OHD-Core 18.1.1  Release Notes

Part of CHPS Build: 19.1.1  Release Date: December, 2019

Overview
This release contains 14 Bug fixes & Enhancements which are new since the release of OHD-Core-17.1.1

Known Issues and Limitations
None at this time.

Scripts / Documentation

Scripts
LRO.py

Documentation
The following pieces of documentation have been modified since the last release and can be found in the directory at the root of the package. All the CHPS documentation may be found online at https://vlab.ncep.noaa.gov/group/chps/wiki/-/wiki/OWP+Documentation/FrontPage

● Modified:
  Calibration Configuration Guide
  LongRangeOutlook Operations Guide
  MEFP Configuration Guide Data ingest

● New: None

● Removed: None
### Fixes

<table>
<thead>
<tr>
<th>Redmine ID</th>
<th>Reported By</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>27113</td>
<td>ABRFC</td>
<td>RESSNGL inputs need to be extended in WaterCoach</td>
</tr>
<tr>
<td>53541</td>
<td>CNRFC</td>
<td>Graphgen Aggregator end date issue.</td>
</tr>
<tr>
<td>54565</td>
<td>OHRFC</td>
<td>Lag/K mod error when default lag is zero</td>
</tr>
<tr>
<td>58561</td>
<td>CNRFC</td>
<td>MEFP Extreme Members (possibly due to random sampling technique?)</td>
</tr>
<tr>
<td>60725</td>
<td>HQ</td>
<td>Replace deprecated <code>string.atof</code> method in LRO.py</td>
</tr>
<tr>
<td>68636</td>
<td>MARFC</td>
<td>AEI via AEICQN modifier artificially bounded by -5 and 5</td>
</tr>
</tbody>
</table>

### Enhancements

<table>
<thead>
<tr>
<th>Redmine ID</th>
<th>Requested By</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>53457</td>
<td>CBRFC</td>
<td>Add CSV support to EVS.</td>
</tr>
<tr>
<td>54083</td>
<td>HQ</td>
<td>Testing the new GraphGen template for the HEFS 10-day product.</td>
</tr>
<tr>
<td>59891</td>
<td>HQ</td>
<td>Update GEFS gather script to reflect change from NCEP.</td>
</tr>
<tr>
<td>59355</td>
<td>WGRFC</td>
<td>ModuleConfigFiles Template WATBAL_SACSMA_Calibration provided in 18.1.1 is incorrect.</td>
</tr>
<tr>
<td>61011</td>
<td>HQ</td>
<td>Remove ensmodels.climatologyFewsAdapter from CHPS Baseline.</td>
</tr>
<tr>
<td>61000</td>
<td>OHRFC</td>
<td>Reduce precision of exceedance probabilities in LRO.py</td>
</tr>
<tr>
<td>61022</td>
<td>NWRFC</td>
<td>Make scp destination an argument passed by module in LRO.py</td>
</tr>
<tr>
<td>68552</td>
<td>OHRFC</td>
<td>UNITHG doesn't show expect warnings in log file or window.</td>
</tr>
</tbody>
</table>

---

**Detailed Description of Software Changes and Enhancements**

**Fixes:**

Release Date: 12/6/2019

Version: OHD-CORE- 18.1.1

2 of 12
**Redmine 27113 – RESSNGL inputs need to be extended in WaterCoach**

**Description**
We (ABRFC) are having issues running WaterCoach for RESSNGL modules. Normally the PELV and RQOT inputs stop at T0. RESSGNL runs fine in normal ops with the configuration. However, in WaterCoach mode RESSNGL fails as it says it needs PELV data out to the end of the run. I talked with Alaina at MARFC and they have the same issue but were able to make a workaround by adding modules to extend the inputs. However, we have many more reservoirs would hope to not have to make changes to every single one to have WaterCoach work. I’m not sure if it’s an issue with FEWS or if it can be fixed in the RESSNGL itself.

**Cause**
N/A

**Fix**
OHD common code was updated to allow for truncated input time series. It will check for the condition of allowing the Model input time series to be extended beyond T0 if that datatype is allowed missing values based on the nwsrfs_datatype_mapping_file.txt file.

