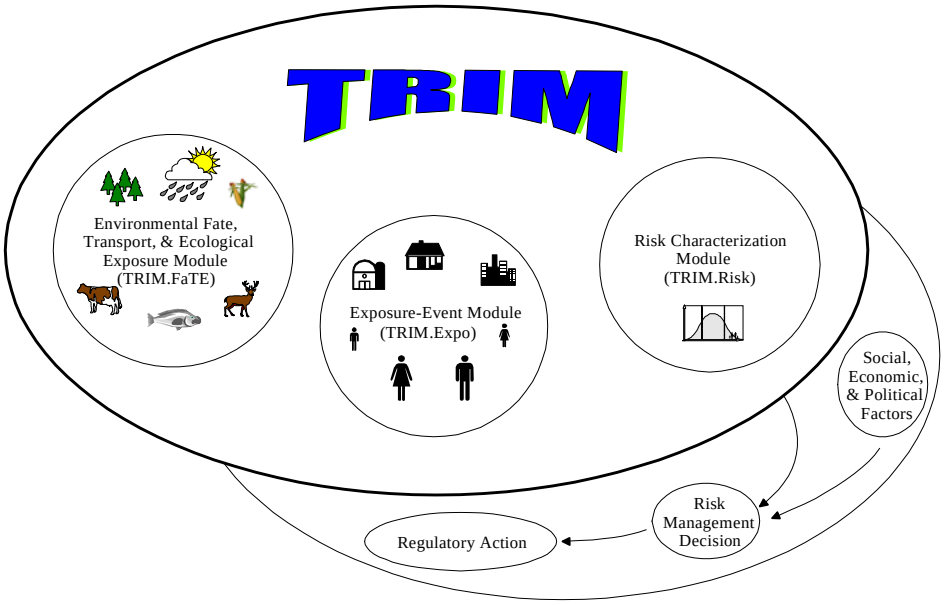
TRIM
Total Risk Integrated Methodology

TRIM.Risk User's Guide
Volume III: Ecological Risk Assessment



U.S. Environmental Protection Agency
Office of Air Quality Planning and Standards
Research Triangle Park, NC 27711

December 2005

TABLE OF CONTENTS

List of Exhibits . . . . . **ii**

List of Appendices . . . . . **iii**

1. Introduction **1-1**
1.1 How TRIM.Risk Is Used 1-1
1.1.1 Problem Formulation Phase 1-1
1.1.2 Exposure and Effects Analysis Phase 1-1
1.1.3 Risk Characterization Phase 1-3
1.2 MIMS Projects 1-3
1.3 How to Use this Document 1-4

2. Risk Outputs **2-1**

3. Ecological Risk Assessment Scenarios **3-1**
3.1 TRIM.FaTE with TRIM.Risk Scenario 3-2
3.2 TRIM.Risk Scenario 3-8

4 Components of the Ecological Risk Assessment Scenarios **4-1**
4.1 TRIM.FaTE 4-1
4.1.1 TRIM.FaTE Setup 4-1
4.2 $TRIM.RISK_{Eco}$ 4-3
4.2.1 $TRIM.Risk_{Eco}$ Setup 4-4
4.3 DAVE 4-16

5. References **5-1**

LIST OF EXHIBITS

Exhibit 1-1. TRIM in the Context of EPA's Framework for Ecological Risk Assessment . . 1-2
Exhibit 2-1. Examples of Ecological Toxicity Metrics 2-3
Exhibit 3-1. Currently Available Scenarios for "Ecological risk assessment with TRIM.FaTE" Project 3-1
Exhibit 3-2. "Ecological risk assessment with TRIM.FaTE" Scenario Decision Tree 3-2
Exhibit 3-3. Parameters on "Settings" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario 3-4
Exhibit 3-4. "Settings" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario 3-4
Exhibit 3-5. Parameters on "FaTE" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario 3-5
Exhibit 3-6. "FaTE" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario 3-6
Exhibit 3-7. Parameters on "Settings" Tab of "TRIM.Risk" Scenario 3-9
Exhibit 3-8. "Settings" Tab of "TRIM.Risk" Scenario 3-9
Exhibit 4-1. TRIM.FaTE Outputs for Use in $TRIM.Risk_{Eco}$ 4-2
Exhibit 4-2. Time Point Types 4-14

LIST OF APPENDICES

A. Data Flow Diagrams

B. Scenario and Module Instance Parameters

C. MySQL Database Schema

D. Troubleshooting

E. TRIM.FaTE Library Settings Required for TRIM.$Risk_{Eco}$

F. Documentation of Ecological Risk Assessment Component Executables

F.1 TRIM.FaTE
F.2 TRIM.$Risk_{Eco}$
F.3 DAVE

[This page intentionally left blank.]

1. Introduction

The TRIM.Risk User's Guide is part of EPA's series of documentation for the overall Total Risk Integrated Methodology (TRIM) modeling system. Volume III of the TRIM.Risk User's Guide describes how multimedia ecological risk assessments can be performed with TRIM.Risk to generate ecological risk values in the form of hazard quotients. In general, the information presented in this document is based on EPA's experience with the test applications of TRIM.Risk that have been implemented to date. The information included in this document may be subject to change as additional model applications are completed and the TRIM.Risk model evolves.

1.1 How TRIM.Risk Is Used

TRIM.Risk calculates ecological risk results by combining ecological exposure estimates (e.g., TRIM.FaTE results) with ecological toxicity values for user-selected receptors. This section describes the general procedure by which TRIM.Risk can be used to generate ecological risk values and what types of inputs and decisions are required for an assessment. More details on how the specific components of the model should be run and the types of ecological risk values that can be calculated are presented in later chapters in this volume.

Exhibit 1-1 illustrates the functions of the ecological risk assessment-related TRIM components and their associated output databases in the context of EPA's framework for ecological risk assessment (EPA 1992). The phases of this process, as they relate to the ecological risk assessment scenarios, are described in more detail below.

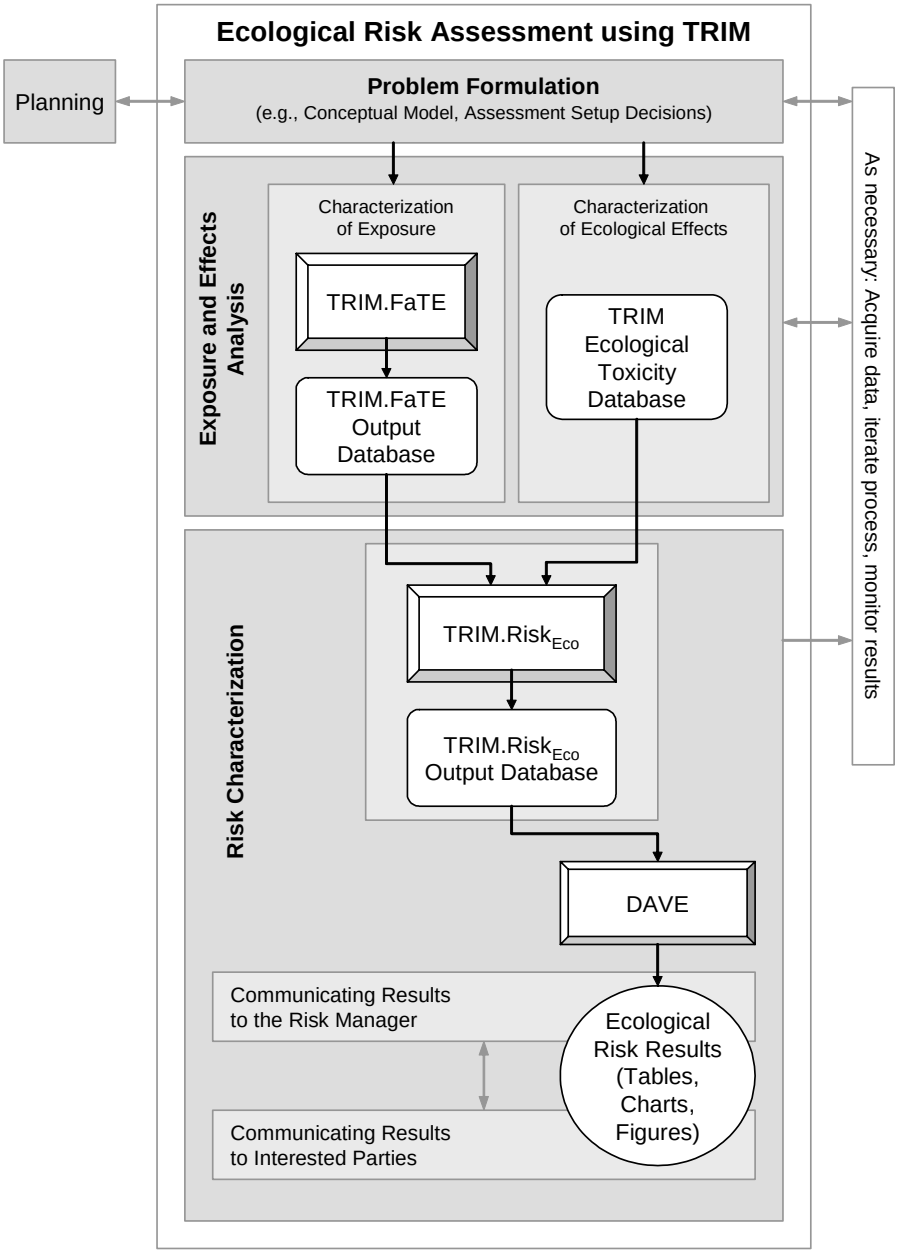
1.1.1 Problem Formulation Phase

During this phase of an ecological risk assessment, the risk assessor identifies the measures of exposure and effect needed for the risk analysis and characterization phases. In the context of an ecological risk assessment, this includes making decisions such as determining the location, the receptors, the time-frame, the chemicals, and the endpoints of concern. These decisions need to be addressed before progressing to the exposure, effects, and risk characterization phases (see the EPA Guidelines for Ecological Risk Assessment (EPA 1998)).

1.1.2 Exposure and Effects Analysis Phase

During this phase, the risk assessor develops exposure estimates (e.g., via TRIM.FaTE modeling) and identifies corresponding ecological toxicity values to appropriately represent the measures of exposure and effect that were selected during the problem formulation phase. For complete details on ecological exposure modeling with TRIM.FaTE, see the TRIM.FaTE documentation. There are multiple documents describing the application of TRIM.FaTE on the EPA website: http://www.epa.gov/ttn/fera/trim_fate.html. In addition, the TRIM.FaTE User's Guide provides useful details on setting up a TRIM.FaTE simulation.

Exhibit 1-1. TRIM in the Context of EPA's Framework for Ecological Risk Assessment



TRIM.FaTE is a multimedia fate and transport model that uses a compartmental mass balance approach to track the movement and transformation of pollutants over time, through a user-defined, bounded system. During an ecological exposure modeling exercise, TRIM.FaTE can be used to estimate pollutant concentrations in environmental media and also to estimate pollutant intakes for terrestrial wildlife. The spatial extent of the modeling region, the media (e.g., surface water, soil, groundwater), and the biota (e.g., terrestrial animals, fish) that are included in the TRIM.FaTE model setup and run are determined by the needs of the assessment. The biota in a TRIM.FaTE run are exposed to the chemical in the environment by various user-specified links. These can include exposure from food, water, air, and soil. The duration of the exposure modeled is also determined by the assessment. Both dynamic and steady-state

exposure scenarios can be performed with TRIM.FaTE. In a dynamic scenario, the pollutant emission can vary with time, or be constant (like all TRIM.FaTE inputs). The duration of the exposure can be set to any length. A steady-state assessment predicts the pollutant distribution (in terms of mass, moles and concentration in each compartment) at the point when – given inputs and processes that are constant with time – the pollutant transfers among compartments do not change with time.

Three different measures of exposure can be generated from TRIM.FaTE and used as exposure estimates for an ecological receptor: biota oral intake[1], biota compartment concentration, and abiotic compartment concentration. The user is responsible for identifying the pertinent and appropriate ecological effect values for evaluation of the exposure estimates. The TRIM ecological toxicity database provides examples of ecological effect values to illustrate to users the types of values they will need to consider in identifying those that they will use for their assessment. For additional details about the TRIM ecological toxicity database, refer to the documentation for the database available from the TRIM.Risk page of the EPA FERA website: http://www.epa.gov/ttn/fera.

1.1.3 Risk Characterization Phase

During this phase, TRIM.Risk$_{Eco}$ is used to calculate ecological hazard quotient (HQ) values for the ecological receptors of interest. The TRIM.Risk$_{Eco}$ graphical user interface (GUI) is designed to guide the user in matching ecological exposure estimates with ecological effect values of the same type (e.g., media concentration, wildlife oral intake, etc), and in focusing the analysis with regard to the locations, time points of interest, and ecological effects of interest. See Chapter 2 for more information on the types of risk outputs that can be generated with TRIM.Risk$_{Eco}$, and see the documentation of the ecological toxicity database (http://www.epa.gov/ttn/fera) for more information on the types of toxicity values available in the database. The ecological risk estimates generated by TRIM.Risk$_{Eco}$ can be processed using the Data Analysis and Visualization Engine (DAVE) to generate tables and charts for communicating risk assessment results (see Volume I of the TRIM.Risk User's Guide for more information on DAVE).

