



Änderungen am Programm ForestSimulator und den Paketen NW-FVAUtilization und TreeGrOSS

© 2014-2015 J. Nagel,
Nordwestdeutsche Forstliche Versuchsanstalt,
Grätzelstr.2, 37075 Göttingen
<http://www.nw-fva.de>

Dokumentation

Im Unterverzeichnis **\user\help** sind mehrere Dateien für die Programmdokumentation hinterlegt.

Datei	Inhalt	Seiten
installation_de.pdf	Hinweise zur Installation des ForestSimulators	5
NWFVA11_TreeGrOSS.pdf	Handbuch zum ForestSimulator und dem TreeGrOSS-Paket	243
FSSQLite.pdf	Bedienung der SQLite- Datenbankschnittstelle	7
FSSilviculture.pdf	Bedienung der Silviculture-Schnittstelle zur Definition und Einstellung waldbaulicher Szenarien	14
FSUtilization.pdf	Bedienung der Schnittstelle zur Sortierung und Berechnung von Biomasse und Nährelementen	7
FSChanges.pdf	Laufende Veränderungen, diese Datei	7
installation.pdf	Installation of the ForestSimulator	4

Veränderungen

Version 7.901 vom 01.12.2016

ForestSimulator.jar:

- Die Klasse TgTreatmentManager wurde überarbeitet und in diese das Silviculture eingebunden. In das Verzeichnis **\user\silviculture** wurden für verschiedene Waldentwicklungstypen Handlungsketten als xml-Datei hinterlegt. Diese können in den Treatmentmanager als Phase geladen werden.
- Die SQLite-Schnittstelle wurde überarbeitet und beschrieben.
- Das Paket NWFVAUtilization wurde komplett überarbeitet und für die Biomasse- und Nährstoffkonzentrationsfunktionen aus dem Projekt EnNa (FNR) angepasst. Für diese Modul wurde eine Beschreibung angefertigt.
- Es wurde eine neue Funktion mit in das Menü aufgenommen. Es kann jetzt nach „Liste durchforstet“ werden. Diese Funktion ist für den Vergleich von Durchforstungen nützlich, die im Rahmen von Lehrgängen auf Versuchflächen oder auf Marteloskopen durchgeführt werden.

Version 7.804 vom 04.11.2015

treegross.jar:

- Es wurde in der Klasse TaperFunctionBySchmidt in der Methode barkreduce() folgender Fehler behoben.
//Eiche,Stieleiche Rumpf, Husmann, Doebbeler 2013

$$\text{bark} = \text{Math.exp}(0.9262 + 0.7229 * \text{Math.log}(D)) * 1.03;$$

Version 7.803 vom 28.9.2015

ForestSimulator.jar:

- Es wurden in der Klasse MyMenubar zwei neue Menüpunkte aufgenommen, nämlich die Auswahl von Z-Bäumen und die Abwahl aller Z-Bäume. Erster Menüpunkt lädt die Behandlungstabelle neu und bestimmt anschließend die Z-Bäume. Die Anzahl der tatsächlich zu wählen den Z-Bäume richtet sich nach den Anzahl pro Hektar, dem Mischungsanteil und der Flächengröße. Darüber hinaus werden die Z-Bäume nur dann gewählt, wenn diese eine bestimmte Höhe haben und sie auf der Fläche nicht mit anderen Z-Bäumen konkurrieren.
- Wenn eine Fläche neu eingelesen wird, wird jetzt der st.trule.status auf Null gesetzt. Davon sind alle Einleseroutinen betroffen.
- Im Treatmentmanager3 wird jetzt st.trule.lastTreament auf Null gesetzt, wenn die Variable größer dem st.year ist. Dieses Rücksetzen wie auch das von st.status haben einen Fehler im Simulator behoben, dass kein Schirmschlag durchgeführt wurde, wenn die Fläche erneut in den Simulator geladen wurde.

TreeGrOSS.jar:

- Wenn eine Fläche neu eingelesen wird, wird jetzt der st.trule.status auf Null gesetzt. Davon sind alle Einleseroutinen betroffen. Betroffen ist die Methode treegrossXML2.

Version 7.801 vom 15.9.2015

ForestSimulator.jar:

