

LANDIS-II Model v5.1 User Guide

Robert M. Scheller
James B. Domingo

University of Wisconsin-Madison

Last Revised: April 20, 2006

Table of Contents

1	INTRODUCTION	3
1.1	What's new in Version 5.1	3
1.2	Acknowledgements	3
2	RUNNING LANDIS-II	4
2.1	Different Versions on a Computer	4
2.1.1	<i>LANDIS_VERSION Environment Variable</i>	4
2.2	Working Directory	5
3	INPUT FILE FORMAT	6
3.1	Text Input Files	6
3.1.1	<i>Line Types</i>	6
3.1.2	<i>End-of-line Comments</i>	6
3.1.3	<i>Parameter and Table Names</i>	6
3.1.4	<i>Data Line With Single Parameter</i>	7
3.1.5	<i>Numeric Parameter Values</i>	7
3.1.6	<i>Yes/No Parameter Values</i>	7
3.1.7	<i>Text Parameter Values</i>	7
3.1.8	<i>File Parameter Values</i>	8
3.1.9	<i>Output File Parameters & Non-existent Directories</i>	9
3.1.10	<i>Tables of Parameter Values</i>	9
3.1.11	<i>LandisData Parameter</i>	10
3.2	Raster Input Maps	10
3.2.1	<i>ERDAS 7.4 (*.gis)</i>	10
4	SCENARIO INPUT FILE	11
4.1	Example File	11
4.2	LandisData	11
4.3	Duration	12
4.4	Species	12
4.5	Ecoregions	12
4.6	EcoregionsMap	12
4.7	CellLength	12
4.8	InitialCommunities	12
4.9	InitialCommunitiesMap	12
4.10	Extensions Table	13
4.10.1	<i>Extension (Plug-in) Column</i>	13
4.10.2	<i>Initialization File Column</i>	13
4.10.3	<i>Succession Extension</i>	13
4.10.4	<i>Disturbance Extensions</i>	13
4.10.5	<i>DisturbanceRandomOrder</i>	13
4.10.6	<i>Other Extensions</i>	13
4.11	RandomNumberSeed	14
5	SPECIES INPUT FILE.....	15
5.1	Example File	15

5.2	LandisData	17
5.3	Table Fields	17
5.3.1	<i>Species name</i>	17
5.3.2	<i>Longevity</i>	17
5.3.3	<i>Sexual maturity</i>	17
5.3.4	<i>Shade tolerance</i>	17
5.3.5	<i>Fire tolerance</i>	17
5.3.6	<i>Effective seeding distance</i>	17
5.3.7	<i>Maximum seeding distance</i>	17
5.3.8	<i>Vegetative reproduction probability</i>	18
5.3.9	<i>Minimum resprouting age</i>	18
5.3.10	<i>Maximum resprouting age</i>	18
5.3.11	<i>Post-fire regeneration</i>	18
6	ECOREGIONS INPUT FILE	19
6.1	Example File	19
6.2	LandisData	19
6.3	Table Fields	19
6.3.1	<i>Active</i>	19
6.3.2	<i>Map Code</i>	19
6.3.3	<i>Ecoregion Name</i>	19
6.3.4	<i>Ecoregion Description</i>	20
7	INITIAL COMMUNITIES INPUT FILE.....	21
7.1	Example File	21
7.2	LandisData	22
7.3	Initial Community Class Definitions	22
7.3.1	<i>MapCode</i>	22
7.3.2	<i>Species Present</i>	22
7.3.3	<i>Grouping Species Ages into Cohorts</i>	22

1 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.

1.1 What's new in Version 5.1

This document describes the current version (5.1) of the LANDIS-II model. 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 4.10.6 *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.

1.2 Acknowledgements

Funding for the development of LANDIS-II has been provided by the North Central 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.

2 Running LANDIS-II

Currently, only a console interface is available for the LANDIS-II model; therefore, the user must 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
```

2.1 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.

2.1.1 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.

2.2 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 3.1.8 *File Parameter Values*).

3 Input File Format

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

3.1 Text Input Files

3.1.1 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.

3.1.2 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.

3.1.3 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
```

3.1.4 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"
```

3.1.5 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
```

3.1.6 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.

3.1.7 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"
```

3.1.8 File Parameter Values

File parameters accept text values (see section 3.1.7) 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".

3.1.8.1 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"
```

3.1.9 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".

3.1.10 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.

3.1.11 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 parameter 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.

3.2 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. Currently, LANDIS-II only supports one format: ERDAS 7.4, the format used by earlier versions of the LANDIS model. In the future, LANDIS-II will support other file formats.

3.2.1 ERDAS 7.4 (*.gis)

Both 8-bit and 16-bit maps are supported.

4 Scenario Input File

This file describes a single model scenario.

4.1 Example File

```
LandisData  Scenario

Duration    300

Species     species.txt

Ecoregions      ./ecoregions.txt
EcoregionsMap   ./ecoregions.gis

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

InitialCommunities      "./initial-communities.txt"
InitialCommunitiesMap   "initial communities.gis"

>> 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
```

4.2 LandisData

This parameter's value must be "Scenario".

4.3 Duration

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

4.4 Species

This parameter is the file which contains the species' parameters (see chapter 5).

4.5 Ecoregions

This parameter is the file which contains the definitions of the ecoregions on the landscape (see chapter 6).