1.2 MIMS Projects

TRIM.Risk uses the MIMS framework to manage inputs, outputs, and the running of multiple models and programs through the use of projects and scenarios (see Volume I of the User's Guide for more information on using MIMS for TRIM). An ecological risk assessment project designed to address anticipated TRIM user needs is available via the TRIM installation program, accessed from the EPA FERA website (http://www.epa.gov/ttn/fera). Users can perform ecological risk assessments using TRIM.Risk with the available project "Ecological risk assessment with TRIM.FaTE," or they may create alternative projects. This TRIM.Risk project contains two scenarios, one of which includes use of the TRIM.FaTE ecological exposure model and the second which relies on an existing database of ecological exposure estimates (e.g., from

[1] Biota oral intake values can only be used if generated from a dynamic TRIM.FaTE run. Intake values (in mg/kg-day) are mass fluxes which will be zero if they are generated from a steady-state run.

a previous TRIM.FaTE simulation). Complete descriptions of the available scenarios are provided in Chapter 3.

1.3 How to Use this Document

This volume provides instructions for using TRIM.Risk to perform an ecological risk assessment. Chapter 2 describes the different ecological risk outputs that are available from TRIM.Risk. Chapter 3 describes the steps involved in setting up and running the currently available scenarios. Chapter 4 describes each of the components of the scenarios associated with this TRIM.Risk project, including detailed instructions for configuring each component (TRIM.FaTE, TRIM.Risk$_{Eco}$, and DAVE). The appendices provide more detailed information on the databases, parameters, and components, as well as troubleshooting tips.

New users will want to review Volume I of this manual (Introduction to TRIM.Risk) for an introduction to TRIM.Risk and its associated required software, as well as instructions on how to perform basic operations in MIMS (the model framework for TRIM.Risk), MySQL (the relational-database used to communicate data between components of TRIM.Risk), and DAVE (the data analysis software provided with TRIM.Risk). Chapters 1 and 2 of this volume contain general information on performing an ecological risk assessment with TRIM.Risk. After becoming familiar with the basics of TRIM.Risk, users can perform an assessment with TRIM.Risk by following the steps provided for each scenario in Chapter 3, referencing the documentation of each of the model components in Chapter 4, and possibly referencing the additional information provided in the appendix.

2. Risk Outputs

In the context of an ecological risk assessment, TRIM.Risk combines ecological exposure estimates with toxicity values to derive hazard quotients for ecological receptors. With the help of the TRIM.Risk$_{Eco}$ GUI, a user can match up estimates of ecological exposure (e.g., from a TRIM.FaTE simulation) with toxicity values for ecological receptors and endpoints of interest. TRIM.FaTE ecological exposure estimates are one of three types:[2]

- Abiotic media concentration (e.g., soil or surface water concentration);
- Ecological receptor body burden (biotic compartment concentration); and
- Wildlife receptor oral intake (e.g., mg[pollutant]/kg[body weight]-day).

TRIM.Risk calculates hazard quotients by matching these exposure estimates with appropriate toxicity values for the receptors (e.g., aquatic life, white-tailed deer, birds) and effects (e.g., reproductive toxicity, growth) of interest for the application. The type of exposure estimates used will depend on the needs and scope of the assessment, the possible output from the exposure model, and the available toxicity values.[3] Hazard quotients can be generated for user-specified receptors, toxicity metrics (e.g., LOAEL, MTL, reference value), and effects (e.g., reproductive success, growth, all), as well as different time periods (steady-state or dynamic) or exposure durations (e.g., annual average, average of specific time period, instantaneous within specific time period). These options are described briefly below, and their selection via the TRIM.Risk$_{Eco}$ GUI is illustrated in Section 4.2. More information is available from the ecological toxicity database documentation, available from the EPA FERA website (http://www.epa.gov/ttn/fera).

Receptors

The receptors for which TRIM.Risk derives hazard quotients can be general (e.g., soil community) or specific (e.g., white-tailed deer), depending on the needs of the assessment, the associated design of the TRIM.FaTE scenario from which exposure estimates are provided, and the availability of various toxicity values. For example, if an assessment is designed to estimate the effects of a chemical exposure on small mammals in an area, the user could set up a TRIM.FaTE scenario to output the chemical oral intake rate for rabbits and mice within the study area. TRIM.Risk could then be used to calculate ecological HQ values based on those modeled intake rates and user-selected toxicity values. The toxicity values could be for specific animal receptor species (e.g., eastern cottontail rabbits and deer mice) and/or for general receptor groups (e.g., mammals). Another assessment may be designed to determine the impact of a chemical contaminant to ecological receptors in a surface water body. In that case, a TRIM.FaTE scenario could be set up to generate abiotic media concentrations (e.g., chemical concentration in surface

[2]Note that users can use exposure estimates from sources other than TRIM.FaTE, but for simplicity, we will refer to TRIM.FaTE outputs in this document.

[3]Examples of ecological toxicity values are provided for a subset of hazardous air pollutants in the ecological toxicity database available for download from the TRIM.Risk Download page of the EPA FERA website (http://www.epa.gov/ttn/fera). Users will need to review the database and augment or revise, as needed for their application.

water and sediment). TRIM.Risk could then be used to generate HQ values for receptors (e.g., fish, the aquatic community, aquatic plants, daphnids, or frogs) from the chemical concentration in the surface water and toxicity values based on surface water concentrations.

Type of Exposure Estimates

The types of exposure estimates considered in the TRIM.Risk analysis will depend on the ecological exposure model simulation and any subsequent data manipulation. Using TRIM.FaTE, these estimates may be instantaneous, averaged (e.g., daily or annual averages), or steady-state estimates. Thus, depending on the needs of the study, HQ values can be generated from either multiple (average or instantaneous) exposure estimates at different time points, or single long-term/steady-state exposure estimates. When using the TRIM.Risk$_{Eco}$ GUI to set up the risk calculations, the user will need to assure consistency between the temporal aspects of the exposure estimates and the toxicity metrics selected for use in deriving HQs (e.g., match short term exposure estimates with short term (e.g., acute) toxicity values and long-term or steady-state exposure estimates with chronic toxicity values).

Toxicity Values

The toxicity values with which TRIM.Risk derives hazard quotients are specified by the user, according to the needs of the assessment as well as the availability of toxicity values. A preliminary ecological toxicity database is provided on the EPA FERA website (http://www.epa.gov/ttn/fera) with ecological toxicity values for some hazardous air pollutants. Users are encouraged to review this MySQL database when designing an assessment and modify it as appropriate for the needs of their assessments (e.g., for different chemicals, receptors, toxicity metrics). See Section 3.2.2 of Volume I for instructions on adding data to a MySQL database.

Examples of the types of toxicity metrics available in the ecological toxicity database as of September 2005 are described in Exhibit 2-1. Users are referred to the database documentation for complete information on the toxicity values in the current database. To assist users in specifying toxicity values to use in TRIM.Risk, the TRIM.Risk$_{Eco}$ GUI presents information associated with each value (e.g., receptor match, chemical, exposure measure, toxicity metric, effect, toxicity value, units). More detailed information for each value is available in the ecological toxicity database and may be viewed by using the MySQL Browser (see Volume I, Section 3.2 for more information on the MySQL Browser).

Exhibit 2-1. Examples of Ecological Toxicity Metrics

Abbreviation	Description	Abbreviation	Description
ASV	Acute Screening Value	LOEL	Lowest Observed Effects Level
CCC	Criterion Continuous Concentration	MATC	Maximum Allowable Toxicant Concentration (geometric mean of NOAEL and LOAEL)
C-CSCL	Community Based CSCL	MATL	Maximum Allowable Toxicant Level (geometric mean of NOAEL and LOAEL)
CMC	Criterion Maximum Concentration	NEC	High No Effects Concentration
CSV	Chronic Screening Value	NOAEL	No Adverse Effects Level
CV	Chronic Value	PEC	Probable Effect Concentration
EC-20	Effects Concentration for 20% of the test population	PEL	Probable Effects Level
EC-50	Effects Concentration for 50% of the test population	SAV	Secondary Acute Value
ER-L	Effects Range - Low: The low end of values showing an effect	SCV	Secondary Chronic Value; Tier II Value
FAV	Final Acute Value	SQB	Sediment Quality Benchmark
FCV	Final Chronic Value	SQC	Sediment Quality Criteria
geomean	Geomean of LC-50	SSL	Soil Screening Level
LC-50	Lethal Concentration for 50% of the test population	TEC	Threshold Effects Concentration
LCV	Lowest Chronic Value	TEL	Threshold Effects Level
LD-50	Lethal Dose for 50% of the test population	TRV	Toxicity Reference Value
LOAEL	Lowest Adverse Effects Level	TRV-High	The mid-point of a variety of adverse effect levels that would be expected to produce an adverse effect to a population of organisms
LOEC	Lowest Observed Effects Concentration	TRV-Low	Lowest credible no adverse effect level (NOAEL) for a population of organisms

[This page intentionally left blank.]

3. Ecological Risk Assessment Scenarios

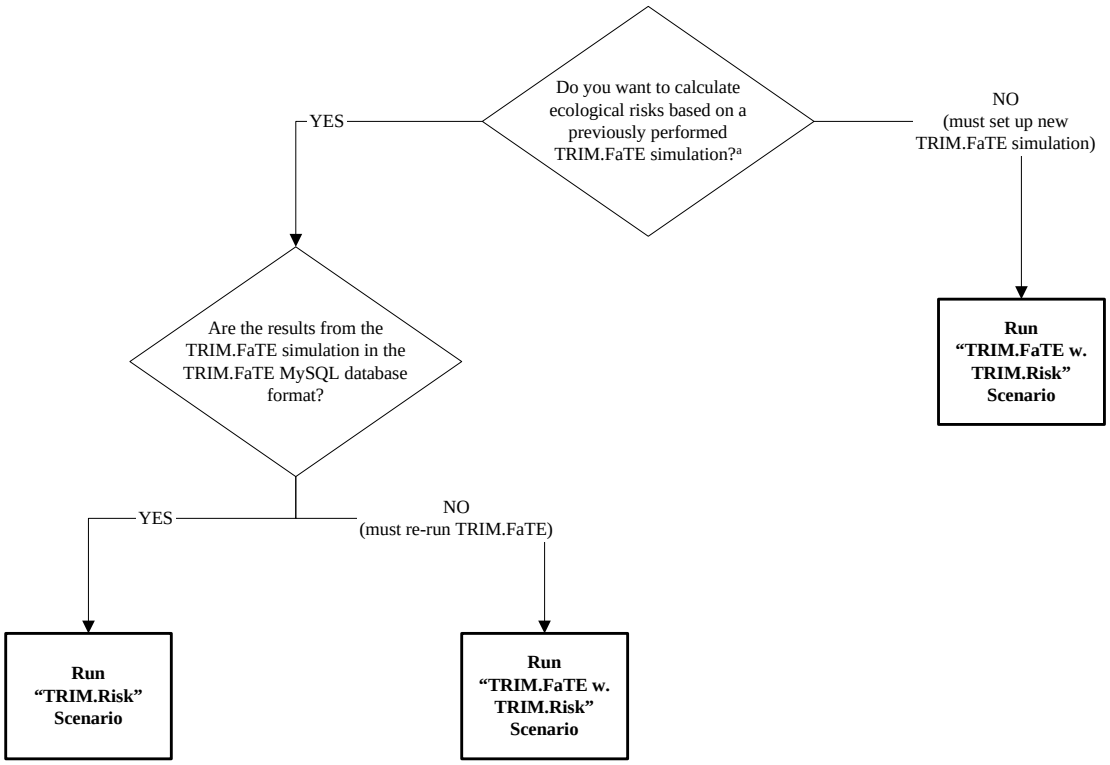
Users can perform ecological risk assessments using TRIM.Risk with either of the two available MIMS scenarios in the "Ecological risk assessment with TRIM.FaTE" MIMS project, or by creating a new scenario (see Volume I, Section 3.1.10). The two existing MIMS scenarios differ only in the way ecological exposure estimates are generated. In the "TRIM.FaTE w. TRIM.Risk" scenario, ecological exposure estimates are generated by running TRIM.FaTE within the scenario, with the outputs being automatically passed on to the TRIM.Risk simulation. In the "TRIM.Risk" scenario, ecological exposure estimates from a previous exposure simulation must be provided as inputs in a TRIM.FaTE-formatted MySQL database. Exhibit 3-1 lists both of the available scenarios and specifies which components they each include. A decision tree is provided in Exhibit 3-2 to assist users in determining which scenario is appropriate for their needs. The appendices to this volume contain additional information on these scenarios, including the sequence in which components must be run (Appendix A), scenario and module instance parameters for each scenario (Appendix B), and the MySQL database schema (Appendix C). Information on troubleshooting is included in Appendix D.

