Power-user Features of FVS and FVSOnline

Nicholas Crookston[1,*]

[1]U.S. Forest Service (retired), *ncrookston.fs@gmail.com

Overview:

- FVS power-users are those that have may have analysis needs beyond those easily fulfilled using the FVSOnline graphical-user interface.
- Software tools are described here that support building, changing, and running FVSOnline using scripts to automate repetitive tasks and support using FVSOnline as a component in other systems.
- The FVS model can be manipulated using a second set of functions that interact with the model itself, modifying growth and mortality rates, simulating regeneration or harvests, and so on, thereby allowing a user to alter the FVS model without modifying the Fortran and C code it is written in.
- These tools are written in the R programming language (R Core Team 2021). The most reliable source of detailed information on how to use them is provided with the related R packages available from the USDA Forest Service FVS web site https://www.fs.usda.gov/fvs/index.shtml.

Summary:

FVSOnline is an R-shiny (Rstudio 2023) application that can be used to build FVS (USDA Forest Service 2023) simulations, run, and display results. The system has a client-server architecture where the server is a specialized R package that is running in an R session and the client is a graphical user interface running in a web browser. In addition to the components that are accessed via the web browser, there are R functions contained in the FVSOnline package that support manipulation of FVSOnline projects. Projects can contain several FVS runs and each run can contain several simulation units (forest stands) and management alternatives. Table 1 lists the R functions designed for this purpose.

FVS itself is the simulation model that projects forest stand development. There exists a set of internal functions in FVS that can be called by another program and thereby manipulate the projected results, perhaps replacing them completely with predictions made with different models. While those functions can be called by programs other than R, there is an R package called rFVS that is designed to interact with these internal functions (table 2). Note that the Acadian growth model (Weiskittel and others 2017) is implemented within the FVSOnline system using the rFVS functions to interact with FVS. In that case, growth, mortality, and regeneration establishment are modeled using functions in rFVS and the Acadian model. One of the motives of building rFVS is to provide a system where modelers can try new approaches to predicting forest stand dynamics. Those new methods and then be tested and delivered to others within FVS without changing that model's internal code.

Key Concepts for FVS Users:

- FVSOnline can be a component in other systems.
- FVSOnline can handle large tasks, and repetitive tasks can be automated.
- FVS can be modified without first learning the FVS computer program and languages (mostly Fortran) thus empowering innovation.

Management Applications/Future Directions:

- The future direction is to maintain this software and enhance it as needs are discovered.

Table 1—Functions in FVSOnline designed to be used externally from the web browser interface.

Function name	Description
addNewRun2DB	Add the output from a run into the master FVSOut.db
extnAddComponentKwds	Add Keyword Components to a run
extnAddStands	Add Stands to a run
extnDeleteComponents	Delete FVS keyword components from a run
extnDeleteRuns	Delete one or more FVS Runs
extnDeleteStands	Delete specific stands from a run
extnDuplicateRun	Duplicate a run and assign the duplicate a new title and new management ID
extnErrorScan	Scan an FVS output file for errors
extnFromRaw	Convert a raw vector to an R object
extnGetComponentKwds	Get FVS keyword components from a run
extnListRuns	Get a list of FVS Runs
extnListStands	List the stands in a run
extnLoadFVSRun	Fetch a run into the R session
extnMakeKeyfile	Generate the FVS .key file for a run
extnMakeRun	Build an FVS run in a project
extnSetRunOptions	Set FVS output and timing options
extnSimulateRun	Simulate (run) a run's .key and RScript
extnStoreFVSRun	Store a run that is in an R session into the FVS project database
extnToRaw	Convert an R object to a raw vector
fvsOL	Run fvsOL (FVS OnLine/OnLocal)

Table 2—Functions in rFVS.

Function name	Description
fvsAddActivity	Add an FVS Activity to the activity schedule
fvsAddTrees	Add new tree records to the simulation
fvsCompositeSum	Build a composite summary statistics table
fvsCutNow	Specify a thinning/harvest by setting the proportion of each tree record's sampling weight (trees/acre) that will be "cut" in the current cycle
fvsGetDims	Get the dimensions of important FVS data storage
fvsGetEventMonitorVariables	Get values of Event Monitor variables
fvsGetRestartcode	Get the current FVS restart code. See fvsRun for a list of the stop codes
fvsGetSpeciesAttrs	Get the values of species-specific data
fvsGetSpeciesCodes	Get a data.frame of species codes
fvsGetStandIDs	Get stand identification codes
fvsGetSummary	Get a data.frame of the summary statistics
fvsGetSVSDims	Get the max dimensions of important Stand Visualization System data storage

Function name	Description
fvsGetSVSObjectSet	Get the Stand Visualization System objects
fvsGetTreeAttrs	Get attributes of trees
fvsInteractRun	Run FVS and specify the R-script code that will be run at each stop point (where specified)
fvsLoad	Load the desired FVS program as a shared library
fvsMakeKeyFile	Make an FVS keyword file suitable for running FVS using fvsLoad, fvsSetCmdLine and fvsRun
fvsRun	Run FVS up to a designated stop point, year, or both. See fvsGetRestartcode for a way to find out the current stop point code
fvsSetCmdLine	Set the command line
fvsSetEventMonitorVariables	Set Event Monitor variables
fvsSetSpeciesAttrs	Set the values of species-specific data
fvsSetTreeAttrs	Set attributes of trees
fvsSetupSummary	Modifies the "summary" statistics so that it is ready to plot
fvsUnitConversion	Get FVS units conversion factor

References:

R Core Team. 2021. R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/

Rstudio. 2023. Shiny. Retrieved April 16, 2023, from https://shiny.rstudio.com/https://shiny.rstudio.com/

USDA Forest Service. 2023. The Forest Vegetation Simulator. https://www.fs.usda.gov/fvs/index.shtml

Weiskittel, A.; John Kershaw, J.; Nicholas Crookston, N.; [et al.]. 2017. The Acadian variant of the Forest Vegetation Simulator: Continued development and evaluation. In: Keyser, C.E.; Keyser, T.L. eds. Proceedings of the 2017 Forest Vegetation Simulator (FVS) e-Conference. e-Gen. Tech. Rep. GTR-SRS-224. Asheville, NC: U.S. Department of Agriculture Forest Service, Southern Research Station. p. 10-13.