- Es wurde eine umfangreiche SQLite Schnittstelle zu 3 Datenbanken geschaffen. Die Schnittstelle wird im Menü über **SQLite-Database** angesprochen. Die Datenbankdateien finden Sie unter \data_standsimulation. In der Maske wählen Sie die entsprechende Datenbank über die Reiterkarte. Es können jeweils auch andere Dateinamen mit passenden Datenbankstrukturen gewählt werden. In der Datenbankdatei **fsdatabase.db** können Sie jederzeit den aktuellen Bestand wie in die XML Datei abspeichern. Es wird jedes Mal beim Speichern eine neue Bestandes ID angelegt. Sie können den Bestand auch zu mehreren Zeitpunkten speichern. In der Datenbankdatei **fsplots.db** können Probekreise gespeichert werden, auf denen die Bäume mit Polarkoordinaten erfasst wurden. Kreisförmige Bestandesflächen lassen sich in diese Datenbank mit Polarkoordinaten speichern. Die Datenbankdatei **nutzungsplaner.db** ist eine Schnittstelle zu der Android App Nutzungsplaner (siehe Google Play).
Die SQLite Datenbankdateien sind sehr ähnlich den Microsoft Access Dateien (*.mdb). Der Vorteil der SQLite Datenbanken ist, dass diese unter Linux wie MS Windows laufen und seitens SQLite gute Datenbanktreiber für Java zur Verfügung stehen. Mit Java 1.8 hat Oracle leider die Bereitstellung von MS Access Treiber eingestellt. SQLite Datenbanken können Sie mit vielen Programmen bearbeiten. Eine gute Möglichkeit ist das kostenlose Firefox Add-on **SQLite -Manager**. Einfach im Firefox über „Extras“ installieren und dann über „Extras“ öffnen. Mit diesem Programm können Sie die Datenbankdateien über SQL-Befehle bearbeiten und auch mit CSV Dateien (Excel, Calc) befüllen. Das Programm erklärt sich mehr oder weniger intuitiv.
- Im ForestSimulator ist es nun möglich, Bestände nach Baumschichten auszuwerten. Es ist die Bildung von 4 Schichten (4= Überhalt, 1= Oberstand, 2= Zwischenstand, 3=Verjüngung oder 0=keine Schichtansprache) möglich. Die Schichten können entweder über die Datenbanken oder das XML-Format eingelesen werden. Darüber hinaus ermöglicht jetzt auch das Fenster „Bäume hinzufügen“ und der Dialog „Stand Data“ die Eingabe der Schichten. In dem Fenster StandInfo finden Sie eine Checkbox mit der Sie die Ausgabe nach Schichten aktivieren können. Für die Verjüngungsplatzhalter (Bäume < 7cm BHD) wird nur der Deckungsgrad ausgegeben. Die Bäume der Verjüngungsschicht 3, welche größer gleich

7 cm BHD sind, werden getrennt beschrieben. Bei langfristigen Simulationen, macht jedoch die Auswertung nach Schichten keinen Sinn, weil das Programm zur Zeit noch nicht über eine Routine zu einer erneuten Schichtenansprache bzw. einer automatischen Baumschichtenklassifikation verfügt. Die Ausgabetabelle Einzelbaumwerte wurde um die Spalte Schicht erweitert.

TreeGrOSS.jar:

- Die Klasse **Groups** wurde um zwei Methoden zu Berechnung der Stammzahl und der Grundfläche des ausscheidenden Bestandes erweitert. Darüber hinaus wurde die Berechnung der Methode **HeightCurve()** verbessert, so dass jetzt auch die Höhe über die Einheitshöhenkurve berechnet werden kann. Die Klasse Groups wird im ForestSimulator zur Berechnung der Schichtenwerte verwendet.
- In der Klasse **Stand** wird jetzt in der Methode **addTreeFromNaturalIngrowth()** die Schicht neuer Bäume automatisch auf layer=3 gesetzt.

Version 7.708 vom 3.9.2015

ForestSimulator.jar:

- Das Programm wurde für die Linux Benutzung weiter optimiert und zahlreiche kleine Fehler entfernt.
- Es wurde in die Maske für die waldbauliche Berechnung die QD-Regel als Durchforstungsverfahren aufgenommen.
- Es wurde in der Maske für die waldbauliche Behandlung die Möglichkeit eines Lochhiebes realisiert.

TreeGrOSS.jar:

- Die TreeGrOSS Klasse Competition wurde um eine Methode erweitert. Mit dieser kann der maximale Kronenschlußgrad, welcher sich aus der Density Funktion berechnet, ermittelt werden.
- Bei der Neuberechnung des c66c und des c66cyx wird jetzt geprüft, ob dieser realistische Werte annimmt. Die c66c Werte wurden jetzt auf den maximalen Kronenschlußgrad begrenzt. Im Zuge von Berechnungen mit den Daten der BWI waren immer wieder sehr hohe Freistellungswerte (c66c und c66cxy) aufgetreten, die zu unplausiblen Durchmesserzuwachs führten.
- In der Klasse TreatmentElements wurde the Methode thinByQD() eingefügt. Diese Methode durchforstet den Bestand ähnlich wie das QD-Verfahren, welches in Rheinland-Pfalz praktiziert wird. Mit der Methode alle bedrängenden nicht Z-Bäume entfernt, die Kronenkontakt+0.2m zum Z-Baum haben. Dieser Methode arbeitet unabhängig von der Bestandesdichte. Je mehr Z-Bäume ausgewählt werden, desto mehr Füllbäume werden durchforstet.
- In der Klasse TreatmentElements wurde die Methode harvestByGaps() eingefügt. Die Methode fügt einen Lochhieb durch. Ist ein Z-Baum erntereif, so werden der Z-Baum und alle Bäume in einem 12 m Radius entfernt, sofern diese größer als die halbe Höhe des Z-Baums sind. Die Nutzung läuft solange, bis da maximale Erntevolumen erreicht ist oder keine Z-Bäume mehr auf der Fläche sind.