The remainder of this section describes the steps involved in using each of these MIMS scenarios. Section 3.1 describes the steps involved in setting up and running the "TRIM.FaTE w. TRIM.Risk" MIMS scenario and Section 3.2 describes the steps involved in setting up and running the "TRIM.Risk" MIMS scenario.

Exhibit 3-1. Currently Available Scenarios for "Ecological risk assessment with TRIM.FaTE" Project

Scenario	Description	Components		
		TRIM.FaTE	$TRIM.Risk_{Eco}$	DAVE
TRIM.FaTE w. TRIM.Risk	Used to perform an ecological risk assessment for one or more chemicals using TRIM.FaTE and $TRIM.Risk_{Eco}$.	✓	✓	✓
TRIM.Risk	Used to perform an ecological risk assessment for one or more chemicals using $TRIM.Risk_{Eco}$ and an existing TRIM.FaTE output database.		✓	✓

Exhibit 3-2. "Ecological risk assessment with TRIM.FaTE" Scenario Decision Tree



[a] An ecological exposure model other than TRIM.FaTE can be used to generate exposure estimates as long as the exposure model results are in the TRIM.FaTE MySQL database format (see Appendix C). The "TRIM.Risk" scenario would then be the scenario to run.

In order to run these scenarios, users must have installed the required software (see Volume I, Chapter 2) and imported the "Ecological risk assessment with TRIM.FaTE" project into MIMS (see Volume I, Section 3.1.2).

3.1 TRIM.FaTE with TRIM.Risk Scenario

This section describes the steps to run the "TRIM.FaTE w. TRIM.Risk" scenario. This scenario is used to generate ecological exposure estimates with TRIM.FaTE and ecological risk estimates with $TRIM.Risk_{Eco}$. Each step provides an overview of how the user carries out the step and references to specific component sections and other relevant user's guides for details of how to perform the step.

Step 1 Create Multimedia Modeling Scenario in TRIM.FaTE and Set Property Values

Create a multimedia scenario in TRIM.FaTE that appropriately represents the conceptual model for the risk assessment. Set the TRIM.FaTE properties that are required to produce output in the format needed for TRIM.Risk$_{Eco}$. Users should review the TRIM.FaTE documentation cited below **before** starting to configure their TRIM.FaTE scenario. After setting up the TRIM.FaTE scenario, the user should (1) verify the scenario using the TRIM.FaTE "Verify Scenario" option under the "Run" menu, (2) save the project, (3) note the project file name and location, and (4) quit TRIM.FaTE. **Do not run the TRIM.FaTE scenario at this point**.

Refer to: TRIM.FaTE User's Guide

Section 4.1, TRIM.FaTE (for a list of properties required for TRIM.Risk$_{Eco}$)

Step 2 Open the "TRIM.FaTE w. TRIM.Risk" Scenario in MIMS

Open TRIM.Risk by selecting "TRIM" from the Start menu, Programs list, TRIM menu, or by double-clicking on "runtrim.bat" in Windows Explorer. Open the "Ecological risk assessment with TRIM.FaTE" project in MIMS, and then open the "TRIM.FaTE w. TRIM.Risk" scenario.

Refer to: Volume I, Sections 3.1.4, Opening a MIMS Project

Volume I, Sections 3.1.6, Opening, Duplicating, Renaming, or Deleting a MIMS Scenario

Step 3 Complete the Input Panels

Follow Steps 3a and 3b (below) to set the necessary parameter settings via the scenario "Input Panels." The "Input Panels" present those scenario and module instance parameters that most commonly need attention from users (see Exhibits 3-3 through 3-6 below). Note that some of these parameters will already be set by the TRIM installation program. Users should use caution when making changes to any preset parameters.

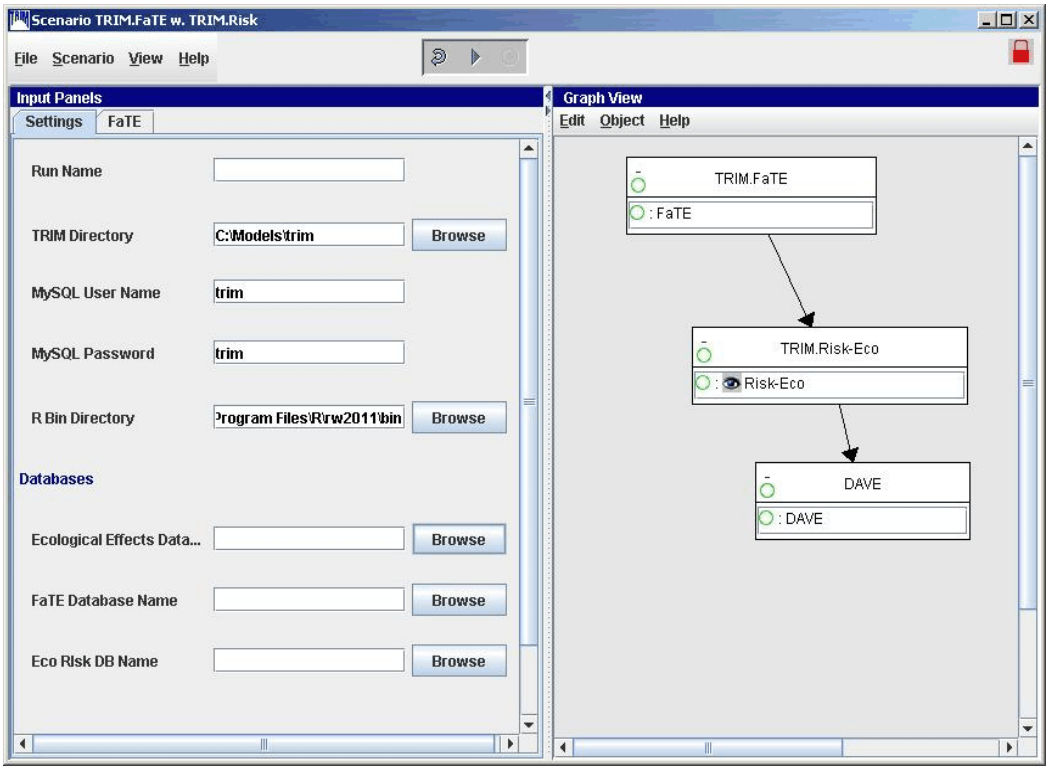
Step 3a Set Parameters on "Settings" Tab

Set the parameters on the "Settings" tab of the scenario "Input Panels." The parameters are listed in Exhibit 3-3 below, shown in Exhibit 3-4, and are set by either typing in a value or clicking on the "Browse" button next to the parameter field and browsing for the correct file or directory.

Exhibit 3-3. Parameters on "Settings" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario

Parameter	Default Value	Set via Installer?	Description
Run Name	–	No	Name of the unique TRIM.Risk run.
TRIM Directory	–	Yes	TRIM installation directory (i.e., the directory into which the TRIM.Risk installation packages were installed).
MySQL User Name	–	Yes	User name for access to the MySQL databases created or used for the TRIM.Risk projects.
MySQL Password	–	Yes	Password for access to the MySQL databases created or used for the TRIM.Risk projects.
R Bin Directory	–	Yes	Location of the bin directory for the R program.
Ecological Effects Database	–	No	Name of database with ecological toxicity reference values.
FaTE Database Name	–	No	Name of output database from TRIM.FaTE simulation.
Eco Risk DB Name	–	No	Name of the output database from TRIM.Risk$_{Eco}$.

Exhibit 3-4. "Settings" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario



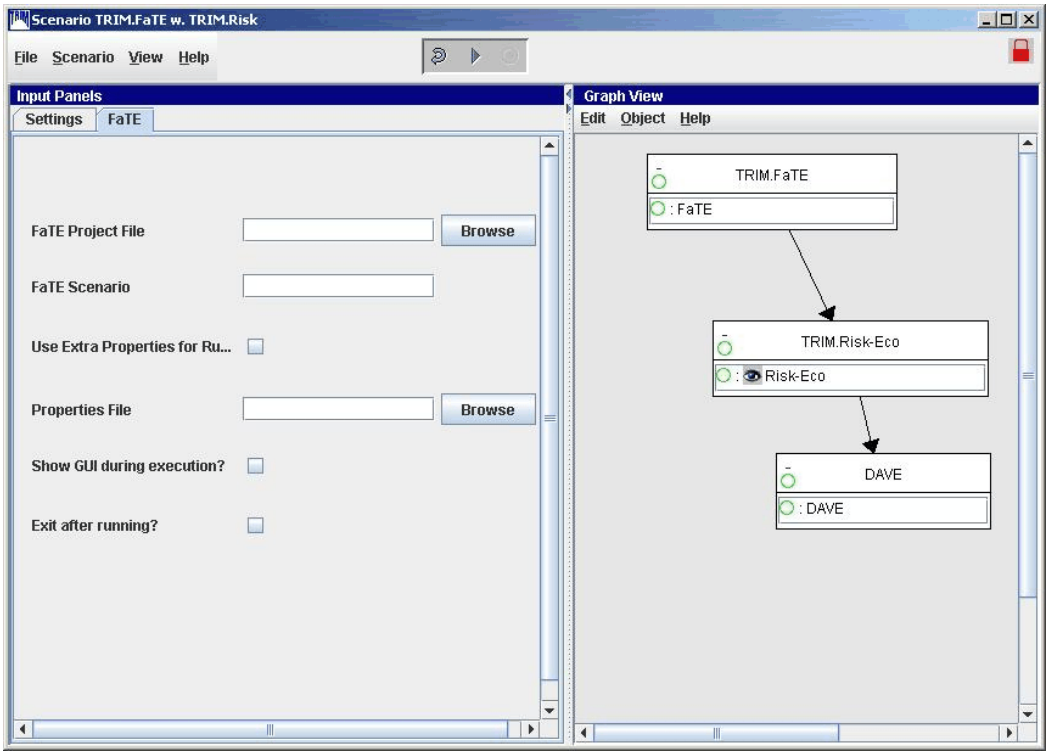
Step 3b **Set Parameters on "FaTE" Tab**

Set the parameters on the "FaTE" tab of the scenario "Input Panels." The parameters are listed in Exhibit 3-5 below, shown in Exhibit 3-6, and are set by typing in a value, clicking on the "Browse" button next to the parameter field and browsing for the correct file or directory, or clicking on a check box.

Exhibit 3-5. Parameters on "FaTE" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario

Parameter	Default Value	Set via Installer?	Description
FaTE Project File	–	No	Name of TRIM.FaTE project file.
FaTE Scenario	–	No	Name of TRIM.FaTE scenario.
Use Extra Properties for Run?	False	No	Indicates whether a properties import file will be used to modify the TRIM.FaTE scenario before running it. If true, the properties file name must be specified by the Properties File parameter. Currently, the only TRIM.FaTE property that can be modified using this file is the outputDatabaseName.
Properties File	–	No	Properties import file for the TRIM.FaTE model run. This file could be used to specify a revised name for the TRIM.FaTE property outputDatabaseName. The same name must be specified in the field "FaTE Database Name." Currently, a file name must be entered in this field even if "Use Extra Properties for Run?" is false and you do not intend to replace the database name. Enter a file name that is not the same as another file in your "Default Directory" for this scenario, and a file will be created by the scenario.
Show GUI during execution?	False	No	Indicates whether the TRIM.FaTE GUI will be visible while the module instance is running.
Exit after running?	False	No	Check box which indicates whether the TRIM.FaTE window will close automatically after the module instance run is complete.

Exhibit 3-6. "FaTE" Tab of "TRIM.FaTE w. TRIM.Risk" Scenario



Step 4 Set Any Additional Parameters

In most cases, users will not be required to set any parameters beyond those in the "Input Panels." However, in some applications users may find it necessary to view or edit additional scenario or module instance parameters. To access the scenario parameters, select "Edit Parameter Values" from the "Scenario" menu of the scenario window. Access the module instance parameters by double-clicking on a module instance name in the "Graph View" pane, which opens the module instance window. On the "Parameters" tab, set values by typing in the "Value" column, clicking on a check box, or selecting the "Browse" button to locate the file. Note that this step may not be required for some applications. No user interaction is required on the "Module" tab.

Refer to: Volume I, Section 3.1.7, Specifying MIMS Scenario Parameter Values

Volume I, Section 3.1.8, Specifying MIMS Module Instance Parameter Values

Appendix B, Scenario and Module Instance Parameters

Step 5 Start the Scenario

After setting all parameter values, the scenario is ready to run.[4] Select "Execute Ready Modules" from the "Scenario" menu or click on the "play" button ( ) at the top of the scenario window. The TRIM.FaTE GUI will appear on the screen as the TRIM.FaTE component is running. After the TRIM.FaTE simulation is completed, the TRIM.Risk$_{Eco}$ GUI will automatically open.

