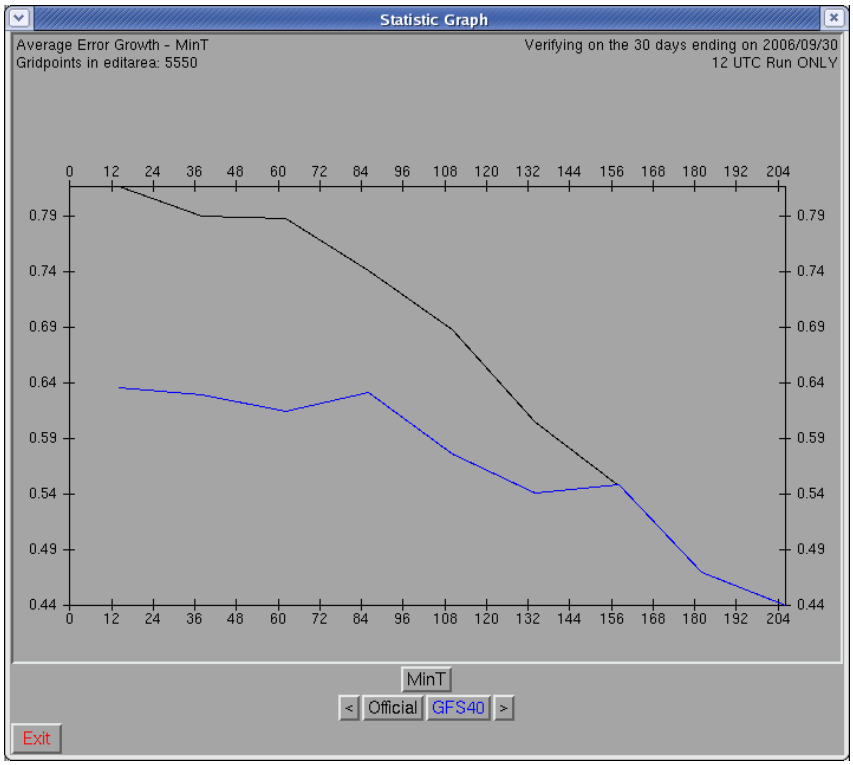
errors than the GFS40 at all forecast times (although the difference is getting smaller at later periods than at early periods).

Here is the RMS statistic for the same edit area over the same period:

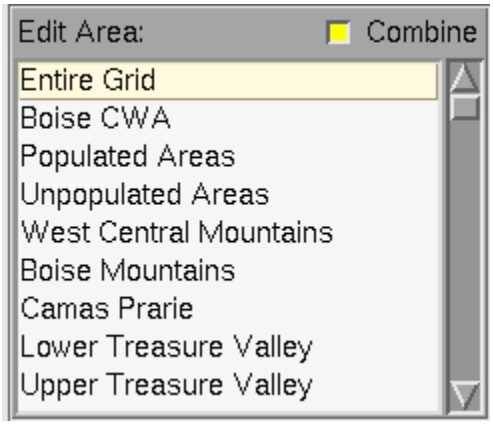


Again, it shows that the forecasters have smaller RMS errors at all forecast times.

You can make similar graphs for the various thresholds. Here, we show the percentage of errors smaller than 5 degrees over the month of September (data for the entire 100 days was not available for this graphic):

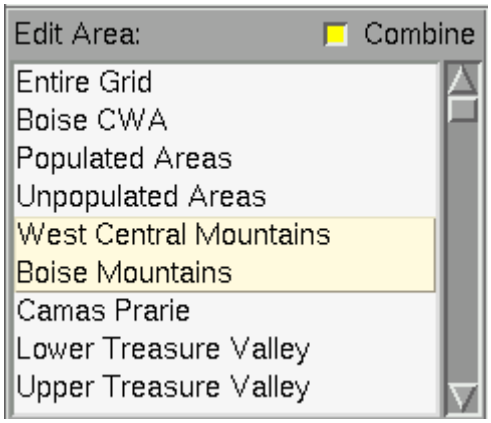


Here it is even more dramatic that the Official forecasts have many fewer gridpoints with error 5 degrees or larger, especially at the short forecast times.

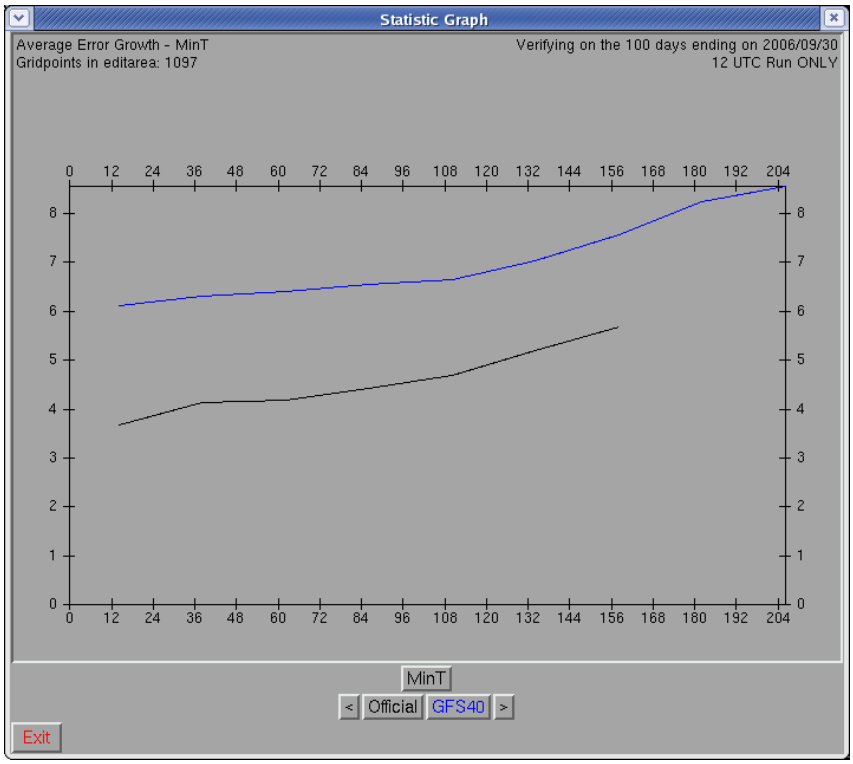


Edit Areas – This is where you choose the edit area (or areas) for which you would like to display statistics. Up to 150 edit areas can be configured locally to have statistics pre-calculated. The scrolled list allows you to select multiple Edit Areas from this list of 150 areas, using the shift and control keys while clicking to add or subtract entries from the selection.