**Notes**
N/A

---

**Redmine 53541 – GraphGen Agreggator End Date issue**

**Description**
CNRFC use Graphgen to compute quantiles for peak flows and date estimates during snowmelt season (Apr-Jul). They've noticed an issue with Graphgen's "wild card" year logic. When the start date of the aggregation (T0) has the same month and day as the end date (0001-08-01 in this case), the end date year remains the same as the T0 year. This causes the Graphen product to fail. Currently, the end date year is selected as the first year beginning AFTER OR ON the start date year. We’d like to see this logic changed so that the end date year is selected as the first year AFTER the start date year.

Here are some examples of how we'd like to see the end date year selection work:

For T0=12/1/2017, end date (0001-08-01) would be 8/1/2018
   for T0=4/1/2018, end date (0001-08-01) would be 8/1/2018
   for T0=8/1/2018, end date (0001-08-01) would be 8/1/2019

**Cause**
What CNRFC is pointing out is a bug, or perhaps undesirable feature, in how calendar dates specified in GraphGen templates with the year starting with 0 are processed. This is an existing capability in GraphGen and CNRFC is merely
recommending a change in how it is computed. Here are two good examples of the capability for a current system time is 8/7/2018 12Z:

8/1/0001 12:00:00 would yield 8/1/2019 12:00:00, because 0001 indicates the first date/time matching the month, day, hour, minutes, seconds AFTER OR ON the current system time. Notice the "OR ON" portion of this, which is the problem (see below).

9/1/0001 12:00:00 would yield 9/1/2018 12:00:00 for the same reason; i.e., its the first Sept 1 AFTER OR ON the current system time.

Now here is an undesirable result:

8/7/0001 12:00:00 would yield 8/7/2018 12:00:00, because that is the first date/time AFTER OR ON the current system time; in this case, "ON". Hence, if that date is the end time of an aggregation with the start time being T0, then the total length of aggregation is 0 and GraphGen errors out. This is not what CNRFC wants. They would prefer to remove the "OR ON" part and have all dates be determined based on only AFTER. Thus, 8/7/0001 12:00:00 should yield 8/7/2019 12:00:00.

Fix

A code change in the GraphGen was required to solve this problem, after correcting the way date are handled the error was solved.

Notes

N/A

<table>
<thead>
<tr>
<th>Redmine</th>
<th>54565 – Lag/K mod error when default lag is zero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>If the default lag is zero and you try to make a lag mod you get an error. It does not matter if you use the OHD core 17.1.1 or the old 16.2.1, you get the same error. The odd thing about this is that if the default K is zero, the mod will work. This does suggest that it is possibly a module parameter issue of some sort and a configuration change could possibly fix it. If that is true, it would be nice to know what that configuration change is so we can make the change.</td>
</tr>
<tr>
<td>Cause</td>
<td>The current location has default PAIR_QT_LAG_CARRYOVER set to 0 in the statesI.txt file, so the user cannot make a Lag mod.</td>
</tr>
<tr>
<td>Fix</td>
<td>The Lag/K source codes have been updated to handle this problem without changing the statesI.txt file.</td>
</tr>
</tbody>
</table>
### Redmine
58561 – MEFP Extreme Members (possibly due to random sampling technique?)

### Description
The option for additional INFO-level logging requested to aid in the RFCs diagnosing extreme precipitation amounts output by the MEFP. The original ticket description is as follows:

“We (CNRFC) have seen very large outlier members being generated by MEFP from time to time that don’t seem very realistic. They can happen within the first 24-72 hours where we don’t expect such behavior. We documented a case for the San Diego River at Fashion Valley recently. In this case, we noticed a very large outlier member that was due to an extreme precip value for period 5 (24-30 hour lead time). Our MEFP configuration uses the RFC forecast as the seed forecast for the first 3 days. In this case, the RFC forecast was zero for all future periods. The MEFP canonical events in play for this case are 1) the base event for period five, and 2) the modulation event that covers days 0 to 3. We suspected that this member might be due to the random sampling technique where a value at the extreme tail of the distribution was selected. We re-ran the mefp pre-processing again, and the extreme member disappeared, and all members looked more realistic. We have a hard time defending these extreme members to our partners; especially when they can completely disappear from a re-run using the same inputs. We’ve documented some of this in the google sheets link below. Can this be proven that it is due to random sampling? Or is it possibly something else? If it is due to the stratified random sampling technique, is there a way to eliminate such behavior in HEFSv2 without other adverse impacts (underdispersed forecasts)?”

