LANDIS-II Model v6.2

User Guide

Robert M. Scheller1

James B. Domingo2Lesley C. Bross1

1Portland State University

2Green Code, LLC

Last Revised: November 30, 2015

# Table of Contents

[1 Introduction 3](#_Toc436664226)

[1.1 Major releases 3](#_Toc436664227)

[1.1.1 Version 6.0 3](#_Toc436664228)

[1.2 Minor releases 4](#_Toc436664229)

[1.2.1 Version 6.2 4](#_Toc436664230)

[1.2.2 Version 6.1 4](#_Toc436664231)

[1.2.3 Version 5.1 4](#_Toc436664232)

[1.3 Acknowledgements 4](#_Toc436664233)

[2 Running LANDIS-II 6](#_Toc436664234)

[2.1 Different Versions on a Computer 6](#_Toc436664235)

[2.1.1 LANDIS\_VERSION Environment Variable 6](#_Toc436664236)

[2.2 Working Directory 7](#_Toc436664237)

[3 Input File Format 8](#_Toc436664238)

[3.1 Text Input Files 8](#_Toc436664239)

[3.1.1 Line Types 8](#_Toc436664240)

[3.1.2 End-of-line Comments 8](#_Toc436664241)

[3.1.3 Parameter and Table Names 8](#_Toc436664242)

[3.1.4 Data Line With Single Parameter 9](#_Toc436664243)

[3.1.5 Numeric Parameter Values 9](#_Toc436664244)

[3.1.6 Yes/No Parameter Values 10](#_Toc436664245)

[3.1.7 Text Parameter Values 10](#_Toc436664246)

[3.1.8 File Parameter Values 10](#_Toc436664247)

[3.1.9 Output File Parameters & Non-existent Directories 11](#_Toc436664248)

[3.1.10 Tables of Parameter Values 11](#_Toc436664249)

[3.1.11 LandisData Parameter 12](#_Toc436664250)

[3.2 Raster Input Maps 12](#_Toc436664251)

[4 Output Map Formats 13](#_Toc436664252)

[5 Scenario Input File 14](#_Toc436664253)

[5.1 Example File 14](#_Toc436664254)

[5.2 LandisData 14](#_Toc436664255)

[5.3 Duration 14](#_Toc436664256)

[5.4 Species 15](#_Toc436664257)

[5.5 Ecoregions 15](#_Toc436664258)

[5.6 EcoregionsMap 15](#_Toc436664259)

[5.7 CellLength 15](#_Toc436664260)

[5.8 Extensions Table 15](#_Toc436664261)

[5.8.1 Extension (Plug-in) Column 15](#_Toc436664262)

[5.8.2 Initialization File Column 15](#_Toc436664263)

[5.8.3 Succession Extension 16](#_Toc436664264)

[5.8.4 Disturbance Extensions 16](#_Toc436664265)

[5.8.5 DisturbanceRandomOrder 16](#_Toc436664266)

[5.8.6 Other Extensions 16](#_Toc436664267)

[5.9 RandomNumberSeed 16](#_Toc436664268)

[6 Species Input File 17](#_Toc436664269)

[6.1 Example File 17](#_Toc436664270)

[6.2 LandisData 19](#_Toc436664271)

[6.3 Table Fields 19](#_Toc436664272)

[6.3.1 Species name 19](#_Toc436664273)

[6.3.2 Longevity 19](#_Toc436664274)

[6.3.3 Sexual maturity 19](#_Toc436664275)

[6.3.4 Shade tolerance 19](#_Toc436664276)

[6.3.5 Fire tolerance 19](#_Toc436664277)

[6.3.6 Effective seeding distance 19](#_Toc436664278)

[6.3.7 Maximum seeding distance 19](#_Toc436664279)

[6.3.8 Vegetative reproduction probability 20](#_Toc436664280)

[6.3.9 Minimum resprouting age 20](#_Toc436664281)

[6.3.10 Maximum resprouting age 20](#_Toc436664282)

[6.3.11 Post-fire regeneration 20](#_Toc436664283)

[7 Ecoregions Input File 21](#_Toc436664284)

[7.1 Example File 21](#_Toc436664285)

[7.2 LandisData 21](#_Toc436664286)

[7.3 Table Fields 21](#_Toc436664287)

[7.3.1 Active 21](#_Toc436664288)

[7.3.2 Map Code 21](#_Toc436664289)

[7.3.3 Ecoregion Name 21](#_Toc436664290)

[7.3.4 Ecoregion Description 22](#_Toc436664291)

# Introduction

This document is a member of the set of documents that describes how to use the LANDIS-II model. This document describes:

* how the user runs a scenario with the model,
* general formatting rules that apply to every input file, and
* the input files which are common to all scenarios.