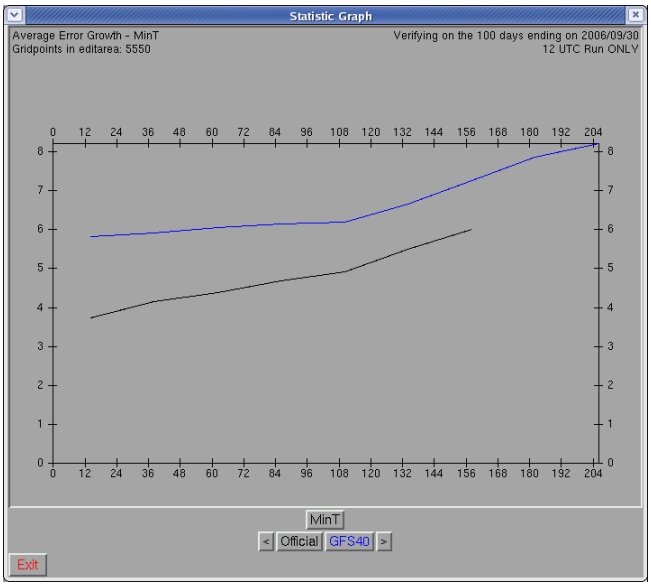
In the examples above, we have been showing time series and statistics versus forecast hour for a single edit area: the Boise CWA. However, sometimes you want to show statistics for a "combined" edit area, made up of several of the pre-defined edit areas. In this case, you would have the "Combine" set on. For example, here we have selected two edit areas that make up most of the mountainous areas within our CWA:

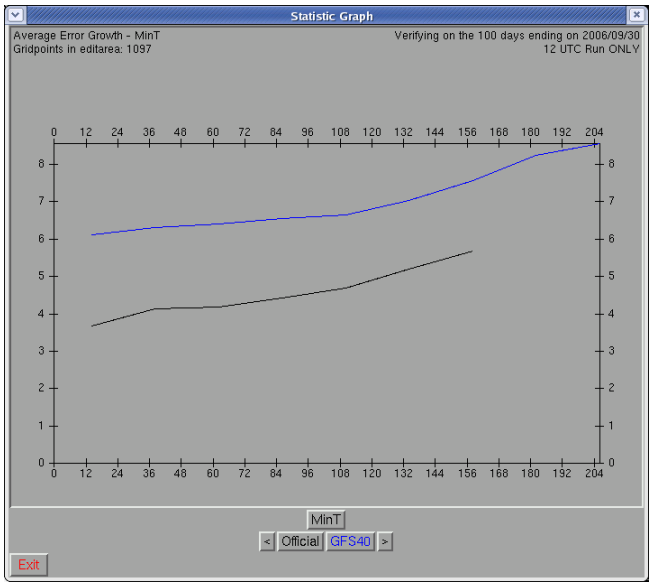


Here we show the RMS errors for these two combined mountain zones for the same 100 days ending on Sept. 30th:

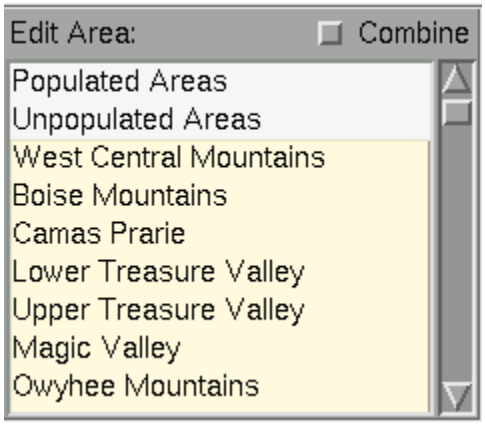


If you compare this to the overall CWA graph of RMS errors shown above, you'll see that GFS40 model has slightly larger RMS errors in the mountains than for the CWA overall, but the Official forecasts are very similar.

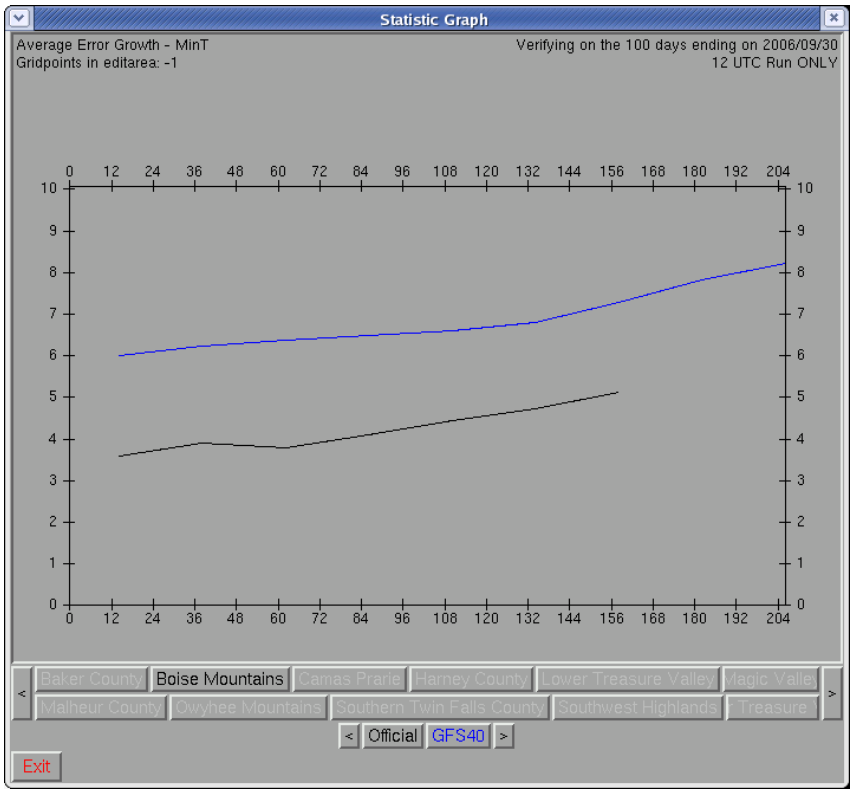




With the "Combine" toggle turned off, you can overlay separate graphs for each edit area selected. For example, here we have chosen all of the edit areas for each of our forecast zones, and have "Combine" turned off:



The display of RMS errors now comes out like this:



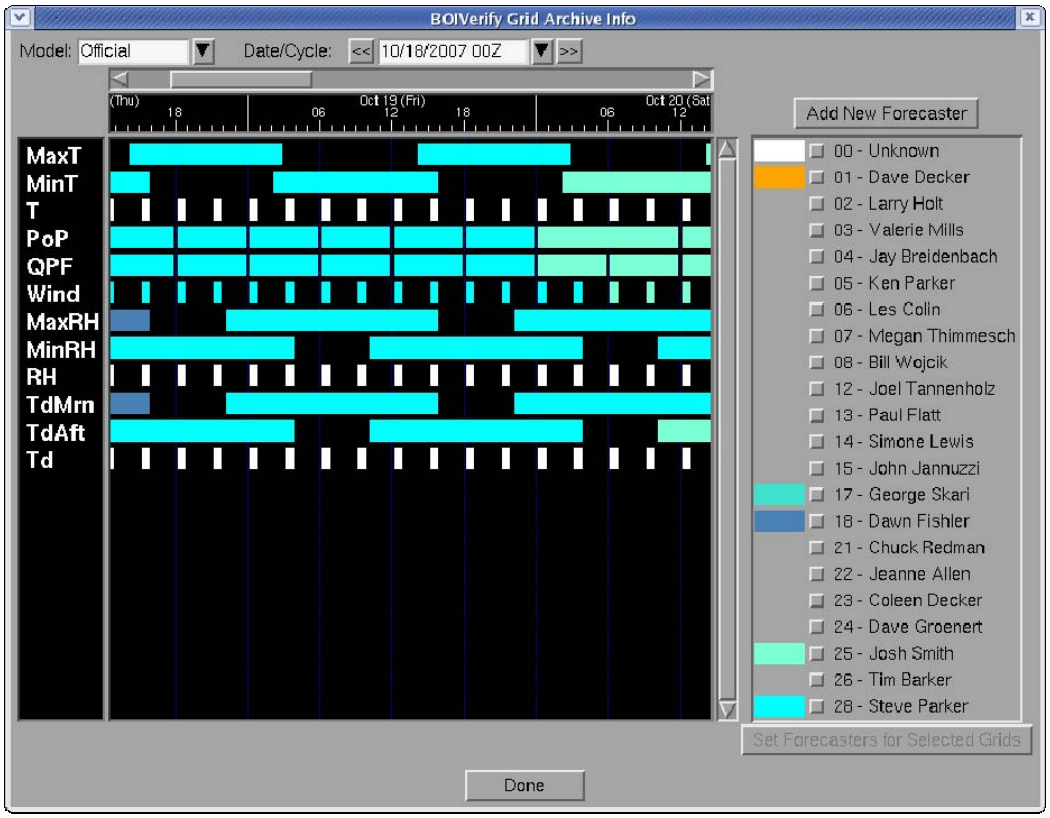
Now, the toggle buttons can be used to switch between the various edit areas, finding areas which have the best and worst RMS error scores.