### Cause
N/A.

### Fix
Implemented the option to include additional logging information related precipitation amounts generated by MEFP for each canonical event and member through the EPT algorithm. The message includes the following (see the MEFP User’s Manual for information on the EPT algorithm):

- the canonical event involved (multiply the start and end by 6-hours to identify the time period);
- the standard normal deviate sampled;
- the value of the precipitation forecast in normal space;
- the probability resulting from pushing both numbers through the EPT linear algorithm and then taking the inverse standard normal QPF;
- the quantile corresponding to that probability based on the distribution of observations; and
the parametric distribution of those observations.

The log output will only be generated if the probability (fourth bullet above) exceeds a threshold specified when the logging is turned on. An example log output line:

```xml
<line level="3" description="EPT Algorithm for event CanonicalEvent(number = 34,
start = 21, end = 24, numsens = 5): After application of EPT linear function to standard normal deviate 1.2658583109229777 with QPF in normal space of 0.8774589927066887, the probability used is 0.9228119594449637 resulting in quantile 34.43493995199982 mm sampled from distribution GammaDist: parameters = [19.61381, 0.58197, 0.249]."/>
```

To turn on the logging, in the MEFPEsembleGeneratorModelAdapter module configuration file, add the following run file property, which includes the aforementioned threshold as the “value”:

- **key** = eptThresholdToIncludeSamplingLogMessages
- **value** = the threshold that, when a probability equals or exceeds it, a log message is output. Must be between 0.0 (results in one message per event and ensemble member) and 1.0 (results in no messages).

For example, the file
`Config/ModuleConfigFiles/hefs/CachePutah/CachePutah_MEFP_FMAP_Forecast.xml`
could be edited to include this XML element within runFileActivities in order to see all generated precipitation amounts:

```xml
<double key="eptThresholdToIncludeSamplingLogMessages" value="0.0"/>
```

This would result in a flood of information messages. To restrict it only to precipitation amounts wherein the corresponding probability is in the upper tail, for example greater than 0.9, the following could be used (you may choose to use a value even greater than 0.9, such as 0.95 or 0.99, to capture more extreme events):

```xml
<double key="eptThresholdToIncludeSamplingLogMessages" value="0.9"/>
```

**Notes**

N/A

**Redmine**

60725 – Replace deprecated `string.atof` method in LRO.py
| Description | O&M noticed the use of string.atof() in DetermineExceedanceValue(). That conversion function was deprecated back in Python 2.0 and will be removed in Python 3.0. They recommend the following replacement now while we've noticed the issue, so it doesn't sneak up on us later.  
Replace `string.atof(theValue)` with `float(theValue)` |
| Cause | N/A |
| Fix | The LRO.py script has been updated as proposed. |
| Notes | N/A |

| Redmine | 68636 – AEI via AEICQN modifier artificially bounded by -5 and 5 |
| Description | The value entered for the AEICQN mod is being applied to the AIADJ mod too. Any value greater than 5.0 gives a warning in the log viewer and only sends 5.0 value to the model - even if the configured maximum is greater than 5. |
| Cause | N/A |
| Fix | Fixed the Fortran code to apply the AEICQN mod properly. |
| Notes | N/A |
### Enhancements:

<table>
<thead>
<tr>
<th>Redmine</th>
<th>Description</th>
<th>Cause</th>
<th>Fix</th>
</tr>
</thead>
</table>
| 53457         | **Add CSV support to EVS.** CHPS has a csv ensemble export available. We have switched to this format as it takes up to a third less disk space. Also, this format is faster to parse for scripting applications. It would be nice for EVS to support this format for re-forecasting verification. | N/A                        | **Added the ability to:**  
1. The ability to read the prescribed CSV format, uncompressed.  
2. The ability to read the prescribed CSV format, gzip compressed. |