- In der Klasse Treatment2 wurden die beiden zuvor genannten Methoden über die Methode executeManager2 ansprechbar gemacht.

Version 7.707 vom 3.3.2015

ForestSimulator.jar:

- Es gab eine Fehlermeldung auf Linux Systemen, wenn ein Bestand im Standeditor aufgerufen wurde, weil der Backslash hardcodiert war. Sollte jetzt funktionieren.

Version 7.706 vom 22.7.2014

TreeGrOSS.jar:

- TaperfunktionBy Schmidt geändert und die Rindenfunktion für Eiche, Buche, Fichte, Kiefer, Esche und Ahorn nach dem Bestbericht eingesetzt,

NWFVA-Utilization.jar:

- Nährstoffbilanzierung die Reiterkarte Durchforstungen korrigiert und die Bäume mit Mortalität als Entnahmegrund entfernt.
- Neue Datei BiomassNWGermany.xml mit Werten für 311,321, 411,421.
- Fehler im BiomassPanel berichtigt, es wird jetzt bei wiederholter Kalkulation auf die richtigen Sortimente zugegriffen.

ForestSimulatorNWGermanyBC4.xml

- Kommentar bei Esche und Ahorn Volumenfunktion von Buche nach Eiche geändert. Die Formel war immer Eiche und keine Buche.

Version 7.705 vom 15.7.2014

ForestSimulator.jar:

- Die Routine die ausgeführt wird, wenn der Knopf "wachsen" mit dem Baumsymbol gedrückt wird, wurde korrigiert. Jetzt wird vor dem Wachstumsschritt die Mortalität ausgeführt. Dies funktionierte in älteren Versionen, bis die Mortalität aus dem Wachstum genommen wurde.

Version 7.704 vom 12.7.2014

TreeGrOSS.jar:

- Die TreeGrOSS Klasse GenerateXY wurde um eine interne Methode shuffleTrees() erweitert. Mit dieser Methode können die nach Durchmesser sortierten Bäume, welche keine Koordinaten haben über eine Zufallszahl umsortiert werden, damit insbesondere bei der Generierung von Trupps nicht mehr die dicken Bäume in einen Trupp gestellt werden. Die Methode ist in der Generierung dann aktiv geschaltet, wenn Trupp, Horst gewählt wird. Nach der Generierung der Koordinaten werden wie Bäume wieder nach BHD sortiert.

Version 7.703 vom 7.8.2014

ForestSimulator.jar:

- Der Simulationsdialog wurde überarbeitet und korrigiert. In den Spalten der Baumartentabelle werden jetzt die Z-Bäume pro Hektar Reinbestand angegeben. Für die Simulation wird die Z-Baumangabe mit der Flächengröße und dem gewünschten Mischungsanteil multipliziert und daraus die Z-Baumzahl für die Simulation abgeleitet.

- Die Ausgabetabelle für die Bestandesdaten wurde erweitert. Man findet jetzt Werte für Stammzahl, Grundfläche und Volumen des ausscheidenden Bestandes getrennt nach Mortalität und Nutzung aufgeführt.
- Es steht eine neue Einlesemöglichkeit zur Verfügung. Hier können die Daten aus der SQLite Datenbank der Android App NutzungsPlaner direkt übernommen werden (s. <https://play.google.com/store/apps/details?id=ftools.nutzung&hl=de>).
- FSChanges.pdf über das Menü aufrufbar. Es soll die Veränderungen anzeigen.
- Das neue offizielle Handbuch (Hansen und Nagel 2014) wurde als Ersatz der Anleitung integriert.

NWFVA-Utilization.jar

- Die Sortierung wurde verbessert. Es lassen sich jetzt die Zugaben prozentual und absolut darstellen. Die Ergebnisse werden in der Ergebnisdatei sortierung.xml ausgegeben und mit der überarbeiteten Datei treegrosslogging.xls entsprechend dargestellt. Man findet jetzt für jedes Sortiment das Volumen nach Huber mit und ohne Rinde. Diese ist ohne Zugaben berechnet. Gleichzeitig wird das Sortiment mit Zugaben nach der Schaffformkurve mit und ohne Rinde ausgegeben.

ForestSimulatorNWGermanyBC4.xml

- Die Einstellungsdatei mit den Funktions- und Modellparametern für Nordwestdeutschland wurde überarbeitet. Sie enthält neue Vorgaben für die Z-Baumzahlen.

Version 7.7

Neu überarbeitete Version. Diese unterscheidet sich von der Version 7.6 hauptsächlich dadurch, dass das TreeGrOSS Paket komplett überarbeitet wurde.

Literatur:

Hansen, J.; Nagel, J. (2014): Waldwachstumskundliche Softwaresysteme auf Basis von TreeGrOSS - Anwendung und theoretische Grundlagen. Beiträge aus der NW-FVA, Band 11, 224 S. (http://webdoc.sub.gwdg.de/univerlag/2014/NWFVA11_TreeGrOSS.pdf)