If you have been paying careful attention, you have realized that with the “Point / Area Stats” tab, you can have either Model, Parameter, or EditArea varying. However, we only ever have two rows of buttons for different graphs, so you can only choose for two of these to vary at the same time. For example, you can show graphs for different models and parameters, but for a single edit area, or you can show graphs for different parameters and edit areas, but for a single model. The BOIVerify software will not allow you to have all three of these vary at the same time, and will display an error message if you try to do so.

BOIVerifyInfo

BOIVerifyInfo is a separate procedure with a GUI somewhat similar to GFE's grid manager. It allows you to see which grids are in the BOIVerify system, change forecaster names or numbers associated with Official grids, delete grids from the system, etc.

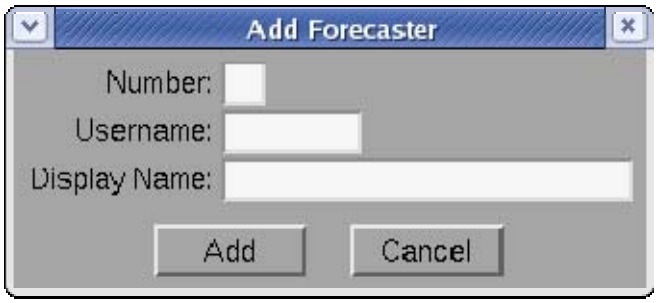
When you start up BOIVerifyInfo, it reads the Official grids from the latest 'run' and displays them in a display similar to the GFE grid manager:



Grid timeblocks are colored based on the forecasters associated with them in the BOIVerify database. The legend on the right hand side shows the colors associated with each forecaster involved in this particular forecast run.

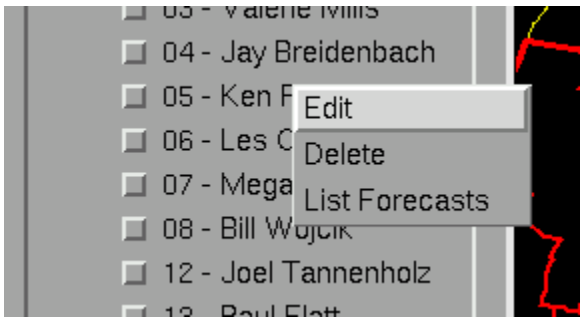
To add new forecasters into the BOIVerify system, click on the "Add New Forecaster" button above the legend on the right hand side:

This will open a new window where you can enter in a forecaster number, username, and full display name:



The number and username must be unique. In other words, if you already have a forecaster using number 15, you cannot add another number 15, or if you already have a forecaster with a username of tbarker, you cannot add another tbarker. The Display Name can be anything you wish.

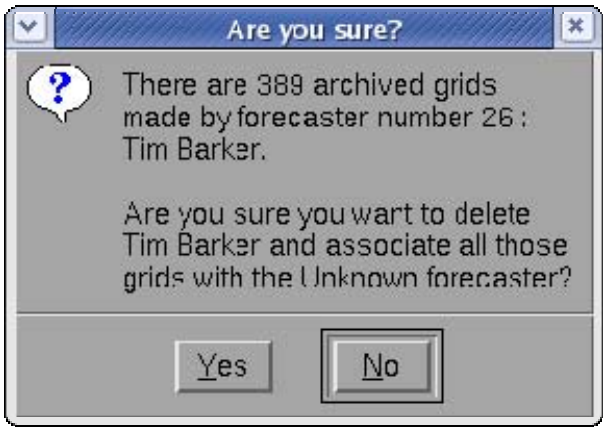
If you right-click over existing forecaster names in the legend, you have the option to 'Edit', 'Delete', or 'List Forecasts' for that forecaster:



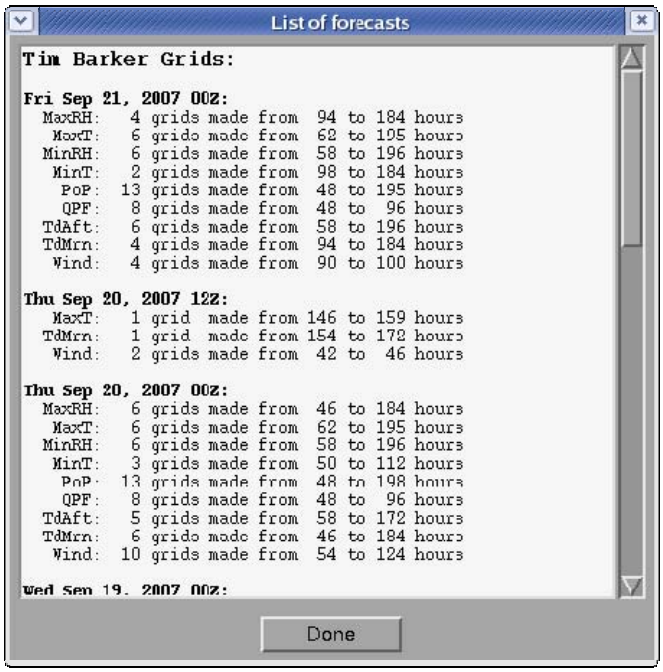
If you choose "Edit", you can edit the values of forecaster Number, Username, and Display Name:

You cannot change the Number or Username to one that is already in use. If you change the forecaster Number, it will search through all the archived Official grids, and change those grids associated with the old number to the new number.

If you choose "Delete", you can delete the forecaster from the BOIVerify system. Any grids associated with that forecaster will be changed to be associated with the unknown forecaster, and it will ask you to confirm that you want to do this:

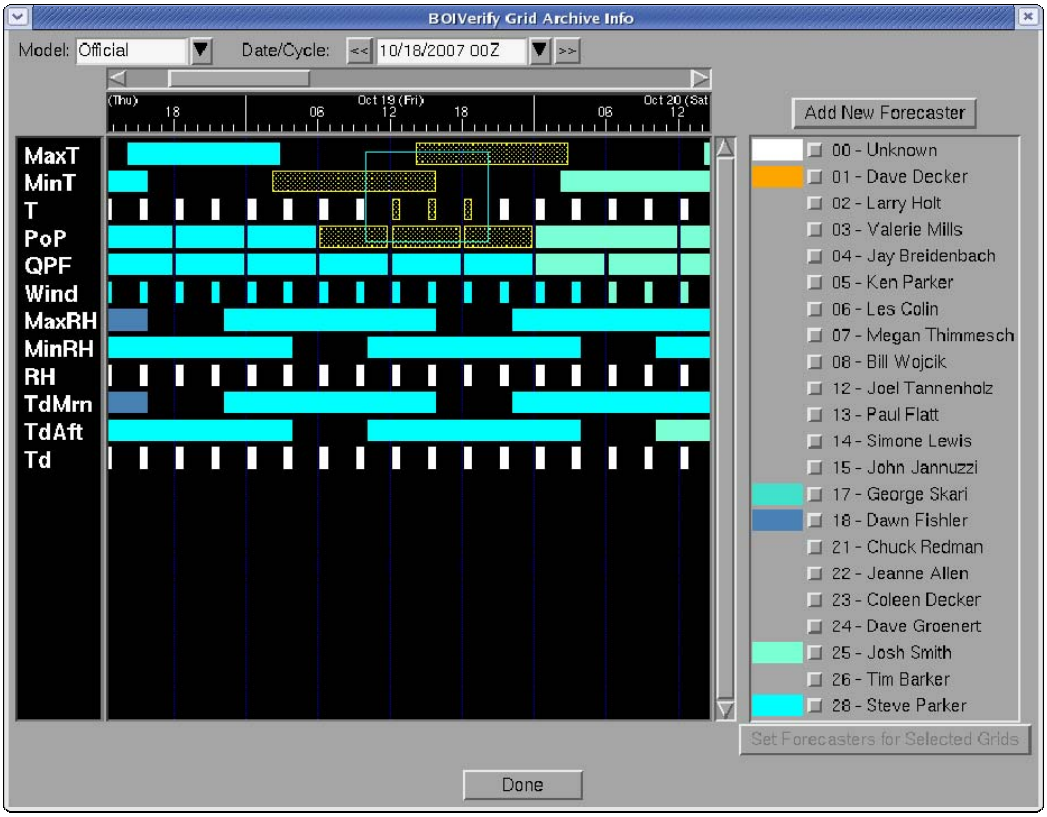


If you choose List Forecasts, a window will be created where all grids associated with that forecaster will be listed (in reverse order):

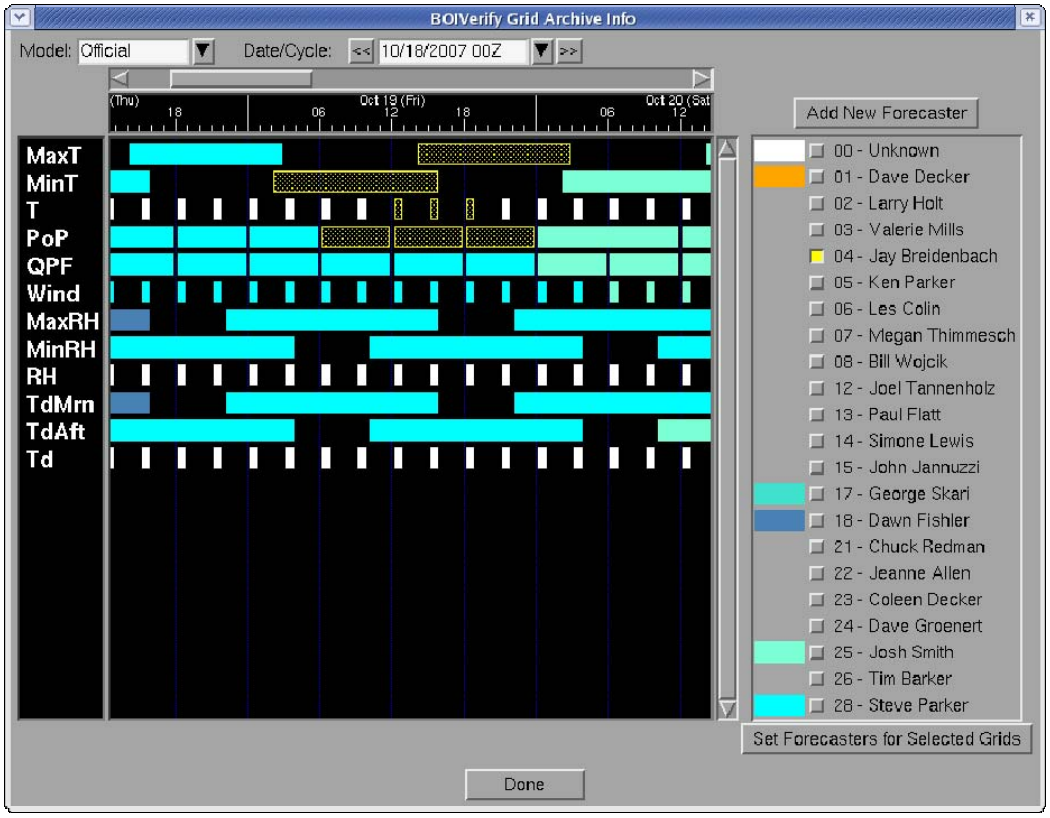


This can be useful for finding grids in the database associated with a particular forecaster, or seeing how many forecasts have been made recently, etc.

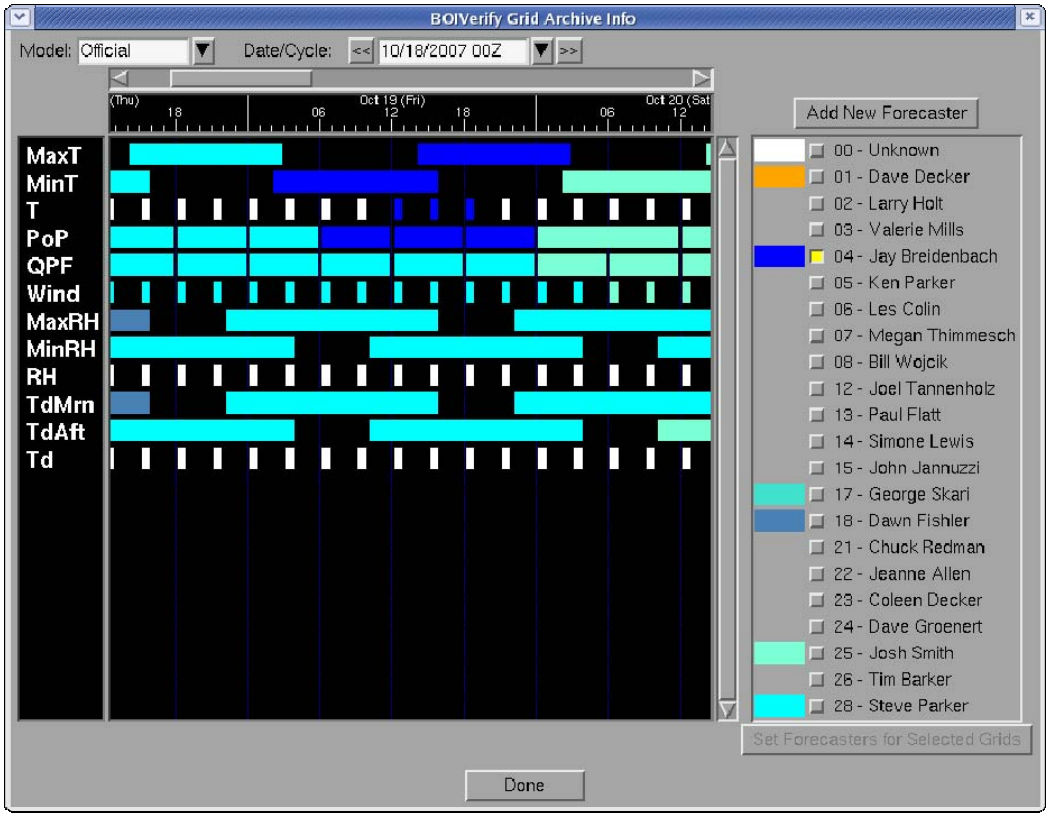
The main use of the BOIVerifyInfo GUI is to CHANGE the forecaster that is associated with a particular grid. Just like the GFE grid manager, if you click or drag with the left mouse button over grid timeblocks, those grids will be ‘selected’. In BOIVerifyInfo, selected grids are shown with a yellow ‘hatching’ pattern:



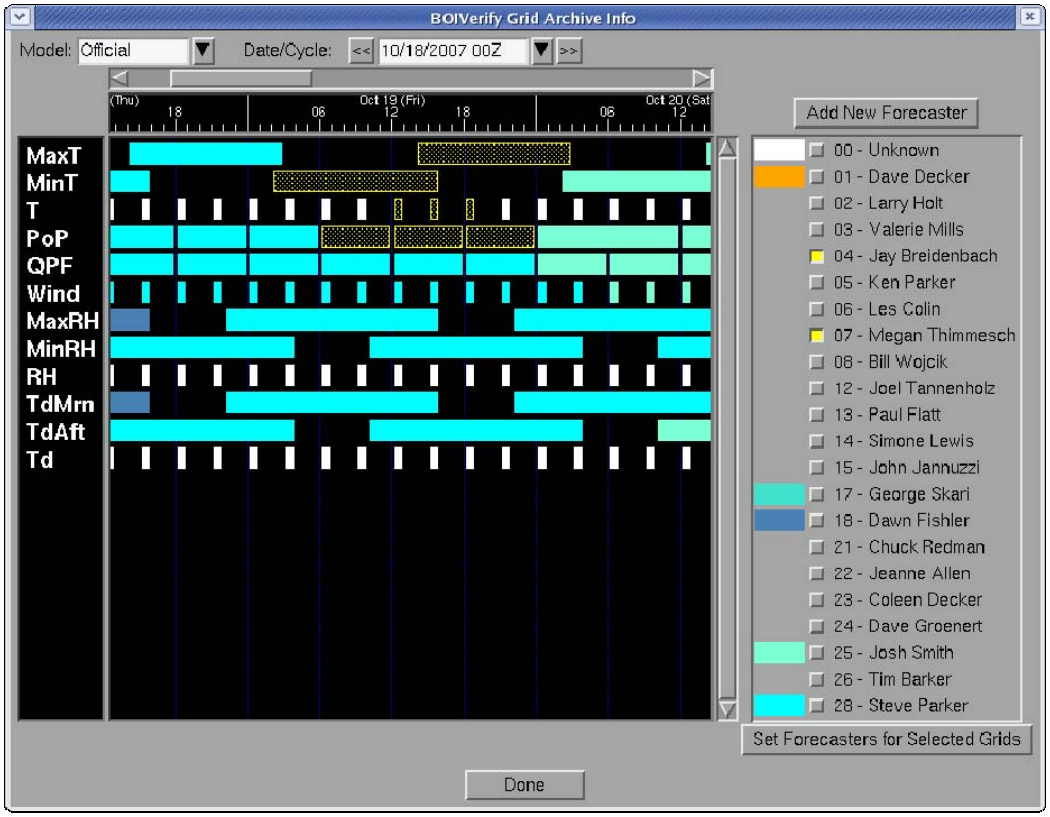
Now, to set which forecasters should be associated with these selected grids, click on the checkbutton next to the forecaster name and number. For example, here we have clicked on the button next to forecaster number 4, Jay Breidenbach:



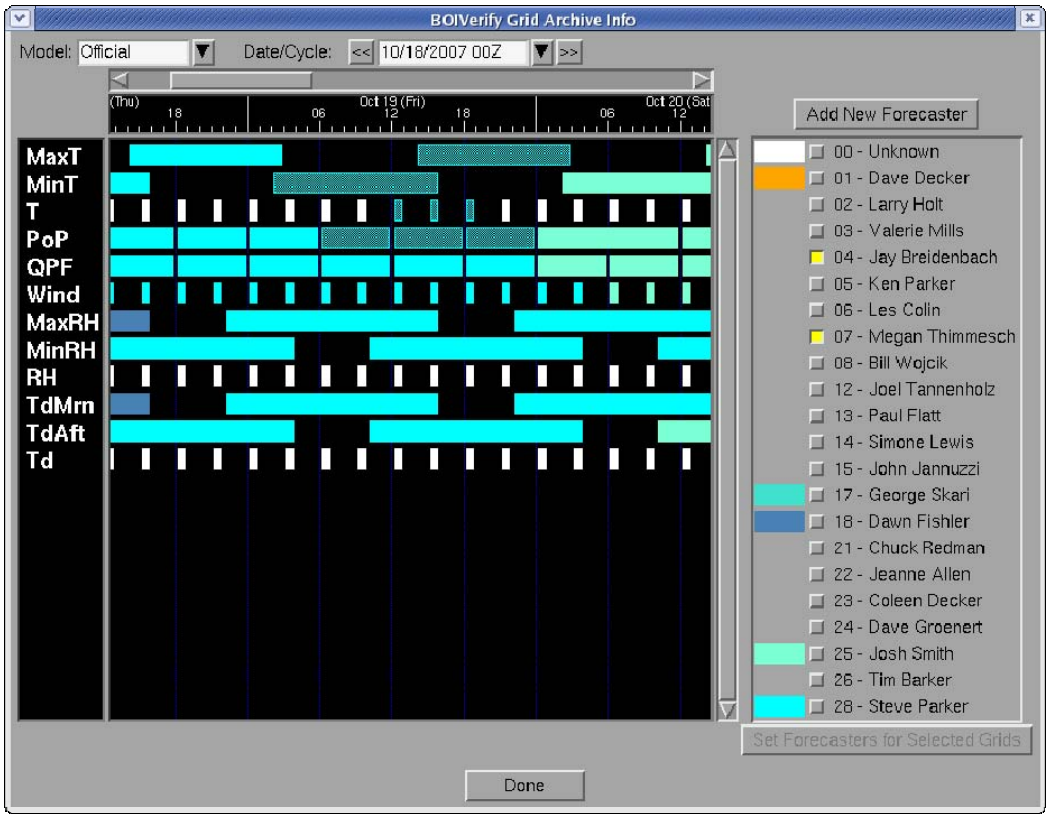
Then, click on the “Set Forecasters for Selected Grids” button at the bottom of the legend on the right-hand side. In this case, we associated forecaster #4, Jay Breidenbach, with these grids, and the display changes, since the grid timeblocks are now put in a different color associated with Jay:



You can associate more than one forecaster with any particular grid (up to 5). This might be helpful in promoting a “team concept” where several forecasters work on different parts of the same grid. To make this association, simply select several forecasters in the forecaster list. For example, here we have selected both forecaster #4, Jay Breidenbach, and forecaster #7, Megan Thimmesch:



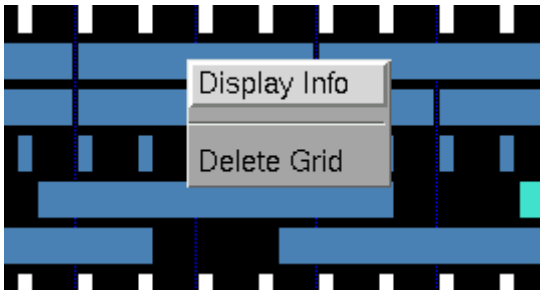
Grid timeblocks that are associated with several forecasters are shown in a different color, and with a 'hatching' pattern:



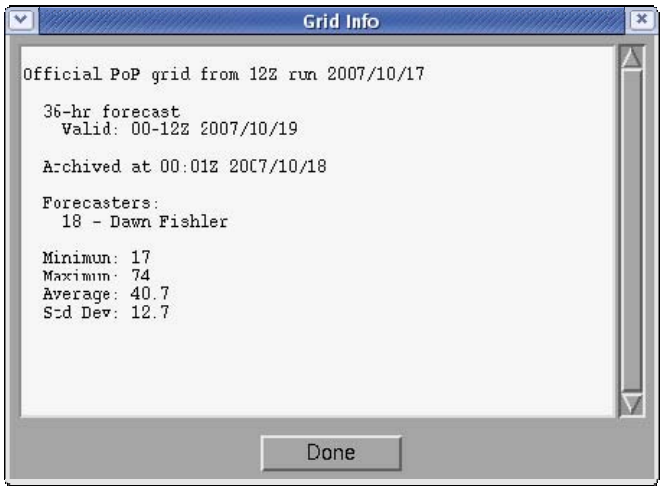
If you middle-click over a grid timeblock, the forecasters associated with that grid will be

selected in the forecaster list on the right. Essentially, the middle-click 'picks up' the forecasters associated with a particular grid. This can be used as a quick way to take a forecaster associated with one grid and then associate them with other grids.