Other user guides in the documentation set describe the extensions that are available for LANDIS-II. The user guide for a LANDIS-II extension describes what the extension does as well as the format of its input files.

This document assumes that the typical user is an **ecologist**. Because this document focuses on how to run the model and the format of input data, the reader should consult the document *LANDIS-II Conceptual Model Description* for a conceptual description of the model.

## Major releases

### Version 6.0

LANDIS-II v6.0 represents a major restructuring of model architecture, with substantial improvements for usability. The differences between this version and the previous version (5.1) include:

* We have incorporated the Geospatial Data Abstraction Library (GDAL) and therefore users can now input and output maps from many more raster file formats. Currently available formats include ENVI (\*.bin), bitmap (\*.bmp), ERDAS 7.4 or LAN (\*.gis), IMAGINE (\*.img), and GeoTIFF (\*.tif).
* The initial community files have moved to the individual succession extension. This allows more flexibility for developing future input files that could contain more information in addition to species and ages.
* All extensions are provided with a complete example scenario.

## Minor releases

### Version 6.2

* v6.2 is a minor update and enables the InitializePhase2() and CleanUp() methods for LANDIS-II extension developers. There are no user enhancements in this release.

### Version 6.1

LANDIS-II v6.1 is a minor update. Enhancements in this release include:

* The RandomNumberSeed is always printed to the Landis log. Previously the seed was not printed if it was supplied by the user.
* Computers using decimal separators other than ‘.’ will be warned when the model initializes that the extension(s) may not work or the output may be incorrect. See Section 3.1.5 for more information about formatting numeric parameters in a scenario file.
* Uninstalling LANDIS-II now uninstalls any extensions as well.

### Version 5.1

The differences between this version and the previous version (5.0) include:

* The addition of the LANDIS\_VERSION environment variable (see section 2.1.1).
* A change to the 3rd section of the extensions table in the scenario file (see section 5.8.6 *Other Extensions*). Previously, only output extensions were allowed in this part of the table. It now accepts any other types of extensions besides succession and disturbance. This change was motivated by work done on a meta-population extension for invasive species.

## Acknowledgements

Funding for the development of LANDIS-II has been provided by the Northern Research Station (Rhinelander, Wisconsin) of the U.S. Forest Service. Valuable contributions to the development of the model and extensions were made by Brian R. Sturtevant, Eric J. Gustafson, and David J. Mladenoff.

# Running LANDIS-II

Most users use a console interface to execute the LANDIS-II model; these users run the model from a command prompt. To open a command prompt window:

* Windows 98/ME: from Start menu, select Programs⏵ MS‑DOS Prompt.
* Windows NT/2000/XP: from Start menu, select Programs⏵ Accessories⏵Command Prompt.

To run the model, the user enters the model name followed by the name of the scenario file (see chapter 4):

C:\> landis-ii *scenario-file*

For convenience, the model’s name may be abbreviated:

C:\> landis *scenario-file*

There are two Widgets available for those users who prefer a windows (graphical) user interface. The Launcher and Replicator provide the capability to select and run one or more scenario files. For more information about the Widgets, consult the LANDIS-II Widgets User Guide.

## Different Versions on a Computer

The LANDIS-II model is designed so that different versions can be installed side-by-side on a computer. This allows the user to compare the model’s behavior and output between versions.

Any version can be specified explicitly at the command prompt by appending the version to the model’s name with a hyphen. For example, the current version can be run as follows:

C:\> landis-5.1 *scenario-file*

Referencing a particular version explicitly is allowed even if it is the only version installed on the computer.

### LANDIS\_VERSION Environment Variable

If two or more versions of the model are installed on a computer, the two commands mentioned above without any version number ­– landis-ii and landis ­– will run a particular version based upon the existence of the LANDIS\_VERSION environment variable.