<table>
<thead>
<tr>
<th>Redmine</th>
<th><strong>Testing the new GraphGen template for the HEFS 10-day product</strong></th>
<th>OHD common code was updated to allow for truncated input time series. It will check for the condition of allowing the Model input time series to be extended beyond T0 if that datatype is allowed missing values based on the nwsrfs_datatype_mapping_file.txt file.</th>
<th>Updated GraphGen files are provided with the release. Files are also attached to the Redmine ticket.</th>
</tr>
</thead>
<tbody>
<tr>
<td>54083</td>
<td>We will provide a new GraphGen template for the HEFS 10-day product. We would like it tested. We will document how to use it in the CHPS/HEFS workflows and would like the template and documentation added into the CHPS baseline.</td>
<td>OHD common code was updated to allow for truncated input time series. It will check for the condition of allowing the Model input time series to be extended beyond T0 if that datatype is allowed missing values based on the nwsrfs_datatype_mapping_file.txt file.</td>
<td>Updated GraphGen files are provided with the release. Files are also attached to the Redmine ticket.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Redmine</th>
<th><strong>Update GEFS gather script to reflect change from NCEP</strong></th>
<th>OHD common code was updated to allow for truncated input time series. It will check for the condition of allowing the Model input time series to be extended beyond T0 if that datatype is allowed missing values based on the nwsrfs_datatype_mapping_file.txt file.</th>
<th><strong>Update GEFS gather script to reflect change from NCEP</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>59891</td>
<td>NCEP has migrated their NOMADS traffic from HTTP to HTTPS server. The GEFS gather script (gefs_data.sh) used for HEFS will need to be updated to reflect this change.</td>
<td>OHD common code was updated to allow for truncated input time series. It will check for the condition of allowing the Model input time series to be extended beyond T0 if that datatype is allowed missing values based on the nwsrfs_datatype_mapping_file.txt file.</td>
<td><strong>Update GEFS gather script to reflect change from NCEP</strong></td>
</tr>
</tbody>
</table>

**Release Date:** 12/6/2019

**Version:** OHD-CORE- 18.1.1

8 of 12
<table>
<thead>
<tr>
<th>Redmine</th>
<th>59355 – ModuleConfigFiles Template WATBAL_SACSMA_Calibration provided in 18.1.1 is incorrect.</th>
</tr>
</thead>
</table>
| Description | Andrew (at WGRFC) found a number of errors in the WATBAL_SACSMA_Calibration.xml template file in CHPS-18.1.1 OHD-Core-17.1.1
1. Tries to obtain RO components from SACSMA module at timeStep "00Z" instead of at timestep unit="hour" multiplier="6"
2. LZFSC is not used, only LZFSF is used, and LZFSF erroneously placed where LZFSC should be expected (right after LZFPC).
3. The WATBAL template should allow for an instantaneous timestep other than 6 hour by using a variable multiplier instead of hardcoded at 6 (which is how it was done for the PEAKFLOW_Selection_Calibration_HOURS.xml)
I've attached that version as well as "proposed" and am including a sample "proposed" Stats_Calibration workflow that can use it. |
| Cause | Fix According to Andrew’s suggestion, some template configuration XML files have been updated. |
| Fix | 1. ModuleConfigFiles/calibration/templates/WATBAL_SACSMA_Calibration.xml
   - Rename all variableId *_00Z and *_6h to *_hourly
   - Change RO components timeStep `<timeStep id="00Z"/>` to `<timeStep unit="hour" multiplier="$TS$"/>`
   - Using a variable multiplier instead of hardcoded at 6 `<timeStep unit="hour" multiplier="$TS$"/>`
   - LZFSC is not used, only LZFSF is used, and LZFSF erroneously placed where LZFSC should be expected (right after LZFPC).
2. SystemConfigFiles/DisplayGroups.xml and DisplayGroups.xmlWithoutSNOW17
   - Replace parameterId **LZFSC** with **LZFSF** in the plot ids `<plot id="WATBAL_SACSMA_CONTENT_Multi-year">` and |
3. RegionConfigFiles/Filters.xml
   - Replace parameterId LZFSF with LZFSC in the filterID
     <filter id="Monthly_End_Of_Month_Soil_Moisture_Storage_Contents_MY"
             name="Monthly End Of Month Soil Moisture Storage Contents">
     <filter id="Annual_End_Of_Year_Soil_Moisture_Storage_Contents_MY"
             name="Annual End Of Year Soil Moisture Storage Contents">
     <filter id="Monthly_End_Of_Month_Soil_Moisture_Storage_Contents_WY"
             name="Monthly End Of Month Soil Moisture Storage Contents">
     <filter id="Annual_End_Of_Year_Soil_Moisture_Storage_Contents_WY"
             name="Annual End Of Year Soil Moisture Storage Contents">

4. WorkflowFiles/calibration/XXXXX_Stats_Calibration.xml
   - Add property TS key (string key="TS" value="6") to
     WATBAL_SACSMA_Calibration <moduleConfigFileName> tag.