If you hold down the right mouse button while over a grid timeblock, a menu is displayed where you can Display Info the grid, or Delete the grid:

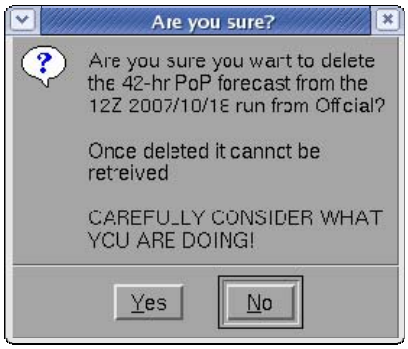


If you choose the Display Info option, a window opens with some information about the grid:



It lists when the forecast was made, when it was archived, the forecasters associated with the grid, and the minimum, maximum, average, and standard deviation of points within the grid. Such information can be helpful when looking for corrupted grids.

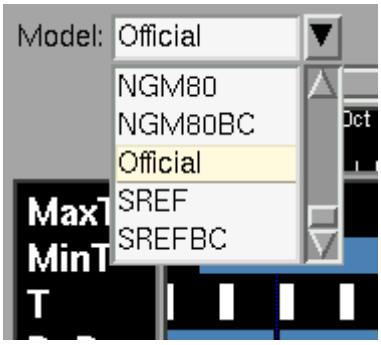
If you choose to Delete the grid, a confirmation message is presented:



Deleted grids cannot be retrieved, so you should be very careful about deleting grids. Nevertheless, there are time when corrupted grid get into the BOIVerify system, and this

provides a way to remove them, so that they do not corrupt statistics.

Normally, the BOIVerifyInfo GUI is used to associate forecaster numbers with Official grids, but you can also display grids associated with any model, by choosing from the Model selected at the upper-left:



Model grid timeblocks are always shown in white, and you cannot associate a forecaster number with a particular grid. Using BOIVerifyInfo to view the grids stored in the database for various models can help when troubleshooting problems with the BOIVerifySave procedure that is saving model data, etc.

Bias Corrected Grids:

The most common use of BOIVerify is to calculate recent model biases, and then subtract them from the current model forecast to create a 'bias corrected' model forecast. You could do this by using the Grid Stats tab to calculate a grid with the model bias over the past, say, 7 days, then use a tool to subtract this grid from the current model forecast grid. This is an excellent forecast strategy, but I wanted something that was more 'automatic' so that forecasters can have the 'bias corrected' grids at their fingertips at all times – without having to run several tools 'on the fly'. So the challenge was to create an automated bias correction technique that was as accurate as possible.

If we were to use the simplest bias correction technique discussed above, the first question is "how many days should we use to calculate the bias?". There are going to be model forecast errors every day, some positive and some negative – and some of which we cannot anticipate. What we really want to know are what are the ***consistent*** errors or biases. To get the best idea of this, we would want to calculate the bias over a long period – say, 30 days or longer. We save grids for up to 50 days, so the 30 day bias can be easily calculated from BOIVerify, and we could remove that bias from the models.

However, the 30 day bias changes very slowly, and we would prefer that the model bias that we remove be more 'responsive' to pattern changes. For example, if we have been under a big ridge for a week or two, the 30 day bias reflects that pattern well. However, if the model then starts forecasting a large trough to move in, the 30 day bias is still strongly influenced by the recent ridge. It seems unlikely that the model errors for the current trough will be similar to what happened during the recent ridge. So, we could try to remove biases calculated over shorter periods, say 10 days, or 7 days, or 5 days. The biases would be more 'responsive' to pattern changes, but they might be ***too responsive*** and bounce back and forth among the random errors, and not really reflect the *consistent* errors.

It strikes me that forecasters believe that the model tends to have similar errors for similar patterns. Thus, I first considered a method of calculating the 'bias grid' via some method that looked for similar patterns in the recent past. In this way, when the model forecast a big change, the bias grid would reflect the errors that the model had for similar conditions in the past, and would NOT reflect the errors of the most recent model runs. Still, there are questions about HOW MANY days to include, and how to determine which forecasts look similar to the current forecast.

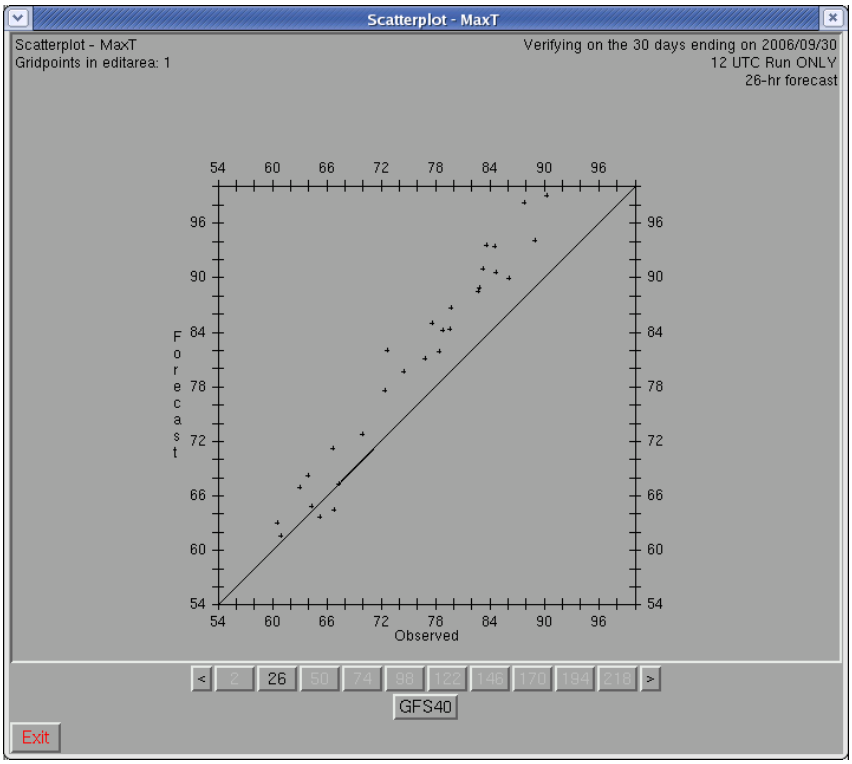
Originally I settled on a method of a ***weighted average*** of recent forecast errors, where the weights were calculated by the pattern correlation of the old forecast grid to the current forecast grid. The pattern correlation was calculated with the recent climatology removed – so that consistent patterns dictated by topography and climatology were not counted when determining similarity. Thus, the similarity was determined via grids that had a pattern of, say, warmer than normal in the southwest and cooler than normal in the northeast, rather than grids that had a pattern of cooler in the mountains, or cooler around the lake – which are basically always there. This meant that the calculated 'bias grid' could change drastically from day-to-day, as the model forecast new patterns – yet it was consistent with the errors that occurred on recent days in the past.