If the LANDIS\_VERSION variable is not set, then those two commands will run the newest version. For example, if versions 5.0 and 5.1 are installed, then both the landis-ii and landis commands will run version 5.1.

If the LANDIS\_VERSION variable is set, then those two commands will run the version specified by the variable. For example, if versions 5.0 and 5.1 are both installed on a server, but a user wishes to use version 5.0 as the default for her work, she can set the LANDIS\_VERSION variable to “5.0”. Then both landis-ii and landis commands will run version 5.0.

This environmental variable can be temporarily set for a session at the command prompt:

C:\> set LANDIS\_VERSION=5.0

or it can be set permanently via the Window’s System Properties:

* Right-click My Computer, and in the context menu, select Properties⏵Advanced tab⏵Environment Variables button.

## Working Directory

The directory (folder) where the user runs LANDIS-II is referred to as the **working directory**. The user should be aware that LANDIS-II uses the working directory when interpreting input parameters which are relative path names for files (see section ).

# Input File Format

LANDIS-II uses two types of input files: text files and raster maps.

## Text Input Files

### Line Types

There are three types of lines in a text input file:

* **Blank line** – a line with no characters or just whitespace. The program ignores blank lines.
* **Comment line** – a line whose first non-whitespace characters are the comment marker ">>" (two adjacent greater-than symbols). The text of the comment follows this marker, and may contain any characters. Typically, they are used for section titles and column headings associated with tabular data. Like blank lines, the program ignores comment lines.

>> Here is a sample comment line.

>>A comment line with spaces before marker

* **Data line** – any line that is not a blank line or a comment line.

### End-of-line Comments

A comment can be placed at the end of any data line in a text input file. An **end-of-line comment** starts with a marker which is two adjacent "<" symbols (i.e., "<<"), and extends to the end of the current line. The user can use an end-of-line comment to provide additional information about an input parameter, such as its units:

Sample\_Parameter 500 << Units: Mg/ha

Like blank lines and comment lines, the program ignores end-of-line comments.

### Parameter and Table Names

The name of a parameter or a table of parameters is a sequence of one or more non-whitespace characters. Examples:

SampleParameter

#ofCycles

Table-of-data

### Data Line With Single Parameter

A common format for a data line is a single parameter. In this format, the parameter’s name (see section 3.1.3) is first, followed by the parameter’s value.

*parameter-name parameter-value*

Whitespace is required between the name and value. Whitespace may precede the name and may follow the value. Examples:

Timestep 10

CellLength 28.5 << meters

Description "southeast pine barrens"

### Numeric Parameter Values

Values for numeric parameters can be specified in integer format, floating-point format, or scientific notation.

CellLength 20

CellLength 20.0

CellLength 2e1 << scientific notation = 2 \* 10^1

<< = 2 \* 10 = 20

ParameterX -.1234

ParameterY +98.765e-22

Some numeric parameters are **integer parameters** which means they only accept integer values.

For user convenience, thousands-separators can be used for large values.

Population 2,000,000

Note: LANDIS-II does not support internationalization. It expects the values for decimals and thousands separators that are included in the character set of the computer on which it runs. For example: if a numeric value has a ‘.’ for a decimal separator but the computer settings use a ‘,’ the parameter value will not be correctly set.

Because most LANDIS-II parameter files are written using a ‘.’ for the decimal separator, the model prints a warning on computers that do not. This does not prevent users from creating and successfully running scenario files with a non-standard character set.

### Yes/No Parameter Values

Yes/no parameters are also known as **boolean parameters**. Acceptable yes values are “yes” and “y”, while acceptable no values are “no”, “n”, and “-” (useful for long columns to make yes values stand out more visually). These values are case-insensitive, so uppercase letters can be used as well. For example, “Yes”, “YES”, and “Y” are also valid yes values.

### Text Parameter Values

Values for text parameters (also known as **string parameters**) are specified using two formats.

* **Unquoted** – A text value must be a sequence of one or more non-whitespace characters.