<table>
<thead>
<tr>
<th>Redmine</th>
<th>61011 – Remove ensmodels.climatology.ClimatologyFewsAdapter from CHPS Baseline.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>OWP thinks that this application is now obsolete and would like to hear from the RFCs if they still have a need for it. We polled the 13 RFCs and it was reported back that they no longer have a need for this adapter.</td>
</tr>
<tr>
<td>Cause</td>
<td></td>
</tr>
<tr>
<td>Fix</td>
<td>We updated the build.xml to remove the classes needed to create the climatology adapter and the test cases. Climatology adapter will not be included in the distribution jar file. Source code was not changed only it will not be compiled.</td>
</tr>
<tr>
<td>Notes</td>
<td>In an effort to &quot;clean up&quot; and streamline our development environment, we have removed this application from the OHD-Core baseline package starting with OHD-Core-18.1.1 in the CHPS-19.1.1 Release since it is no longer used. The less code we need to support, maintain, and test, the better.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Redmine</th>
<th>61000 – Reduce precision of exceedance probabilities in LRO.py</th>
</tr>
</thead>
</table>

**Release Date:** 12/6/2019  
**Version:** OHD-CORE- 18.1.1  
10 of 12
### Description
Reduce the significant digits of the probabilities shown via the tool tip hover on the website. Showing exceedances to the hundredth of a percentage point (ie. 52.39%) is implying precision way beyond our accuracy. Whole numbers (52%) would be much more appropriate.

### Cause
N/A

### Fix
The LRO.py script has been updated as proposed.

### Notes
N/A

---

### Redmine
61022 – Make scp destination an argument passed by module in LRO.py

### Description
The LRO.py script has a hard-coded destination for sending LRO output: ldad@ls1:/data/ldad/public/images/lro. This is problematic for testing on SA or devTest, as output from a test gets sent to the operational destination. We would like the destination changed to an argument that may be passed by the general adapter that calls the LRO script. This way, the destination may be set by a global property.

### Cause
N/A

### Fix
The LRO.py script has been updated as proposed.

### Notes
The configuration needs to be updated as below.

1. Add a destination variable to global_properties file
   
   `LRO_SCP_DEST=/<scp_destination>`

   **Note:** If `LRO_SCP_DEST` variable is blank (`LRO_SCP_DEST=`), then the default destination is used "ldad@ls1:/data/ldad/public/images/lro"

2. Add the third “<argument>$LRO_SCP_DEST$</argument>” tag to after the “<argument>$LRO_DIR$</argument>” tag in the LRO module General Adapter file `Config/ModuleConfigFiles/graphgen/GraphGen_AHPS_Products_LRO.xml`
<table>
<thead>
<tr>
<th>Redmine</th>
<th>68522 – UNITHG doesn't show expect warnings in log file or window.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The issue is we are having a difficult time trying to get any UNIT-HG adapter warning or error messages to show up in the IFD log, or any other log for that matter.</td>
</tr>
<tr>
<td>Cause</td>
<td>Not all the WARNING and ERROR messages were being triggered and displayed correctly.</td>
</tr>
<tr>
<td></td>
<td>ERROR messages are logged when the PrintDebugInfo value is 1 or greater.</td>
</tr>
<tr>
<td></td>
<td>WARNING messages are logged when the PrintDebugInfo value is 2 or greater.</td>
</tr>
<tr>
<td>Fix</td>
<td>Reviewed, corrected (if needed), and verified that all the WARNING and ERROR level messages listed in the Unit Hydrograph documentation are triggered and displayed correctly for the corresponding PrintDebugInfo value.</td>
</tr>
<tr>
<td>Notes</td>
<td>N/A</td>
</tr>
</tbody>
</table>