BioSum 5.11.1 Release Notes  
November 7, 2024

**Current Release Notes (5.11.1)**

**Please read these notes in their entirety before converting an existing BioSum project as there are many changes in database architecture and some specific actions required to ensure a smooth transition. BioSum v5.11.1 is a significant departure from earlier versions of BioSum in that it is the first version to replace MS Access databases with SQLite databases in many parts of the application. An existing BioSum project must be at v5.10.1 before it can be upgraded to v5.11.1. BioSum will (ideally) automatically migrate the affected databases to SQLite when a v5.10.1 project is opened. If the conversion is not successful, the details provided below may be of assistance in completing a successful upgrade of your project. Because of the scope of the database changes, once a project is migrated to v5.11.1, it must not be opened again in v5.10.1 as doing so may corrupt the project irretrievably. Making a backup copy of a project before upgrading is always recommended.**

1. **biosum\_ref.accdb**: This database lives in the analyst’s AppData directory and contains system-maintained reference tables for BioSum. We are updating the fiadb\_fvs\_variant table with this release. BioSum will automatically start using the new table, but any pre-existing customizations to this table will be lost. The first time v5.11.1 is started, it will download the new database. This database is quite large so be patient if BioSum takes longer than usual to start up the first time.
2. **FVS Output Data**: Sequence number definitions have been migrated to SQLite. There is a new database in the project db folder named fvsmaster.db that stores the sequence number definitions. As part of the upgrade process, any sequence number settings will be copied from the old tables in fvsmaster.accdb to this new database. These old tables will remain untouched in fvsmaster.accdb if needed for reconciliation.  
     
   The FVS Output audits and append process have been migrated to SQLite. There is no longer a need to create Access databases from the FVSOut.db and this button has been removed. Note that the PREPOST\_FVS\_XXX.ACCDB databases in the project data folder will NOT be migrated to SQLite as part of the upgrade process. Analysts will need to re-run the FVS audits and append process following the upgrade of legacy projects to v5.11.1. Audit output has been consolidated in the data/FVS\_AUDITS.db database. The PRE/POST FVS Out tables are now in a single PREPOST\_FVSOUT.db in the data folder (same location as FVS\_AUDITS.db).
3. **OpCost 10.1.6**: OpCost inputs, configurations, and outputs have all been migrated to SQLite. Analysts will need to reset the ‘Directory Path of the OPCOST .R file name’ in the ‘Settings’ menu to "C:\Program Files\FIA PNW Portland Forestry Sciences Lab\FIA Biosum 5.11.0\opcost\Opcost\_10\_1\_6.R".  
     
   A prerequisite of OpCost 10.1.6 is the RSQLite package. This package needs to be installed if it is not present in the R installation. The RODBC package is no longer required or used as of OpCost 10.1.6.  
     
   OpCost relies on configuration tables stored in the opcost\_ref.db which should be located in the db folder of every project. This database should be automatically copied into the correct location for legacy projects when the conversion script runs.
4. **Processor**: Processor configuration tables and system outputs will be stored in SQLite databases starting with this release. For older projects configurations will be migrated from scenario\_processor\_rule\_definitions.mdb to scenario\_processor\_rule\_definitions.db as part of the upgrade process, but the older .mdb file will be left in place for reconciliation purposes.  
     
   The scenario\_results.mdb files for existing scenarios will not be converted. To generate Processor output that can be consumed by Treatment Optimizer, processor scenarios will need to be re-run for all variant packages.
5. **GIS data**: The process that loads user-supplied travel times and processing sites into a BioSum project has been converted to use SQLite for data storage. The user-supplied data must be created in an SQLite database. Analysts have the option of downloading the gis\_travel\_times\_master.db from the <http://biosum.info> site and placing it in their ‘AppData’ folder if they don’t want to conduct their own travel time calculations via transportation and mill data they supply.  
     
   As part of the upgrade process, existing projects will have their data migrated from /gis/db/gis\_travel\_times.accdb to its SQLite equivalent. Travel times and Processing Sites do not need to be reloaded.
6. **Treatment Optimizer**: Optimizer configuration tables and system outputs will be stored in SQLite databases effective with this release. For older projects configurations will be migrated from scenario\_optimizer\_rule\_definitions.mdb to scenario\_optimizer\_rule\_definitions.db as part of the upgrade process, but the older .mdb file will be left in place for reconciliation purposes. Settings in the optimizer\_definitions.accdb will also be migrated.  
     
   Because the FVS PRE-POST tables are now organized into one SQLite database, the FVS PRE-POST Context Database option has been removed from the ‘Run Optimizer Scenario’ screen.  
     
   The weighted variables data tables will not be converted for existing projects. Use the “Recalculate All’ button to populate this data before running an Optimizer scenario that uses it. The optimizer\_results.accdb files for existing scenarios will not be converted. Treatment Optimizer scenarios will need to be re-run for all variant packages. Because outputs are now in SQLite, there is no need for the ‘Export to SQLite’ button and it has been removed.

**Quick Reference: What processes need to be rerun for existing projects?**

|  |  |  |
| --- | --- | --- |
| **Process** | **Rerun with v5.11.1?** | **Notes** |
| Loading plot data | No |  |
| Defining prescriptions | No |  |
| Running FVS to generate an FVSOut.db | No |  |
| Audits related to FVS Output data | Recommended | FVS Append Output needs to be re-run |
| FVS Append Output | **Yes** | Required by Processor, Optimizer |
| Run Processor scenario(s) | **Yes** | Required by Optimizer |
| Load GIS Data | No | Existing data will be migrated |
| Optimizer calculated variables | **Yes** | Uses SQLite FVS Output. Use ‘Recalculate All’ button |
| Run Optimizer scenario(s) | **Yes** |  |

1. Many additional minor enhancements, and anomaly resolutions, are described in the BioSum [v5.11.0](https://github.com/USFS-PNW/Fia-Biosum-Manager/releases/tag/5.11.0) and [v5.11.1](https://github.com/USFS-PNW/Fia-Biosum-Manager/releases/tag/5.11.1) online release documentation. Users with projects that predate this release will find these especially worthwhile.  
     
   One that may be particularly germane to users with existing projects is:

“issue [#352](https://github.com/USFS-PNW/Fia-Biosum-Manager/issues/352): OPTIMIZER: Calculation for USEBIOMASS\_YN was incorrect; Incorrect escalators were used in the calculations”.

This issue also included incorrect application of escalators that predates version 5.11.0 - earlier projects that had different escalators for costs and revenues would have net revenue incorrectly calculated and will need to be re-processed and re-optimized; those with identical escalators for costs and revenues would have been calculated correctly.

Another is critical for users of weighted variables to know about. What the release docs reference as “issue [#367](https://github.com/USFS-PNW/Fia-Biosum-Manager/issues/367): OPTIMIZER: Handle negative numbers in prepost tables as null when calculating weighted variables (12-NOV-2024)” addresses (by converting to nulls) negative values (such as the –1 value reported by FVS-FFE's POTFIRE table as a missing value code for canopy base height in cases of no trees present larger than 1-inch diameter) when calculating weighted sums (the BioSum approach to obtaining weighted averages, when weights sum to 1). However, any null attributes values (including the internally converted –1’s) associated with non-zero weights are handled as 0’s in this calculation, which may or may not make sense analytically (if there is no tree canopy to receive fire, and no tree ladders to for fire to climb, does a value of 0 feet make sense, keeping in mind that a real 0 with branches extending down to the ground on every tree would indicate a stand with no resistance to crown fire). An interim work-around for analysts to adjust weighted values when nulls are present is being tested and will be shared later.