Species pinurigi

File C:\dir\sub-dir\input\_data.txt

* **Quoted** – The text value is bracketed by a pair of single quotes ('*text*') or double quotes ("*text*"). A quoted text value may include whitespace. It may also have no characters at all (i.e., the value is an empty string) which is represented by a pair of adjacent quotes.

Species "pinurigi"

Description '' << empty string; 0 characters

File "model runs\scenario XYZ\file.txt"

### File Parameter Values

File parameters accept text values (see section ) which represent valid file names on the user’s computer system.

file.txt

some-dir/SPECIES

C:\landis-ii\my project\ecoregions.txt

../../landis-ii/climate/species.txt

"model runs/scenario XYZ/file.txt"

Please note that a **relative file name**, i.e., one that does not refer to the root directory on one of the system's drives, is interpreted relative to the working directory (see section 2.2). It's important that the user keep this point in mind when organizing the data files for the program.

For example, suppose the program is reading an input file "dir/file.txt", and encounters a filename parameter whose value is "more.dat". The file that's being referenced is in the working directory where the program is being executed, i.e., "./more.dat"; it's not the file "dir/more.dat".

***User Tip****: Beware of file type extensions. Users often hide their file types in their file explorer. Sometimes, the result is that a user creates a file with a double file extension:* species.txt.txt *that the model will not recognize.*

#### Variables

Some file parameters allow the use of variables in their values. A variable is a name enclosed by curly braces: "{*name*}". When the file parameter value is used by the program, the variable’s name and braces are replaced with the variable’s value. The set of variables that can be used varies among file parameters. The documentation for a file parameter will specify which variables, if any, are permitted.

As an example, suppose an output extension generates output maps showing shade at the sites on the landscape, one map for each timestep that the extension runs. The extension’s input file has a file parameter whose value represents the maps’ names. The extension defines a variable called “timestep” for this file parameter; its value is the current timestep. The user puts this variable in the parameter value to ensure that each map name is unique.

MapNames "output/shade-{timestep}.gis"

### Output File Parameters & Non-existent Directories

If the name of an output file contains one or more directories that do not exist, the program will make these directories before writing the output file. For example, suppose an output file name is "dir/sub\_dir/file.txt" and the directory "dir" does not exist. The program will make the directory "dir" and its subdirectory "dir/sub\_dir" before creating the output file "dir/sub\_dir/file.txt".

Tables of Parameter Values

A table of parameter values has one or more columns. Column headings are, by convention, specified on comment lines as an aid to users.

>> Map

>> Active Code Name Description

>> ------ ---- ----- -----------

yes 1 eco1 "Mesic soils"

yes 2 eco2 "Sandy outwash plain"

- 400 water "lakes, rivers"

- 99 urban "urban area"

Each row of parameter values is a data line. Blank lines and comment lines may be interspersed among the table’s rows.

If a table requires a name, the name appears on a separate data line before the table’s rows. Table names are usually not required when an input file contains only one table. If an input file has two consecutive parameter tables, the name of the second table will be required to denote where the first table has ended and the second table begins.

LandisData Parameter

The first parameter in a LANDIS-II text input file is the LandisData parameter. This parameter requires a text value (see section 3.1.7) which uniquely identifies the type of data in the file. Examples:

LandisData Species

LandisData "Initial Communities"

This parameters helps the program as it starts reading a text input file to quickly detect if the user has provided the wrong type of input data.

## Raster Input Maps

An input map is a raster data file. LANDIS-II uses the extension in the file’s name to determine the file’s format.

We have incorporated the Geospatial Data Abstraction Library (GDAL) and therefore users can now input maps from many more raster file formats. For a list of currently available formats, visit <http://www.gdal.org/formats_list.html>

***Note****: Legacy \*.gis files often contain negative numbers if they were created via ArcGIS. In ArcGIS, you can use the raster calculator to add the number of the lowest number to all values. For example, if -30 is the minimum, add 30 to all values.* ***Caution:*** *This only works for values that are not explicitly tied to any other input data, such as the initial community map.*

# Output Map Formats

Beginning with LANDIS-II v6.0, users can output maps in many different formats. These formats are derived from the Geospatial Data Abstraction Library, from which the LANDIS Spatial Modeling Library is derived.

The following extensions and file types are currently supported:

.bin ENVI .hdr-Labelled Raster

.bmp Windows Bitmap (**Note**: bmp is a very ‘shallow’ format that will only work with output data < 255).