4.6 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 6).

4.7 CellLength

This parameter is the length of a cell's edge in the ecoregions map (see section 4.6). 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.

4.8 InitialCommunities

This parameter is the file with the definitions of the initial communities at the active sites on the landscape (see chapter 7).

4.9 InitialCommunitiesMap

This parameter is the input map indicating the initial communities at the active sites on the landscape. Each cell value for an active site on the landscape must be one of the map codes listed in the initial communities input file (see chapter 7).

4.10 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.

4.10.1 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).

4.10.2 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.

4.10.3 Succession Extension

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

4.10.4 Disturbance Extensions

A scenario may have zero or more disturbance extensions.

4.10.5 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).

4.10.6 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.

4.11 RandomNumberSeed

This parameter is the integer value that the program uses to initialize its internal random-number generator. Value: integer $\neq 0$. 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.

5 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.

5.1 Example File

See Figure 1.

Figure 1 – Example of species input file.

```
LandisData Species

>>
>> Name      Longevity  Sexual      Shade  Fire  Seed Dispersal Dist  Vegetative  Sprout Age  Post-Fire
>> -----  -----  -----  -----  -----  -----  -----  -----  -----  -----
>> 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
```

5.2 LandisData

This parameter's value must be "Species".

5.3 Table Fields

5.3.1 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.

5.3.2 Longevity

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

5.3.3 Sexual maturity

This parameter is the age at which the species matures sexually.
Value: $0 < \text{integer} \leq \text{Longevity}$. Units: years.

5.3.4 Shade tolerance

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

5.3.5 Fire tolerance

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

5.3.6 Effective seeding distance

This parameter is the species' effective distance for dispersing seeds.
Value: $0 < \text{integer} \leq \text{Max Seed Dist}$. 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.

5.3.7 Maximum seeding distance

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

5.3.8 Vegetative reproduction probability

This parameter is the probability that the species resprouts (vegetation reproduction). Value: $0.0 \leq \text{number} \leq 1.0$

5.3.9 Minimum resprouting age

This parameter is the minimum age required for the species to resprout. Value: $0 < \text{integer} \leq \text{Longevity}$. Units: years.

5.3.10 Maximum resprouting age

This parameter is the maximum age required for the species to resprout. Value: $\text{Min Sprout Age} \leq \text{integer} \leq \text{Longevity}$. Units: years.

5.3.11 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.

6 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.

6.1 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"
```

6.2 LandisData

This parameter's value must be "Ecoregions".

6.3 Table Fields

6.3.1 Active

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

6.3.2 Map Code

This parameter is the code used for the ecoregion in the input map (see section 4.6). Value: $0 \leq \text{integer} \leq 65,535$. 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.

6.3.3 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.

6.3.4 Ecoregion Description

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

7 Initial Communities Input File

This file contains the definitions of the initial community classes. Each active site on the landscape is assigned to an initial community class. The class specifies the tree species that are present along with the particular age classes that are present for each of those species.

7.1 Example File

```
LandisData    "Initial Communities"

>>Old jackpine oak
MapCode  7
    acerrubr 30
    pinubank 80 90
    pinuresi 110 140
    querelli 40 120 240

>> young jackpine oak
MapCode  0
    pinubank 30 50
    querelli 10 40 70

>> young aspen
MapCode  2
    poputrem 10 20

>> old maple hardwoods
MapCode 55
    abiebals 10 60 120
    acerrubr 90 120
    acersacc 20 50 150 200
    betualle 40 140 200
    fraxamer 10 100 130 180
    piceglau 180
    querrubr 100 160 180
    thujocci 200 240 260
    tiliamer 20 80 110 150
    tsugcana 30 80 120 220 320 340

>> old pine - spruce - fir
MapCode  6
    abiebals 10 50 80
    piceglau 100 140 180 200 220
    pinuresi 140 160 180
    pinustro 200 280 350
```

7.2 LandisData

This parameter's value must be "Initial Communities".

7.3 Initial Community Class Definitions

Each class has an associated map code and a list of species present at sites in the class.

7.3.1 MapCode

This parameter is the code used for the class in the input map (see section 4.9). Value: $0 \leq \text{integer} \leq 65,535$. Each class' map code must be unique. Map codes do not have to appear in any order, and do not need to be consecutive.

7.3.2 Species Present

A list of species present at the class' sites comes after the map code. Each species is listed on a separate data line.

```
species age age age ...
```

The species name comes first, followed by one or more ages. The name and ages are separated by whitespace. An age is an integer and must be between 1 and the species' Longevity parameter (see section 5.3.2). The ages do not have to appear in any order.

```
acersacc 10 5 21 60 100
```

The list may be empty, which will result in the sites in the class being initialized with no species cohorts.

7.3.3 Grouping Species Ages into Cohorts

The list of ages for each species is grouped into cohorts based on the succession extension's timestep. This timestep determines the size of each cohort. For example, if the timestep is 20, then the cohorts are ages 1 to 20, 21 to 40, 41 to 60, etc.

Suppose an initial community class has this species in its list:

```
acersacc 10 25 30 40 183 200
```

If the succession timestep is 10, then the cohorts for this species initially at each site in this class will be:

```
acersacc 10 20 30 40 190 200
```

If the succession timestep is 20, then the cohorts for this species initially at each site in this class will be:

acersacc 20 40 200