Refer to: Volume I, Section 3.1.9, Running a MIMS Scenario

Step 6 Set Inputs and Run TRIM.Risk$_{Eco}$

Make the appropriate selections in the TRIM.Risk$_{Eco}$ GUI based on the needs of the risk assessment. After configuring the TRIM.Risk$_{Eco}$ GUI and clicking "Save" and "Exit," the risk calculations will be performed.

Refer to: Section 4.2, TRIM.Risk$_{Eco}$

Chapter 2, Risk Outputs

Step 7 Run DAVE

After the TRIM.Risk$_{Eco}$ component completes, DAVE will automatically open and can be used to analyze or export the ecological risk estimates which are contained in the newly created ecological risk database. Use DAVE to generate tables and charts of ecological risk estimates or to create delimited text exports of the data that can be imported into spreadsheet or data analysis software. Alternatively, risk estimates can be retrieved directly from the database using MySQL queries.

Refer to: Section 4.3, DAVE

Volume I, Section 3.3, Basic Operations with DAVE

DAVE User's Guide

---

[4] Note that if this scenario has been run previously, and/or if selections have been made in the TRIM.Risk$_{Eco}$ GUI in this MIMS scenario previously, the scenario must be reset and the GUI must be reloaded before the run can begin. Select "Reset All" from the "Scenario" drop-down menu to return all processes to a ready-to-execute state (resetting will ***not*** clear any scenario or module instance parameters that have been set), and right-click on the TRIM.Risk$_{Eco}$ module instance and select "Reload Custom GUI," to clear previous selections from the GUI.

3.2 TRIM.Risk Scenario

This section describes the steps to run the scenario involving TRIM.Risk alone. This scenario is used to generate ecological risk estimates from an already existing ecological exposure database. Each step provides a basic overview of how the user carries out the step and references to specific component sections and other user's guides. Several of the steps are similar to steps in the "TRIM.FaTE w. TRIM.Risk" MIMS scenario described in the section above.

Step 1 Run a Multimedia Modeling Scenario in TRIM.FaTE and Generate Output in a MySQL Database (or confirm applicability of existing ecological exposure database)

Run a TRIM.FaTE simulation with the appropriate property values to produce an ecological exposure MySQL database. Alternatively, confirm that a database from a previous TRIM.FaTE run was created with the appropriate properties. Output from other ecological exposure models can also be used as input for $\text{TRIM.Risk}_{\text{Eco}}$; however, these results must be provided in the TRIM.FaTE MySQL output format. After confirming that the exposure data are in the appropriate format, users can proceed to Step 2.

Refer to: Section 4.1, TRIM.FATE

Appendix C, MySQL Database Schema

Step 2 Open the "TRIM.Risk" Scenario in MIMS

Open TRIM.Risk by selecting "TRIM" from the Start menu, Programs list, TRIM menu, or by double-clicking on "runtrim.bat" in Windows Explorer. Open the "Ecological risk assessment with TRIM.FaTE" project in MIMS, and then open the "TRIM.Risk" scenario.

Refer to: Volume I, Sections 3.1.4, Opening a MIMS Project

Volume I, Sections 3.1.6, Opening, Duplicating, Renaming, or Deleting a MIMS Scenario

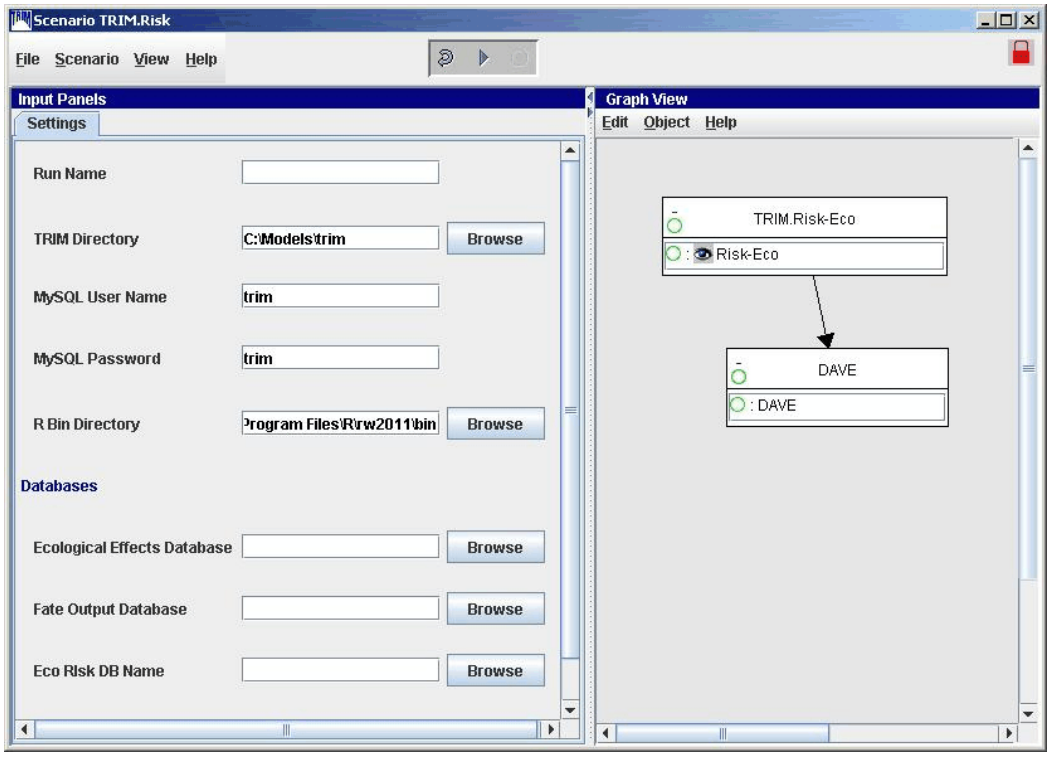
Step 3 Complete the Input Panels

Set the necessary parameter settings via the scenario "Input Panels" (note there is only one tab on the "Input Panels" in this project; it is called "Settings"). The "Input Panels" present those scenario and module instance parameters that most commonly need attention from users (see Exhibits 3-7 and 3-8 below). Note that some of these parameters will already be set by the TRIM installation program. Users should use caution when making changes to any preset parameters.

Exhibit 3-7. Parameters on "Settings" Tab of "TRIM.Risk" Scenario

Parameter	Default Value	Set via Installer?	Description
Run Name	–	No	Name of the unique scenario run.
TRIM Directory	–	Yes	TRIM installation directory (i.e., the directory into which the TRIM.Risk installation packages were installed).
MySQL User Name	–	Yes	User name for access to the MySQL databases created or used for the TRIM.Risk projects.
MySQL Password	–	Yes	Password for access to the MySQL databases created or used for the TRIM.Risk projects.
R Bin Directory	–	Yes	Location of the bin directory for the R program.
Ecological Effects Database	–	No	Name of database with ecological toxicity reference values.
FaTE Output Database	–	No	Name of output database from TRIM.FaTE simulation.
Eco Risk DB Name	–	No	Name of the output database from TRIM.Risk$_{Eco}$.

Exhibit 3-8. "Settings" Tab of "TRIM.Risk" Scenario



Step 4 Set Any Additional Parameters

In most cases, users will not be required to set any parameters beyond those in the "Input Panels." However, in some applications users may find it necessary to view or edit additional scenario or module instance parameters. To access the scenario parameters, select "Edit Parameter Values" from the "Scenario" menu of the scenario window. Access the module instance parameters by double-clicking on a module instance name in the "Graph View" pane, which opens the module instance window. On the "Parameters" tab, set values by typing in the "Value" column, clicking on a check box, or selecting the "Browse" button to locate the file. Note that this step may not be required for some applications. No user interaction is required on the "Module" tab.

Refer to: Volume I, Section 3.1.7, Specifying MIMS Scenario Parameter Values

Volume I, Section 3.1.8, Specifying MIMS Module Instance Parameter Values

Appendix B, Scenario and Module Instance Parameters

Step 5 Start the Scenario

After setting all parameter values, the scenario is ready to run.[5] Select "Execute Ready Modules" from the "Scenario" menu or click on the "play" button ( ) at the top of the scenario window. The TRIM.Risk$_{Eco}$ GUI will automatically open.

Refer to: Volume I, Section 3.1.9, Running a MIMS Scenario

Step 6 Set Inputs and Run TRIM.Risk$_{Eco}$

Make the appropriate selections in the TRIM.Risk$_{Eco}$ GUI based on the needs of the risk assessment. After configuring the TRIM.Risk$_{Eco}$ GUI and clicking "Save" and "Exit," the risk calculations will be performed.

Refer to: Section 4.2, TRIM.Risk$_{Eco}$

Chapter 2, Risk Outputs

[5] Note that if this scenario has been run previously, and/or if selections have been made in the TRIM.Risk$_{Eco}$ GUI in this MIMS scenario previously, the scenario must be reset and the GUI must be reloaded before the run can begin. Select "Reset All" from the "Scenario" drop-down menu to return all processes to a ready-to-execute state (resetting will ***not*** clear any scenario or module instance parameters that have been set), and right-click on the TRIM.Risk$_{Eco}$ module instance and select "Reload Custom GUI," to clear previous selections from the GUI.

Step 7 Run DAVE

After the TRIM.Risk$_{Eco}$ component completes, DAVE will automatically open and can be used to analyze or export the ecological risk estimates which are contained in the newly created ecological risk database. Use DAVE to generate tables and charts of ecological risk estimates or to create delimited text exports of the data that can be imported into spreadsheet or data analysis software. Alternatively, risk estimates can be retrieved directly from the database using MySQL queries.

Refer to: Section 4.3, DAVE

Volume I, Section 3.3, Basic Operations with DAVE

DAVE User's Guide

[This page intentionally left blank.]

4. Components of the Ecological Risk Assessment Scenarios

This section provides descriptions to aid in the setup and execution of each of the components included in the ecological risk assessment scenarios. Refer to Appendix A for a flowchart that shows how the components fit together. Section 4.1 describes the setup of TRIM.FaTE, Section 4.2 describes the setup of TRIM.Risk$_{Eco}$, and Section 4.3 describes the setup of DAVE.

4.1 TRIM.FaTE

TRIM.FaTE is the ecological exposure module of the TRIM model which can be run either inside or outside of MIMS. See the TRIM.FaTE User's Guide for guidance on configuring a TRIM.FaTE simulation. TRIM.FaTE can be utilized in two different ways to generate ecological exposure estimates that can be input to TRIM.Risk. A user can configure a TRIM.FaTE scenario to produce the appropriate outputs for an ecological risk assessment, save the TRIM.FaTE scenario and close TRIM.FaTE, and then run TRIM.FaTE through MIMS with the "TRIM.FaTE w. TRIM.Risk" MIMS scenario. Alternatively, a user can configure TRIM.FaTE and run a TRIM.FaTE simulation outside of MIMS. The ecological exposure database created by TRIM.FaTE can then be input to TRIM.Risk using the "TRIM.Risk" MIMS scenario (without running TRIM.FaTE through MIMS).

In order to use TRIM.FaTE output in TRIM.Risk, the TRIM.FaTE output must be in a MySQL database. This is assured by configuring the TRIM.FaTE scenario (within TRIM.FaTE, outside of MIMS) with the appropriate property values (i.e., settings). Instructions for setting these properties are provided in this section.

4.1.1 TRIM.FaTE Setup

Various TRIM.FaTE outputs can be used as inputs to TRIM.Risk$_{Eco}$. If used as ecological exposure inputs, these TRIM.FaTE outputs need to be in the units specified in Exhibit 4-1. All of these outputs must be written to the TRIM.FaTE results MySQL database in order for TRIM.Risk$_{Eco}$ to use them as inputs. In order to do this, certain TRIM.FaTE library and scenario properties must be set correctly. The default settings in the TRIM.FaTE Public Reference Library are compatible with TRIM.Risk. However, if the public reference library has been changed, or if a different TRIM.FaTE library is used, the user should refer to Appendix E for the necessary library configuration to produce the TRIM.Risk compatible output database.

Exhibit 4-1. TRIM.FaTE Outputs for Use in TRIM.Risk$_{Eco}$

TRIM.FaTE output	Required units for TRIM.Risk$_{Eco}$
Vapor air concentrations mapped to surface soil compartments	ug/m^3
Surface soil concentrations	ug/g dry weight
Root zone soil concentrations	ug/g dry weight
Sediment concentrations	ug/g wet weight **and** ug/g dry weight
Surface water concentrations	mg/L
Dissolved surface water concentrations	mg/L
Biotic intake for all mammal and bird compartments	mg/kg-day
Biota concentrations (i.e., body burdens) for all mammal and bird compartments	mg/kg wet weight

Certain values must also be set in the TRIM.FaTE scenario properties in order to run TRIM.Risk with the TRIM.FaTE output database. See the TRIM.FaTE User's Guide for more information about TRIM.FaTE scenario properties. The steps for setting these properties are described here:

1. The following TRIM.FaTE scenario properties must be set to "true":

   - **exportRiskEcoInputs:** tells TRIM.FaTE to create a MySQL database;
   - **exportAllResultstoDB:** exports instantaneous or steady-state results to the database, depending on whether the simulation was dynamic or steady-state[6]; and
   - **exportConcentration:** exports concentration results to the database.