.img Erdas Imagine

.tif GeoTIFF

.gis ERDAS 7.4 or LAN

.vrt Virtual Raster

.ingr Intergraph Raster

# Scenario Input File

This file describes a single model scenario.

## Example File

LandisData Scenario

Duration 300

Species species.txt

Ecoregions ./ecoregions.txt

EcoregionsMap ./ecoregions.gis

CellLength 100 << meters, so cell area = 1 ha

>> Succession Extension Initialization File

>> -------------------- -------------------

"Age-only Succession" succession.txt

>> Disturbance Extensions Initialization File

>> ---------------------- -------------------

"Base Wind" wind.txt

"Base Fire" fire.txt

DisturbancesRandomOrder yes

>> Output Extensions Initialization File

>> ----------------- -------------------

"Max Species Age" max-spp-age.output.txt

Reclass reclass.output.txt

RandomNumberSeed 4,357

## LandisData

This parameter’s value must be "Scenario".

## Duration

This parameter represents how long the scenario lasts. Value: integer > 0. Units: years.

## Species

This parameter is the file which contains the species’ parameters (see chapter ).

## Ecoregions

This parameter is the file which contains the definitions of the ecoregions on the landscape (see chapter ).

## EcoregionsMap

This parameter is the input map showing where the ecoregions are located on the landscape. Each cell value must be one of the map codes listed in the ecoregions input file (see chapter ).

## CellLength

This parameter is the length of a cell’s edge in the ecoregions map (see section ). Value: number > 0. Units: meter. The cells are assumed to be square. If the ecoregions map has a cell length in its metadata, that cell length is compared with this parameter. The program reports the result of the comparison. **If they differ, the program uses the value specified by this parameter**. In other words, this parameter overrides the cell length in the map’s metadata.

This parameter is optional. If it is not present in the file, then the ecoregions map must have a cell length.

## Extensions Table

This table lists the extensions (plug-ins) used in the scenario. Each row in the table specifies one extension. Extensions must be in the table in an order based on their type: succession first, then disturbance, and finally output. The table has two columns.

### Extension (Plug-in) Column

This column is the extension’s name. The user can view a list of extensions installed on her computer by looking at the on-line LANDIS-II documentation page (Start⏵All Programs⏵Landis-II⏵ Documentation).

### Initialization File Column

This column is the file with the input parameters used by the extension. The extension’s user guide describes the format of that file.

### Succession Extension

The first extension in the table must be a succession extension. The table must have just one succession extension.

### Disturbance Extensions

A scenario may have zero or more disturbance extensions.

### DisturbanceRandomOrder

This yes/no parameter determines if the disturbances occur in random order (see section 5.3.1 in the model’s conceptual description). A yes value directs LANDIS-II to run any disturbance extensions that occur at the same timestep in a random order. A no value directs the model to run disturbance extensions in the order that they appear in the table.

The parameter is optional. If it is not in the table, then the default value is no (LANDIS-II runs disturbance extensions in the order that they appear in the table).

### Other Extensions

The table may have one or more other extensions whose types are not succession or disturbance, for example, an output extension or a meta-population extension.

## RandomNumberSeed

This parameter is the integer value that the program uses to initialize its internal random-number generator. Value: integer ≠ 0. This parameter allows the user to reproduce the same output for a scenario; very useful when working with a stochastic model like LANDIS-II.

The parameter is optional. If it is not present in the file, the program uses a random value based on the current system time to initialize the generator. The program displays this initialization value so that it can be used to reproduce the scenario’s output if need be.

# Species Input File

This file contains a table of basic parameters for tree species. Each row in the table has the parameters for one species.

Extensions may require additional species-specific parameters, for example, reclassification map coefficient, leaf longevity, foliage combustibility. These additional parameters are specified in the initialization files for those extensions.

## Example File

See Figure 1.

Figure 1 – Example of species input file.