This method worked very well, but when I looked closely, I became a bit more dissatisfied with the weighting calculations. I would look visually at the forecast anomalies, and the anomalies of the old forecasts, and was not happy with the grids that it would choose to weight strongly and those it would choose to weight very little. The problem is that all I could use was a single pattern correlation number for the entire grid (or, the pattern correlation over a part of the grid, say, the CWA). What I found is that the grids would look similar in some areas, but not similar in others. For example, the

current forecast might have a nice warm pattern in the southeast part of the grid – that was very similar to a pattern from last week, but in the northwest part of the grid, the forecast was very different from what happened on that day last week. The pattern correlation was low, and that grid was weighted very little in the average, even though in ***parts*** of the grid, I thought it should be weighted very strongly.
I considered a method where I would weight different parts of the grid differently, based on ***local*** pattern correlations, but then I would have had to figure out the optimal size for the 'local' determination, and that just didn't seem like it would be easy to determine. Instead I wondered what if I could determine how similar *each gridpoint* in the current forecast was to *each gridpoint* of the recent forecasts, and weight each gridpoint appropriately. What I found after looking at this for a while, is that this basically comes down to making a linear regression of gridpoint parameter value to gridpoint forecast error at each gridpoint.

Thus, I settled on a method where I perform a linear regression of forecast anomaly values to forecast errors on each gridpoint independently – and then apply that to the current model forecast to estimate what the forecast error will be – to remove the expected forecast error at each gridpoint. The regression uses the past 30 days as it's training period. It's possible that more or fewer days might produce better results, but this seemed like a reasonable compromise.

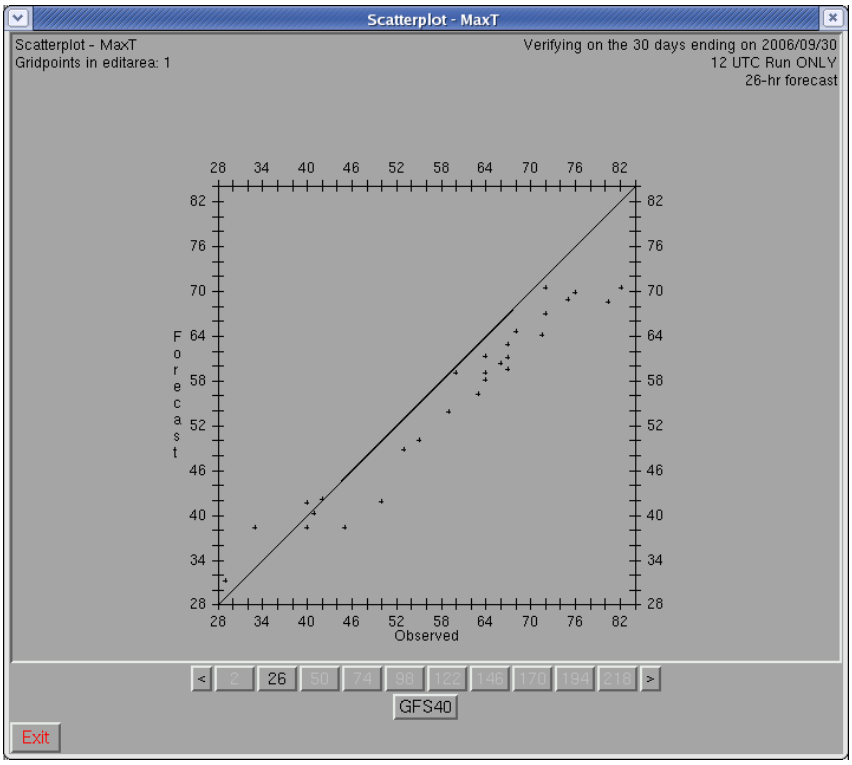
To get a feel for what the technique does, lets consider some examples. Shown here is a scatterplot produced by BOIVerify for a gridpoint in the center of our CWA, for the 12 UTC GFS40 MaxT forecasts at 26-hrs that verified during the month of September:



This point shows some interesting forecast errors. When the model forecast cool temperatures it tended to make fairly accurate forecasts. However, when the model forecast warm temperatures, it tended to be a bit on the warm side. Basically this technique finds the best-fit linear regression line to this scatterplot. This line can then be used to predict what the error will be at this point for another forecast. If the new forecast for this gridpoint is relatively cool – say in the low 60s, then it will expect very little error and will change the model forecast very little. However if the new model forecast for

this gridpoint is 96 degrees – it will cut back that forecast considerably.

Because gridpoints close to each other have similar forecasts and observations, the graph you see above, and the 'bias correction' that it produces is also similar for the gridpoints closeby. Here, on the other hand, is a scatterplot for the same times, but for a gridpoint at a different elevation and few hundred miles away from the first:



Though not quite as consistent as the graph shown above, here the tendency is almost the opposite. When the forecast is cool, it tends to work out pretty well – but when the forecast is warm, it isn't warm enough. Since the linear regression is calculated independently for each gridpoint, it can easily handle the case of the first gridpoint being forecast to be relatively warm, while the second gridpoint is forecast to be relatively cold – and produce very different corrections in each area.

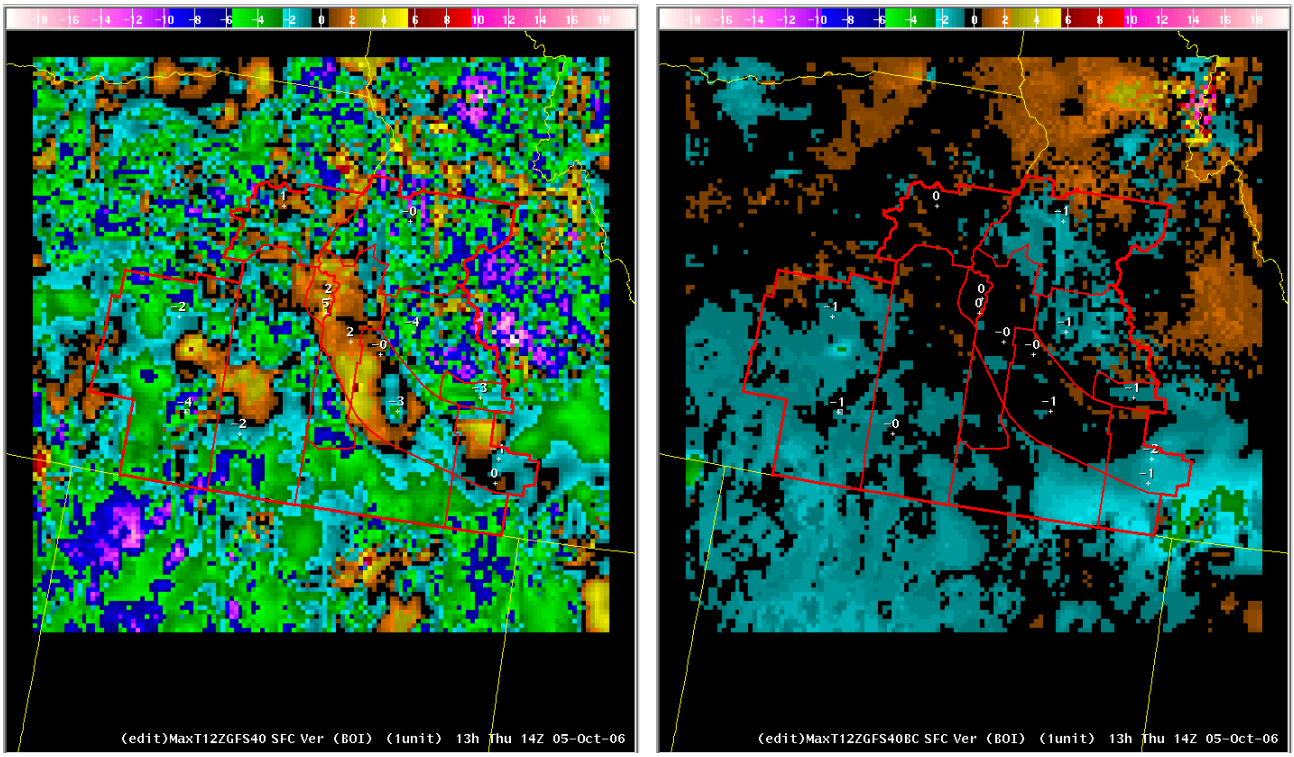
Experience has shown some problems with this procedure when the current forecast lies well outside the range of forecast values experienced during the recent past (30 days in this case). For example, for the gridpoint shown in the example above, we have had GFS40 forecasts in the past 30 days ranging from 31 to 71 degrees. Now, what if the new GFS forecast for this gridpoint is 72 degrees? It seems reasonable to assume that the regression line that fits somewhere near the old forecasts of 71 degrees, will yield reasonable results for a new forecast of 72 degrees. Essentially, we are 'extrapolating' the regression line beyond what was in the training period. But for a just a few degrees, this seems reasonable.