2. The following TRIM.FaTE scenario properties may be set to true or false, depending on the needs of the assessment:

   - **exportAvgResultstoDB:** exports the average results to the database, if an averaging period was specified. To use averaged results with the initial release of TRIM.Risk, the averaging period (specified by the FaTE property averagingInterval) must be set to one of the following: 12 hour, 24 hour, 168 hour, monthly, or annual[7]; and

[6] The initial TRIM.Risk release requires the TRIM.FaTE property exportAllResultstoDB to be true, however this requirement might change in subsequent releases.

[7] Note that if the averaging interval is input as 720 or 8760 hours, the interval will be interpreted as monthly or annual, respectively, by TRIM.Risk$_{Eco}$.

- **exportBiotaIntakeRates:** exports to the database the chemical intake rates for the biota that were included in the scenario. Note that if the TRIM.FaTE simulation was a steady-state run, the biota intake rates will all be zero and cannot be used by TRIM.Risk$_{Eco}$.

3. The property "mySQLDataDir" must be set by the user to indicate the location of the "data" directory folder in the MySQL installation.

4. By default, TRIM.FaTE sets the name of the MySQL output database to the name of the scenario appended by "DB." For example, for a TRIM.FaTE scenario named "HgSimpleLake50years," the MySQL output database would be named "HgSimpleLake50yearsDB." It is important to note that in creating the database, TRIM.FaTE will overwrite any existing database with the same name in the specified output directory. To avoid this, users desiring to run a scenario twice with some slight differences between the two should change the name of the scenario for the second run (this will create two separate scenarios, facilitating documentation of differing inputs and setup specifications), or specify a different output directory. Alternatively, the user can specify an alternate database name, provided the name is a valid MySQL database name (see text box below).[8]

VALID DATABASE NAMES IN MySQL

A MySQL database name must be 64 characters or less and cannot contain spaces or any of the following characters:"\", "/", or ".". In addition, the database name must not match any of the "reserved" names in MySQL (see http://www.mysql.com/doc/en/Reserved_words.html).

4.2 TRIM.RISK$_{Eco}$

TRIM.Risk$_{Eco}$ assists ecological risk assessors by deriving hazard quotients for ecological receptors from ecological exposure estimates. The TRIM.Risk$_{Eco}$ GUI assists the assessor in specifying the details needed for the TRIM.Risk$_{Eco}$ analysis. This section describes the steps for setting up TRIM.Risk$_{Eco}$ to suit the needs of the assessment. See Appendix F.2 for additional information on TRIM.Risk$_{Eco}$.

As with running TRIM.FaTE in a MIMS scenario, scenario and module instance parameters for TRIM.Risk$_{Eco}$ are set through the "Input Panels" of the MIMS scenario window.

[8] The database name is a TRIM.FaTE scenario property. Scenario properties can be set by the user in the scenario view within the TRIM.FaTE scenario window (see Module 9 of TRIM.FaTE User's Guide for more information on scenario properties). In TRIM.FaTE, to set the database name, the user must first add the "outputDatabaseName" property to the scenario by clicking the "New" button above the scenario properties, selecting "outputDatabaseName" from the list that is displayed, clicking "OK," selecting "None" (because the property is not chemical-specific), and clicking "OK" from the last screen. Then a valid MySQL database name can be typed in the "Value" field for the "outputDatabaseName" property. Be sure to hit return after typing the name to insure that it is saved to the "Value" field.

See Step 3 of the instructions in both Sections 3.1 and 3.2 for a list of the parameters in the "Input Panels." In addition, a complete list of all of the module instance parameters for TRIM.Risk$_{Eco}$ are provided in Appendix B, and instructions for accessing the complete module instance parameter table are in Volume I, Section 3.1.8.

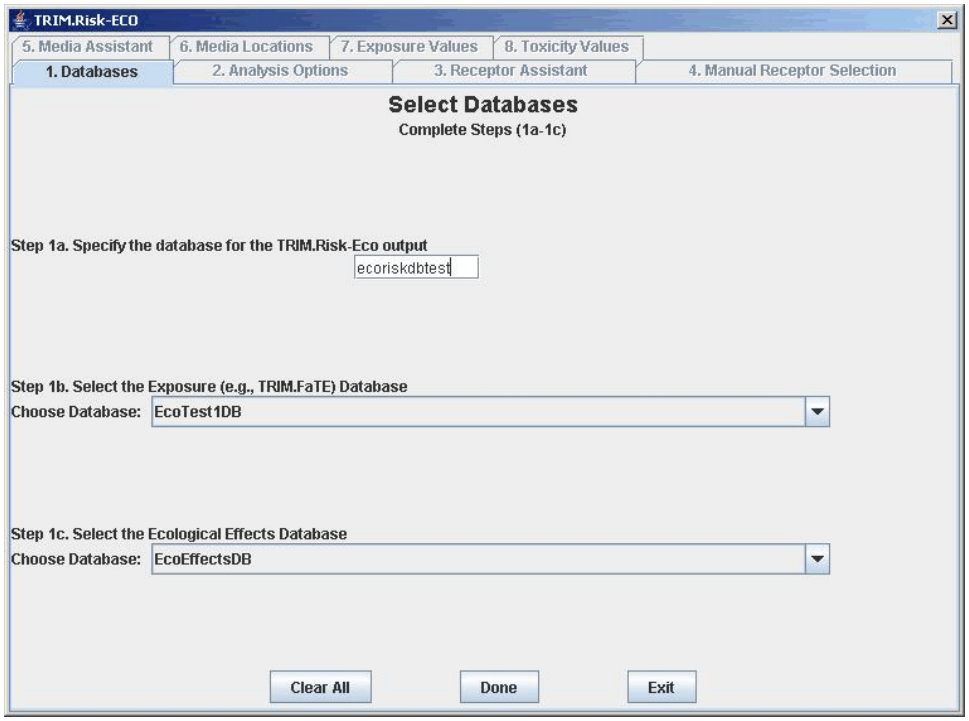
4.2.1 TRIM.Risk$_{Eco}$ Setup

The TRIM.Risk$_{Eco}$ GUI opens automatically when running either the "TRIM.Risk w. TRIM.FaTE" or "TRIM.Risk" scenario. It will open after TRIM.FaTE is complete in the "TRIM.Risk w. TRIM.FaTE" scenario or after starting the model execution in the "TRIM.Risk" scenario. The GUI contains eight tabs. Each tab is addressed in sequence during the TRIM.Risk$_{Eco}$ setup (i.e., no tabs can be skipped); however, users can return to previously completed tabs provided they proceed through the tabs in sequence (including all tabs after the changed tab, even if these tabs were previously completed) after making any changes.

Listed here are detailed step-by-step instructions for setting up TRIM.Risk$_{Eco}$, organized by the tab number in the TRIM.Risk$_{Eco}$ GUI.

Tab 1 **Databases**

The TRIM.Risk$_{Eco}$ GUI will open to Tab 1 (Databases). This tab is used to create or select the MySQL databases required to perform the ecological risk calculations. Users **must** complete Tab 1 before proceeding to any other tabs. Some or all of the database names may already be filled in based on the database names that were set in the scenario "Input Panels." If the user changes here names that were previously specified, the new names will be conveyed to the "Input Panels."



Step 1a Specify the database for the TRIM.Risk$_{Eco}$ output

Enter a name for the TRIM.Risk$_{Eco}$ output database. This name must be a valid MySQL database name (see text box in Section 4.1). If the user enters an invalid name, an error message will appear after the "Done" button on this tab is clicked. If a database with this name already exists, the user will be given the choice to overwrite the existing database, or to rename the new database. Otherwise, after clicking "Done," a box opens telling the user that a MySQL database was created. The database will be created in the MySQL data directory, which is specified by the user during the TRIM.Risk installation.

Step 1b Select the Exposure (e.g., TRIM.FaTE) Database

Select the TRIM.FaTE database[9] to be used in the analysis from the drop-down menu which contains a list of available databases.

Step 1c Select the Ecological Effects Database

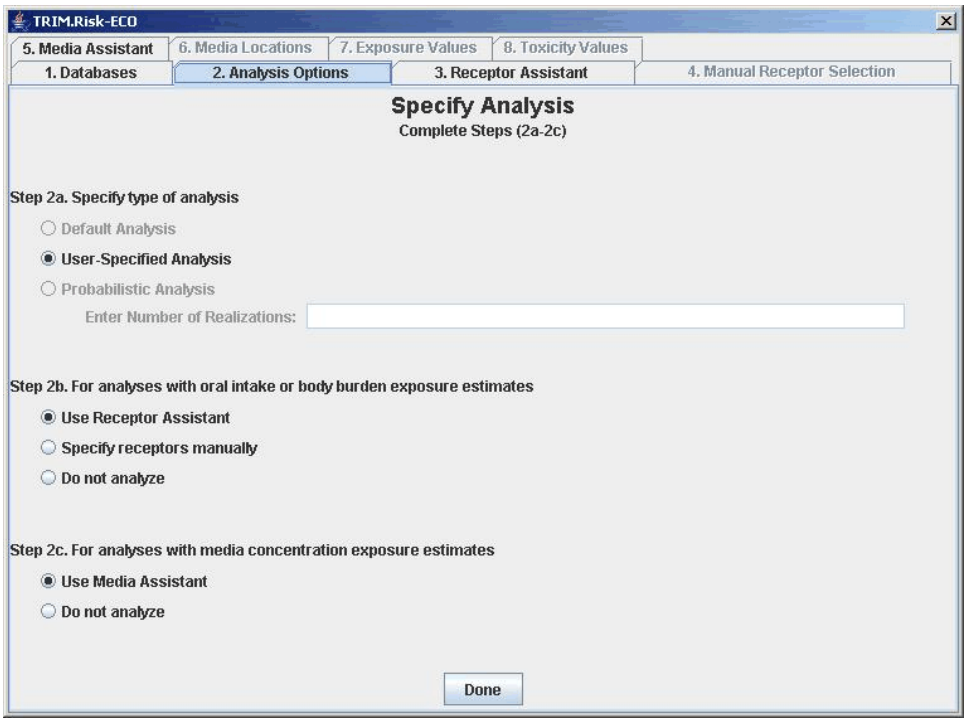
Select the ecological effects (i.e., ecological toxicity) MySQL database from the drop-down menu of available databases.

After completing Steps 1a through 1c, click "Done" to confirm that the output database name and the selected TRIM.FaTE and ecological effects databases are valid. If any of the database names are not selected, an invalid name was used in Step 1a, or an incompatible database was chosen in Steps 1b or 1c, an error message will appear instructing the user to correct the problem. To delete the database names specified in Steps 1a through 1c, click the "Clear All" button. Names of valid databases will have to be specified, however, before proceeding to other tabs. After clicking "Done," if there are no errors, a window will pop up that says "A MySQL database was created." Click "OK," the pop-up will close, and the GUI will open the next tab.

Tab 2 Analysis Options

After completing Tab 1, the databases will be created and Tab 2 will open. On this tab, the user can specify the types of analysis to perform.

[9] As discussed previously, in addition to output from TRIM.FaTE, TRIM.Risk$_{Eco}$ can be run with output from other ecological exposure models; however, these results must be provided in the TRIM.FaTE MySQL output format as presented in Appendix C.



Step 2a Specify type of analysis

There are three analysis types "Default Analysis," "User-Specified Analysis," and "Probabilistic Analysis." **In the first TRIM.Risk release, the only available option is "User-Specified Analysis."** This analysis type allows the user to select the ecological receptors, toxicity values, and times of interest to generate deterministic risk results.

Step 2b For analyses with oral intake or body burden exposure estimates

If users are calculating ecological risk based on exposure in terms of oral intake or body burden (see Chapter 2 for description of the risk output options), they may either pick the option to "Use Receptor Assistant" (see Tab 3) or the option to "Specify receptors manually" (see Tab 4). Both of these options allow users to "match" TRIM.FaTE compartment names with ecological receptor names for which the ecological toxicity database has toxicity values in those exposure terms for the chemicals being assessed. These matches are a first step in specifying to TRIM.Risk which toxicity values are to be combined with which exposure estimates in creating hazard quotients. The "Receptor Assistant" gives the user a "head start" by recognizing and pairing up all of the exactly matching names; it then allows the user to manually match up TRIM.FaTE compartment names with ecological receptors having different names (e.g., a TRIM.FaTE "white-tailed deer" compartment with "black-tailed deer"). The option to "Specify receptors manually" allows the user to view all possible combinations in one table and manually select all the desired combinations. Select "Do not analyze" if analysis of these types of outputs is not desired.