LandisData Species

>> Sexual Shade Fire Seed Disperal Dist Vegetative Sprout Age Post-Fire

>> Name Longevity Maturity Tol. Tol. Effective Maximum Reprod Prob Min Max Regen

>> ---- --------- -------- ----- ---- --------- ------- ----------- ---------- --------

abiebals 200 25 5 1 130 160 0.0 0 0 none

acerrubr 150 10 4 1 100 200 0.5 0 100 none

acersacc 300 40 5 1 100 200 0.1 10 60 none

betualle 300 40 4 2 100 400 0.1 10 180 none

betupapy 100 30 2 2 200 5,000 0.5 10 70 none

piceglau 300 25 3 2 30 200 0.0 0 0 none

pinubank 100 15 1 3 30 100 0 0 0 serotiny

pinuresi 200 35 2 4 30 275 0 0 0 none

pinustro 400 40 3 3 60 210 0 0 0 none

poputrem 100 20 1 1 1,000 5,000 0.9 10 100 none

querelli 300 35 2 2 30 3,000 0.75 30 280 none

quermacr 300 30 2 5 30 1,000 0.9 10 300 none

querrubr 250 25 3 2 30 3,000 0.75 20 200 none

tiliamer 250 30 4 2 30 120 0.1 10 200 none

tsugcana 450 30 5 2 30 100 0 0 0 none

## LandisData

This parameter’s value must be "Species".

## Table Fields

### Species name

This text parameter is the species’ name. Each name must appear only once in the table. The names can appear in any order in the table.

### Longevity

This parameter is the species’ maximum age. Value: integer > 0. Units: years.

### Sexual maturity

This parameter is the age at which the species matures sexually. Value: 0 < integer ≤ Longevity. Units: years.

### Shade tolerance

This parameter represents the species’ tolerance to shade. Value: integer between 1 (lowest tolerance) and 5 (highest tolerance).

### Fire tolerance

This parameter represents the species’ tolerance to fire. Value: integer between 1 (lowest tolerance) and 5 (highest tolerance).

### Effective seeding distance

This parameter is the species’ effective distance for dispersing seeds. Value: 0 < integer ≤ Max Seed Dist. Units: meters.

**Optional Universal Dispersal:**

The User may enter –1 or “uni” for the effective seed distance, indicating that this species has seeds that are universally distributed, i.e., there are seed source limitations to the availability of seeds. Such may be the case if a species, such as aspen, has a long-lived seed bank.

### Maximum seeding distance

This parameter is the species’ maximum distance for dispersing seeds. Value: integer > 0. Units: meters.

### Vegetative reproduction probability

This parameter is the probability that the species resprouts (vegetation reproduction). Value: 0.0 ≤ number ≤ 1.0

### Minimum resprouting age

This parameter is the minimum age required for the species to resprout. Value: 0 < integer ≤ Longevity. Units: years.

### Maximum resprouting age

This parameter is the maximum age required for the species to resprout. Value: Min Sprout Age ≤ integer ≤ Longevity. Units: years.

### Post-fire regeneration

This parameter describes the form of reproduction (regeneration) that the species does after fire events. Valid values are "none", "serotiny" or "resprout". See the *Post-Fire Regeneration* section in the model’s conceptual description.

# Ecoregions Input File

A landscape can be divided into ecologically defined **land types** or **ecoregions**. A subset of the landscape’s sites are **active** during a simulation. Some sites within the region may be inactive because they represent locations where forests do not grow (for example, bodies of water, urban areas).

This file contains a table of ecoregion definitions. Each row in the table has one ecoregion.

## Example File

LandisData Ecoregions

>> Map

>> Active Code Name Description

>> ------ ---- ----- -----------

- 1 water water

Y 22 eco22 MesicLoam

yes 3 eco3 "Sandy outwash plain"

no 0 none "not in region-of-interest"

## LandisData

This parameter’s value must be "Ecoregions".

## Table Fields

### Active

This yes/no parameter indicates whether the ecoregion’s sites are active or not during the simulation.

### Map Code

This parameter is the code used for the ecoregion in the input map (see section ). Value: 0 ≤ integer ≤ 65,535. Each ecoregion’s map code must be unique. Map codes do not have to appear in any order, and do not need to be consecutive.

### Ecoregion Name

This text parameter is the ecoregion’s name. Because ecoregion names are used in other input files, it is recommended that names be kept short.

### Ecoregion Description

This text parameter describes the ecoregion for the user’s benefit.