However, what if the new GFS forecast is 95 degrees? Or 106 degrees? It becomes increasingly unreasonable to expect that 'extrapolating' will be appropriate. Thus, in BOIVerify 2.0, we added code that trends the forecast error back toward the 'simple bias' when large extrapolations occur. The 'trending' starts at 10% of the range of forecast values, and is complete when 20% of the range is reached. In the example above, the range of old forecasts was from 31 to 71 degrees, or a range of 40 degrees. Thus, for forecasts up to 4 degrees outside the range (or 75 degrees) we use the regression line completely. Beyond 75 degrees, we start heading back toward only using the average

error, and reach that by 79 degrees. For a case where the GFS40 makes a new forecast of 95 degrees, the adjustment will truly only remove the average bias in all the old forecasts.

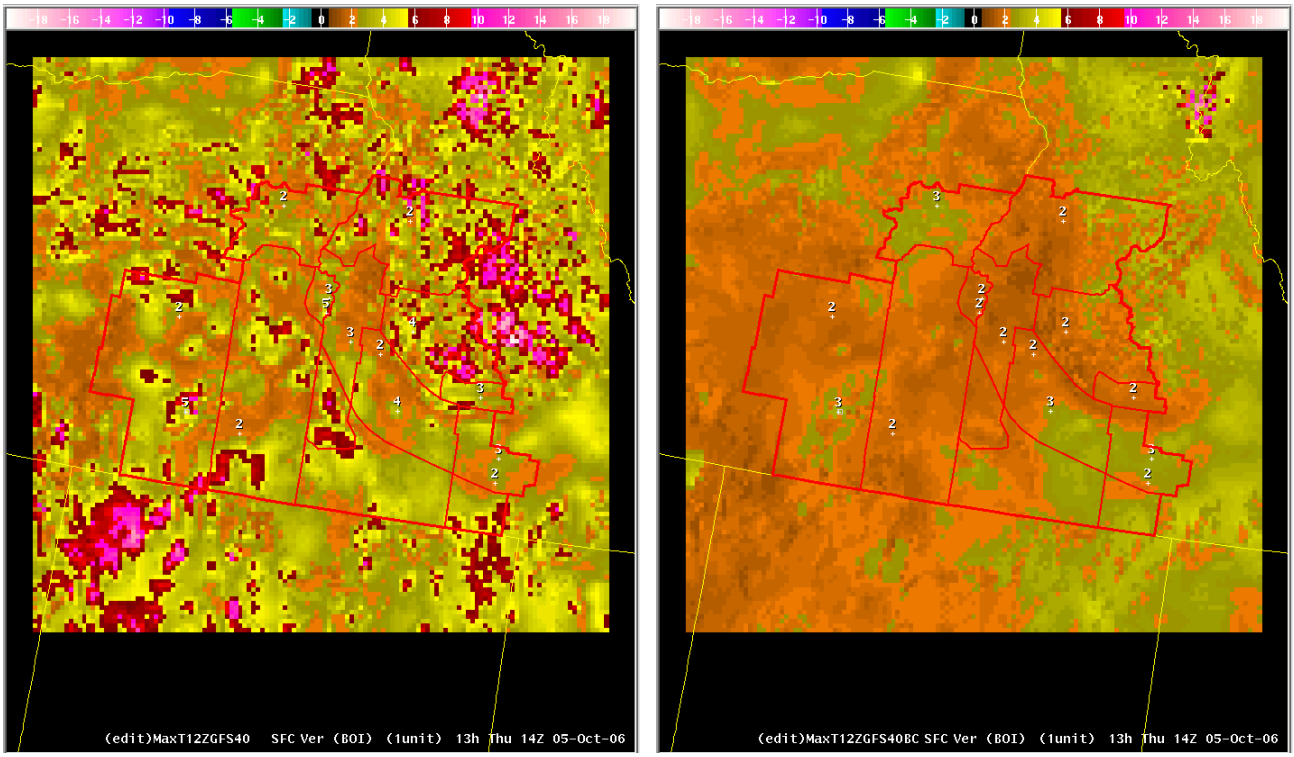
This ‘bias correction’ procedure happens automatically ‘in the background’ as new model data arrives. Scripts see when new model grids become available, perform the linear regression, and store new model grids that are ‘bias corrected’ into GFE. These “bias corrected” grids are stored as different GFE models or databases – with the letters “BC” added to the name. For example, in addition to GFS40 grids, we also have GFS40BC grids available. Forecasters can copy from, or populate from, these “bias corrected” models.

These bias corrected models are just like any other GFE model, and we can save those grids into the BOIVerify system and verify them as well. The results have been very encouraging. For example here are the average bias of the 26hr MaxT forecasts from the 12 UTC GFS40 model compared to the GFS40BC model for the month of September:



There are relatively large temperature biases in the GFS40 grids (most likely due to problems with the smartInit methods), which have been drastically reduced in the GFS40BC grids.

Although the bias has been reduced, model grids still have more ‘random’ errors from day-to-day. Here the MAE grids for the same period are compared:



Again the benefits are obvious. The GFS40BC grids still have errors, but they are more similar across the entire area. The GFS40 grids have small pockets of large MAE errors, mainly in areas where large biases are found (probably due to problems with the smartInit).

These bias corrected grids provide a very good “starting point” for future forecasts. Forecasters can now concentrate on making changes to the forecast that are meteorologically oriented, rather than just trying to account for problems in the smartInit. For example, forecasters that think the incoming trough will move in slower than the model, can make reasonable edits to warm up the model forecast (perhaps more warming in areas more affected by winds than others), rather than having to spend lots of time also trying to remove the typical model forecast errors.

Of course, the techique is not perfect. Since it uses a linear regression on only 30 data points, a single bad observation can really skew the results. If a bad observation of -40 degrees shows up in the observed grids for one day when the forecast was 50, the regression will then try to make future forecasts of 50 degrees come out with a forecast of -40 instead! This is not what you want! And those bad observation grids can have an influence for 30 days! It pays to spend time making sure that you have quality observation grids – so that the bias corrected grids will be as accurate as possible.