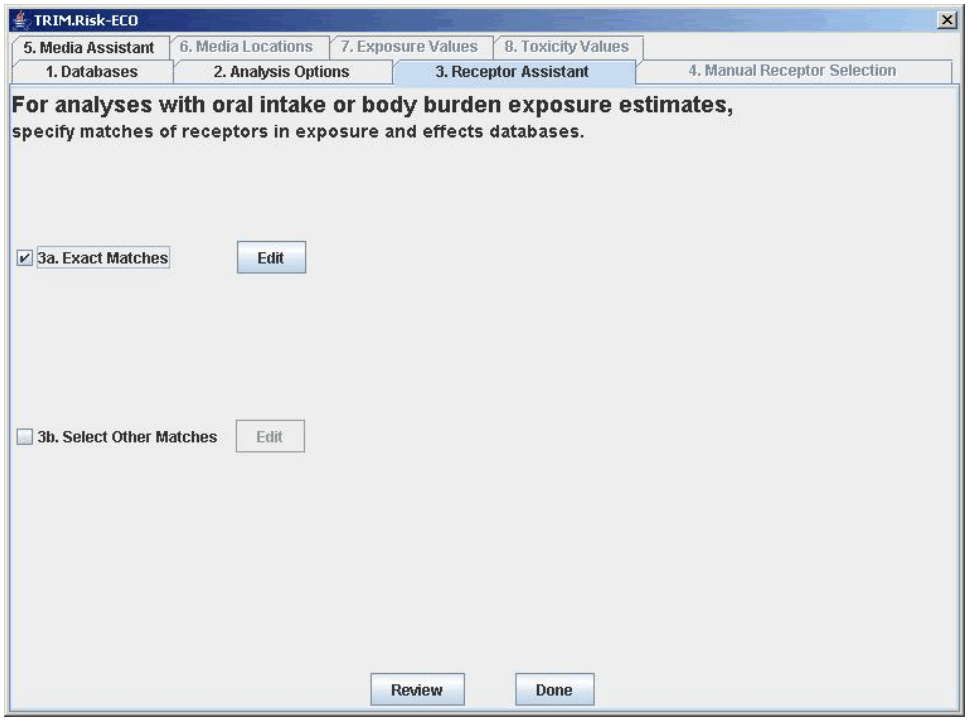
Step 2c **For analyses with media concentration exposure estimates**

To select ecological receptors for an ecological risk assessment that will use exposure in terms of media concentrations (see Chapter 2 for a description of the risk output options), select "Use Media Assistant;" otherwise select "Do not analyze."

After the appropriate selections are made on Tab 2, click "Done."

Tab 3 Receptor Assistant

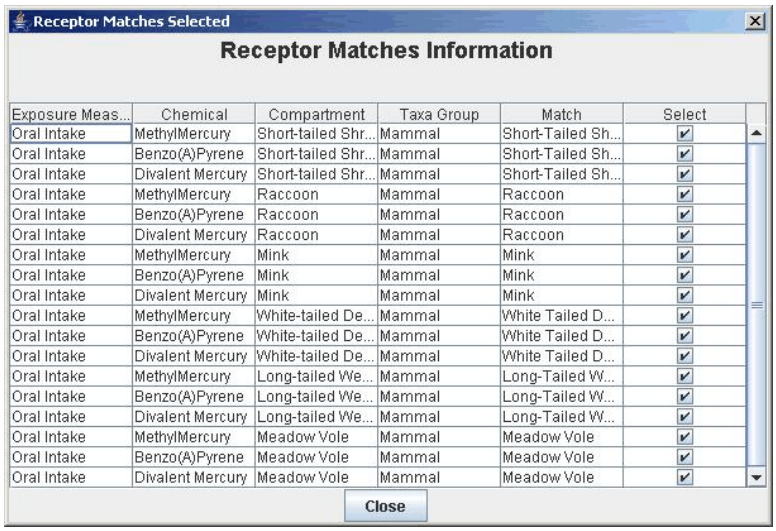
Tab 3 will open if "Use Receptor Assistant" was selected in Step 2b of Tab 2. On this tab, the user can match receptor compartments in the TRIM.FaTE database to receptors in the ecological toxicity database.[10]



[10] TRIM.FaTE classifies each compartment by a hierarchical classification system. For example, the bald eagle compartment is classified with the category "Bird|Bald Eagle" and the top layer of the soil is classified as "Abiotic|Soil|Surface Soil." This classification system is used to match the ecological exposure results from TRIM.FaTE with appropriate toxicity data from the ecological toxicity database. TRIM.Risk$_{Eco}$ determines all of the possible matches between the compartments from TRIM.FaTE (based on the categories for each compartment) to the receptors and receptor groups in the ecological toxicity database, and the user can pick the desired matches from this list. Matches are made when the specific TRIM.FaTE compartment name is the same as a receptor name (exact matches); when any name in the TRIM.FaTE category (e.g., bird, mammal) matches the receptor name or receptor group (other matches); and when abiotic compartment types from TRIM.FaTE match the abiotic media used for a media-based toxicity value in the ecological toxicity database (media matches).

Step 3a **Exact Matches**

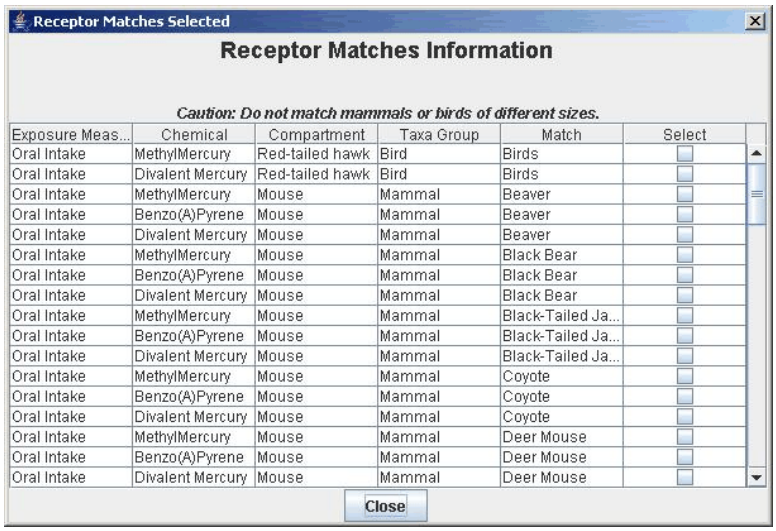
Check this box, and click on the "Edit" button to match all or some TRIM.FaTE compartments to receptors in the ecological toxicity database with the same name for the type of exposure measure (e.g., oral intake) and the chemical being assessed. A window will open in which all exact matches are shown as selected (see below). De-select any matches based on the needs of that assessment, or leave all selected, and click on the "Close" button when finished.



Exposure Meas...	Chemical	Compartment	Taxa Group	Match	Select
Oral Intake	MethylMercury	Short-tailed Shr...	Mammal	Short-Tailed Sh...	✔
Oral Intake	Benzo(A)Pyrene	Short-tailed Shr...	Mammal	Short-Tailed Sh...	✔
Oral Intake	Divalent Mercury	Short-tailed Shr...	Mammal	Short-Tailed Sh...	✔
Oral Intake	MethylMercury	Raccoon	Mammal	Raccoon	✔
Oral Intake	Benzo(A)Pyrene	Raccoon	Mammal	Raccoon	✔
Oral Intake	Divalent Mercury	Raccoon	Mammal	Raccoon	✔
Oral Intake	MethylMercury	Mink	Mammal	Mink	✔
Oral Intake	Benzo(A)Pyrene	Mink	Mammal	Mink	✔
Oral Intake	Divalent Mercury	Mink	Mammal	Mink	✔
Oral Intake	MethylMercury	White-tailed De...	Mammal	White Tailed D...	✔
Oral Intake	Benzo(A)Pyrene	White-tailed De...	Mammal	White Tailed D...	✔
Oral Intake	Divalent Mercury	White-tailed De...	Mammal	White Tailed D...	✔
Oral Intake	MethylMercury	Long-tailed We...	Mammal	Long-Tailed W...	✔
Oral Intake	Benzo(A)Pyrene	Long-tailed We...	Mammal	Long-Tailed W...	✔
Oral Intake	Divalent Mercury	Long-tailed We...	Mammal	Long-Tailed W...	✔
Oral Intake	MethylMercury	Meadow Vole	Mammal	Meadow Vole	✔
Oral Intake	Benzo(A)Pyrene	Meadow Vole	Mammal	Meadow Vole	✔
Oral Intake	Divalent Mercury	Meadow Vole	Mammal	Meadow Vole	✔

Step 3b **Select Other Matches**

Other matches can be made between TRIM.FaTE compartments and receptors in the ecological toxicity database, even if the receptor name is not an exact match, as long as the taxa group is the same. Check the box next to option 3b of Tab 3, click on the "Edit" button, and a window will open which shows all the possible combinations between TRIM.FaTE compartments and receptors in the ecological toxicity database based on taxa group specifications, exposure measure type, and chemical. None of the matches are selected by default. Check the boxes next to the desired matches. Note that it is recommended that mammals or birds of significantly different sizes not be matched. After making all selections, click on the "Close" button.

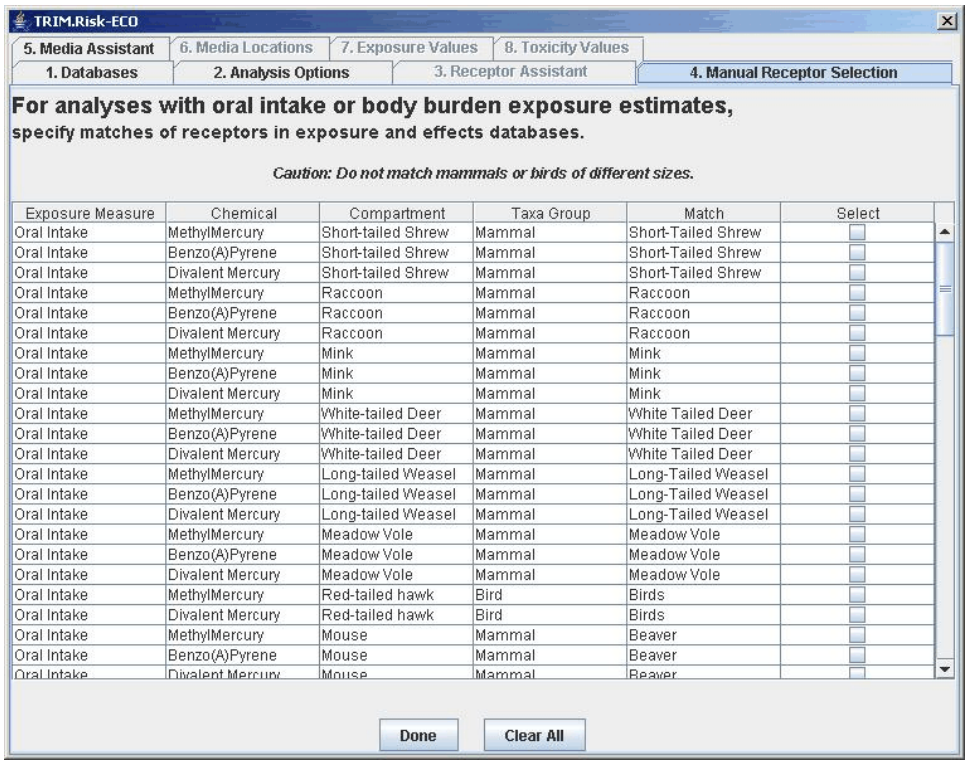


Exposure Meas...	Chemical	Compartment	Taxa Group	Match	Select
Oral Intake	MethylMercury	Red-tailed hawk	Bird	Birds	☐
Oral Intake	Divalent Mercury	Red-tailed hawk	Bird	Birds	☐
Oral Intake	MethylMercury	Mouse	Mammal	Beaver	☐
Oral Intake	Benzo(A)Pyrene	Mouse	Mammal	Beaver	☐
Oral Intake	Divalent Mercury	Mouse	Mammal	Beaver	☐
Oral Intake	MethylMercury	Mouse	Mammal	Black Bear	☐
Oral Intake	Benzo(A)Pyrene	Mouse	Mammal	Black Bear	☐
Oral Intake	Divalent Mercury	Mouse	Mammal	Black Bear	☐
Oral Intake	MethylMercury	Mouse	Mammal	Black-Tailed Ja...	☐
Oral Intake	Benzo(A)Pyrene	Mouse	Mammal	Black-Tailed Ja...	☐
Oral Intake	Divalent Mercury	Mouse	Mammal	Black-Tailed Ja...	☐
Oral Intake	MethylMercury	Mouse	Mammal	Coyote	☐
Oral Intake	Benzo(A)Pyrene	Mouse	Mammal	Coyote	☐
Oral Intake	Divalent Mercury	Mouse	Mammal	Coyote	☐
Oral Intake	MethylMercury	Mouse	Mammal	Deer Mouse	☐
Oral Intake	Benzo(A)Pyrene	Mouse	Mammal	Deer Mouse	☐
Oral Intake	Divalent Mercury	Mouse	Mammal	Deer Mouse	☐

After making selections in Steps 3a and/or 3b, the "Review" button on Tab 3 can be selected to view a summary table of the compartment-receptor matches. (Note that in the initial release of TRIM.Risk, clicking the "Review" button is ***required*** if selections have been made using option 3b.) Click the "Done" button on Tab 3 to move to the next tab. A message will appear that indicates the name and location of a file that lists the receptor matches. Click "OK" to continue.

Tab 4 Manual Receptor Selection

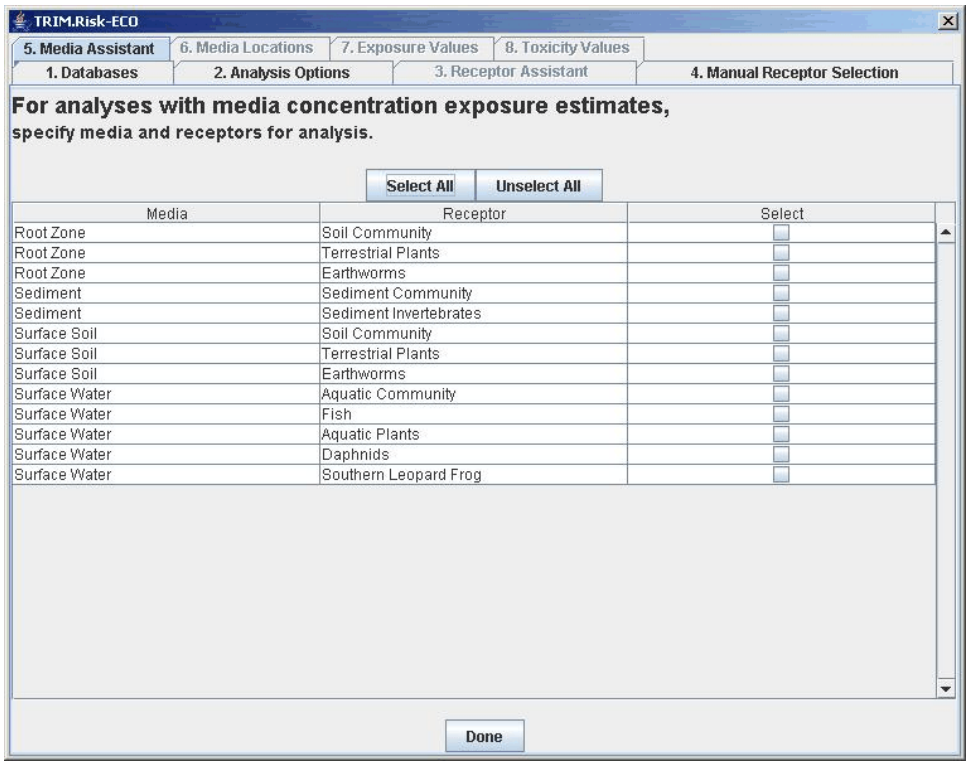
Tab 4 opens if "Specify receptors manually" was selected in Step 2b of Tab 2. This tab contains a table of all possible matches between TRIM.FaTE compartments and receptors in the ecological toxicity database based on exposure measure, chemical, and either receptor name or taxa group name (i.e., the options are essentially the options on Tables 3a and 3b combined into one table). The exposure measure, chemical, TRIM.FaTE compartment, group, and ecological toxicity database receptor match are all listed for each possible combination. Click the box in the "Select" column to select the appropriate matches. Click the "Done" button when complete to move to the next tab.



Exposure Measure	Chemical	Compartment	Taxa Group	Match	Select
Oral Intake	MethylMercury	Short-tailed Shrew	Mammal	Short-Tailed Shrew	
Oral Intake	Benzo(A)Pyrene	Short-tailed Shrew	Mammal	Short-Tailed Shrew	
Oral Intake	Divalent Mercury	Short-tailed Shrew	Mammal	Short-Tailed Shrew	
Oral Intake	MethylMercury	Raccoon	Mammal	Raccoon	
Oral Intake	Benzo(A)Pyrene	Raccoon	Mammal	Raccoon	
Oral Intake	Divalent Mercury	Raccoon	Mammal	Raccoon	
Oral Intake	MethylMercury	Mink	Mammal	Mink	
Oral Intake	Benzo(A)Pyrene	Mink	Mammal	Mink	
Oral Intake	Divalent Mercury	Mink	Mammal	Mink	
Oral Intake	MethylMercury	White-tailed Deer	Mammal	White Tailed Deer	
Oral Intake	Benzo(A)Pyrene	White-tailed Deer	Mammal	White Tailed Deer	
Oral Intake	Divalent Mercury	White-tailed Deer	Mammal	White Tailed Deer	
Oral Intake	MethylMercury	Long-tailed Weasel	Mammal	Long-Tailed Weasel	
Oral Intake	Benzo(A)Pyrene	Long-tailed Weasel	Mammal	Long-Tailed Weasel	
Oral Intake	Divalent Mercury	Long-tailed Weasel	Mammal	Long-Tailed Weasel	
Oral Intake	MethylMercury	Meadow Vole	Mammal	Meadow Vole	
Oral Intake	Benzo(A)Pyrene	Meadow Vole	Mammal	Meadow Vole	
Oral Intake	Divalent Mercury	Meadow Vole	Mammal	Meadow Vole	
Oral Intake	MethylMercury	Red-tailed hawk	Bird	Birds	
Oral Intake	Divalent Mercury	Red-tailed hawk	Bird	Birds	
Oral Intake	MethylMercury	Mouse	Mammal	Beaver	
Oral Intake	Benzo(A)Pyrene	Mouse	Mammal	Beaver	
Oral Intake	Divalent Mercury	Mouse	Mammal	Beaver	

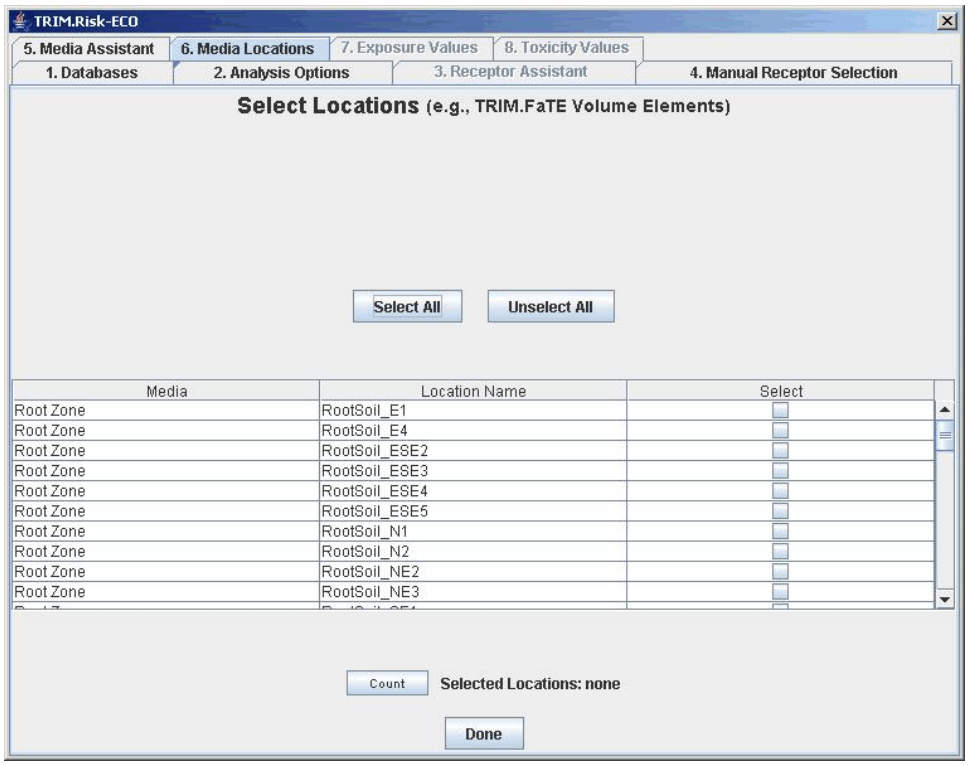
Tab 5 Media Assistant

Tab 5 will open if "Use Media Assistant" was selected in Step 2c of Tab 2. On this tab, the user can select the media and ecological receptors to be assessed using media concentrations for exposure estimates. The list of receptors shown on this tab is generated from a comparison of the list of abiotic media in the TRIM.FaTE database with the abiotic media concentration toxicity values for the chemicals being assessed that are available in the ecological toxicity database. The user may select media and receptors to be included in the TRIM.Risk assessment by clicking the box next to the media and receptor combination. Clicking the "Select All" button will select all, while clicking the "Unselect All" button will clear all selections. When finished, click the "Done" button. To select none of the media for assessment, click "Done" without clicking any of the "Select" boxes.



Tab 6 Media Locations

On this tab, the user can select the locations at which to calculate the HQs using abiotic media concentrations as the exposure estimates (using the selections made on Tab 5). The table on this tab lists the abiotic media selected in Tab 5 and all of the locations (e.g., in terms of TRIM.FaTE volume element names) with the corresponding media types from the TRIM.FaTE database. Click the boxes in the "Select" column next to the locations of interest. Note that the capability to restrict the locations to fewer than all is not yet implemented for hazard calculations using oral intake or body burden for ecological exposure. Thus, if selections are made in Tabs 3 and/or 4, the risk calculations will be performed for the selected receptors in all locations for which data are available (e.g., all TRIM.FaTE volume elements where they were modeled in the TRIM.FaTE simulation).

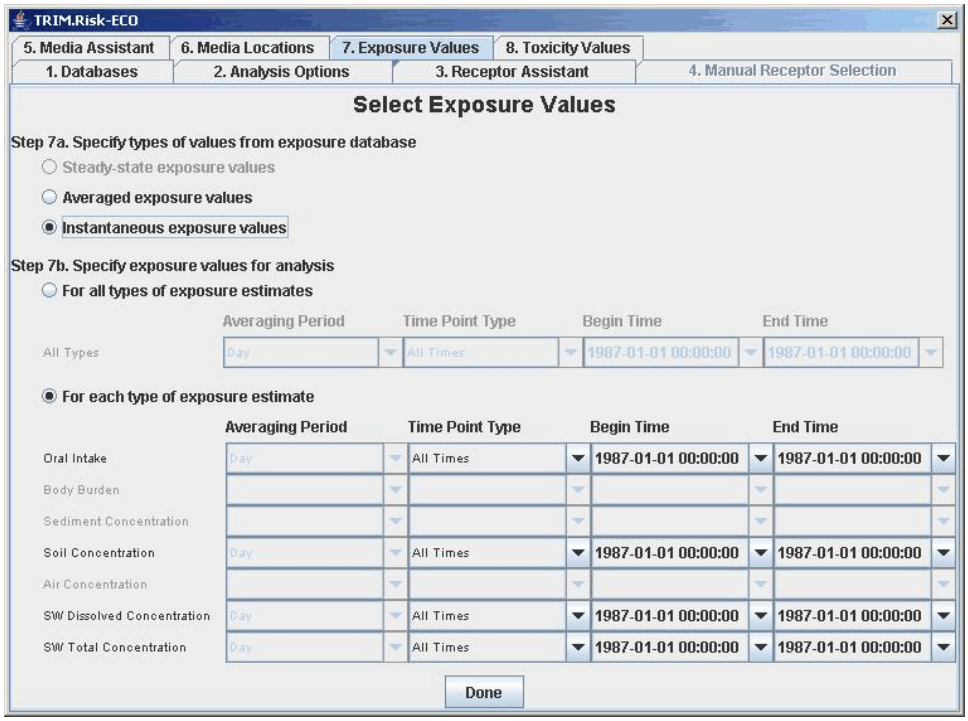


Tab 7 Exposure Values

On Tab 7, the user can specify temporal characteristics of the exposure values to be used in the risk characterization, depending on the time steps of the data in the TRIM.FaTE database.

Step 7a Specify types of values from exposure database

Three types of temporal data could be output from TRIM.FaTE and used in TRIM.Risk$_{Eco}$. The types available in the TRIM.FaTE database can be selected in this step. Data types not available will be grayed out. Steady-state exposure values are available if the TRIM.FaTE run was set up and run as steady-state; averaged exposure values are available if the results from a dynamic run were exported as averages to the output database; and instantaneous results (i.e., results exported every certain number of timesteps) will be available if the TRIM.FaTE run was dynamic. Select the type of data to analyze; only one type can be selected. Note that if the data type is steady-state, the options in Step 7b will be grayed out because they are not applicable to steady-state results.



Step 7b **Specify exposure values for analysis**

In this step, the user makes temporal specifications regarding the exposure data to be compared with each type of exposure estimate (e.g., oral intake, sediment concentration, soil concentration, etc). The first option in this step ("For all types of exposure estimates") should be selected if the same temporal specification is to be used for all exposure types (e.g., oral intake, sediment concentration, etc). The second option ("For each type of exposure estimate") should be selected to make a different temporal specification for each type of exposure estimate. If the second option is selected, choices for types of exposure estimates that have not been selected in previous tabs will be grayed out.

If the "Averaged exposure values" option is selected in Step 7a, the averaging period(s) in the database will be indicated by the first drop-down menu for each exposure estimate type. Note that only the averaging periods in the database which are supported by this version of TRIM.Risk$_{Eco}$ will show up in the drop-down menu. In the initial release, these supported averaging periods are 12 hour (twice-daily), 24 hour (daily), 168 hour (weekly), monthly, and annual. If "Averaged exposure values" is selected in Step 7a, but there are no averaging periods in the drop-down menu, that means there are no supported averaging periods in the database, and no hazard quotients will be calculated for average results (however, instantaneous results can still be analyzed if "Instantaneous exposure values" is selected in 7a). In most cases there will only be one supported averaging period in the database. To make a temporal specification for a type of exposure estimate or all types of exposure estimates, the user should first select a time point type from the pull-down list in the "Time Point Type" column. The time point type options include All Times, Maximum Time, Last Time, Time Point, and Time Range. Depending on the time

point type, the user may need to specify additional information in the "Begin Time" and "End Time" columns. Exhibit 4-2 describes each of the Time Point Types and how to set them once the type is selected. After selecting time points for each type of exposure estimate, click the "Done" button. If the second option was selected for Step 7b and different time point types are selected for each exposure estimate type, a table will appear listing the type of exposure estimate, the selected time point or range, the data interval (if applicable), and the averaging time (if applicable). If some of the temporal information is missing or contains errors (e.g., begin time greater than end time), an error message will appear describing the problem.

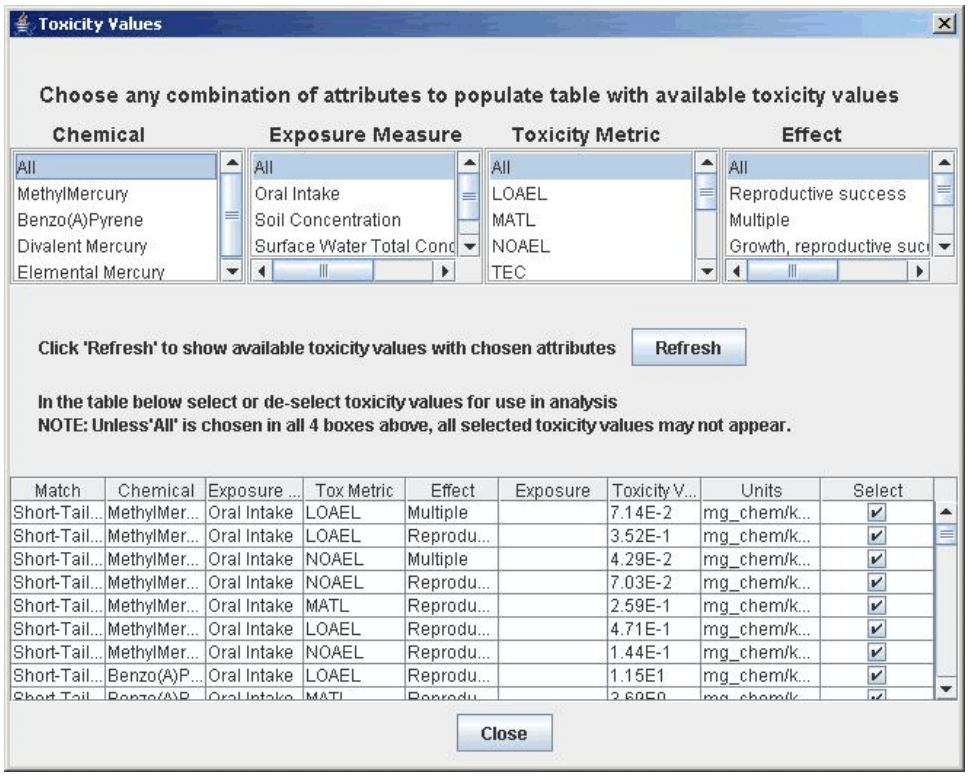
Exhibit 4-2. Time Point Types

Time Point Type	Description	How to Set the Time Point
All Times	This specifies that all time points in the TRIM.FaTE database will be compared against the selected toxicity values.	After selecting the Time Point Type, no further action is necessary.
Maximum Time	This specifies that the maximum value within the TRIM.FaTE database for the relevant media or biota will be compared to the selected toxicity values.	After selecting this Time Point Type, no further action is necessary.
Last Time	This specifies that the exposure value at the last time step for the selected receptor(s) in the TRIM.FaTE database will be compared to the selected toxicity values.	After selecting this Time Point Type, no further action is necessary.
Time Point	This specifies that the exposure value at a user-specified time point for the selected receptor(s) in the TRIM.FaTE database will be compared to the selected toxicity values.	When this Time Point Type is selected, the user must select one of the time points from the pull-down list in the "Begin Time" column. In addition, the same time must be selected from the "End Time" column.
Time Range	This specifies that the exposure value for all of the time points within a user-specified range for the selected receptor(s) in the TRIM.FaTE database will be compared to the selected toxicity values.	When this Time Point Type is selected, the user must select one time point from the pull-down list in the "Begin Time" column and another time point from the pull-down list in the "End Time" column. The selected value in the "End Time" column **must** be greater than the selected value in the "Begin Time" column.

Tab 8 Toxicity Values

Tab 8 is used to select the appropriate toxicity values for the analysis. In the initial release of TRIM.Risk, the main page of this tab contains a "Select All" check box. Users may check this or not, and they then should click on the "Edit/Review" button to confirm and refine their selections. In subsequent TRIM.Risk releases, the GUI will improve upon this initial manual approach. Note that – particularly in the initial release – it is incumbent on users to carefully select toxicity values such that they are appropriate to the exposure estimates (e.g., in terms of exposure duration, etc).

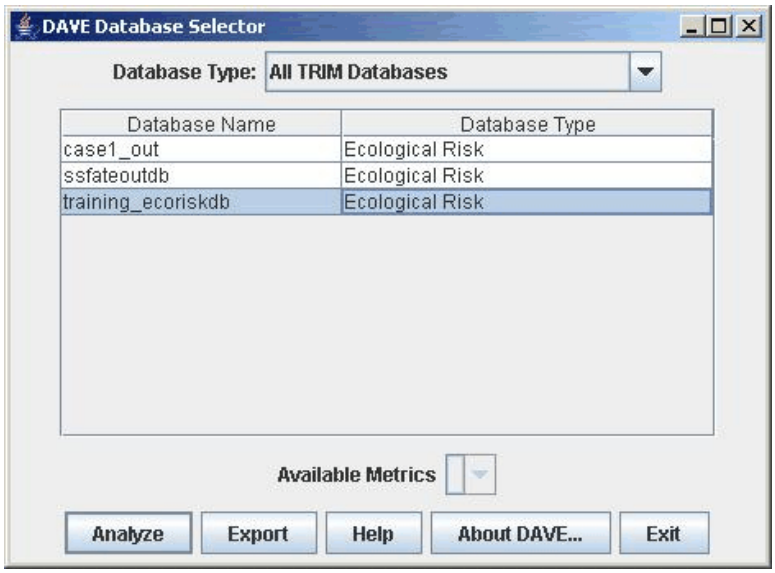
A window will open with all possible toxicity values from the ecological toxicity database based on all selections made in the previous tabs. The toxicity values can be narrowed down based on the Chemical, Exposure Measure, Toxicity Metric, and Effect options. Select options in the tables at the top of the window, and click the "Refresh" button to generate the table at the bottom of the tab with the specified selections. The table shows the information about the toxicity value from the ecological toxicity database. To select the appropriate toxicity value(s), click the boxes in the "Select" column. When finished, click the "Close" button. After closing the table, click "Save" on Tab 8 to save the selections, and then click "Exit." A "Confirm Exit" dialogue box will appear, click "Yes," and the GUI will close.



4.3 DAVE

After running one of the ecological risk assessment scenarios in MIMS, the ecological risk estimates are contained in the newly created ecological risk database (with the name specified by the user prior to running the scenario) in MySQL format. Users can use the Data Analysis and Visualization Engine (DAVE) to generate tables and charts of ecological risk estimates, as well as to create delimited text exports of the data that can be imported into spreadsheet or data analysis software. The DAVE module instance parameters must be specified correctly to use the component. See Appendix B for a list of these parameters. In most cases, the DAVE module instance parameters will not need to be changed by the user.

The “DAVE Database Selector” window (see image below) is the first window that opens when DAVE has been executed through a TRIM.Risk MIMS project, or when it has been opened by clicking on the DAVE executable batch file. In the database selector window, the user can select an output database, and select either the “Analyze” or “Export” buttons. The “Export” button will provide a prompt to save the HQ results from the ecological risk assessment to a delimited file which can be opened in a spreadsheet or other analysis program. Clicking the “Analyze” button will allow the user to make selections to create graphs or tables from specified HQ results.



See the DAVE User's Guide, which is available on the TRIM.Risk page of the EPA FERA website (http://www.epa.gov/ttn/fera) and downloadable with the TRIM installation package, for detailed instructions. In addition, some basic instructions are provided in Volume I, Section 3.3. of this User's Guide.

[This page intentionally left blank.]

5. References

U.S. Environmental Protection Agency (EPA). 1998. Guidelines for Ecological Risk Assessment. Risk Assessment Forum, Washington, DC. EPA/630/R095/002F. Available online at: http://cfpub.epa.gov/ncea/cfm/recordisplay.cfm?deid=12460.

U.S. Environmental Protection Agency (EPA). 1992. Framework for Ecological Risk Assessment. Risk Assessment Forum, Washington, DC. EPA/630/R-92/001. Available online at: http://cfpub.epa.gov/ncea/cfm/recordisplay.cfm?deid=30759.

[This page intentionally left blank.]

Appendix A
Data Flow Diagrams

Included in this appendix are data flow diagrams for the TRIM.Risk ecological risk assessment project. These diagrams articulate the sequence in which components are run and indicate key databases used. Two diagrams are presented for the project: a simplified figure that demonstrates the overall process and includes key TRIM components but omits more detailed aspects of the process, including interim databases created during a run and the presence of the graphical user interface (GUI) that allow for user input; and a more comprehensive diagram that provides additional detail on the process flow and the databases that are created and accessed by each component.

Exhibit A-1 presents the simplified sequence of components. Below is a brief description of each component.

TRIM.FaTE is the ecological exposure model included in the project. It can be used to generate ecological exposure estimates (oral intake, body burden, and media concentrations) from a simulation of the multimedia fate and transport of a pollutant in a defined study area.

TRIM.Risk$_{Eco}$ uses ecological exposure estimates, an ecological toxicity database, and user selections input via a GUI to generate ecological risk estimates.

DAVE uses the risk estimates generated by TRIM.Risk$_{Eco}$ to create charts, tables, and other results summaries as specified by the user.

A more comprehensive data flow diagram for this project is presented in Exhibit A-2. This figure includes the TRIM components portrayed in Exhibit A-1, but it demonstrates in more detail the interaction and feedback between these components. In addition, this figure indicates the GUI that enables user interaction with the model and specifies the input files and databases used for this project.

Exhibit A-1. Simplified Data Flow Diagram for the "Ecological risk assessment with TRIM.FaTE" TRIM.Risk Project

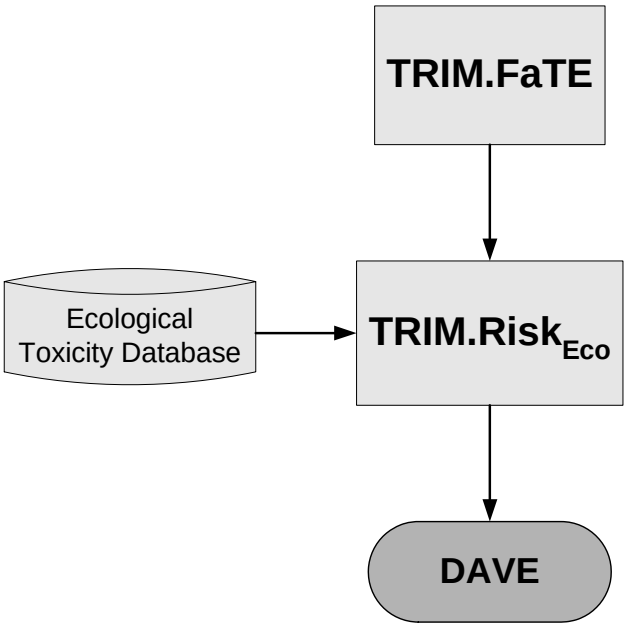
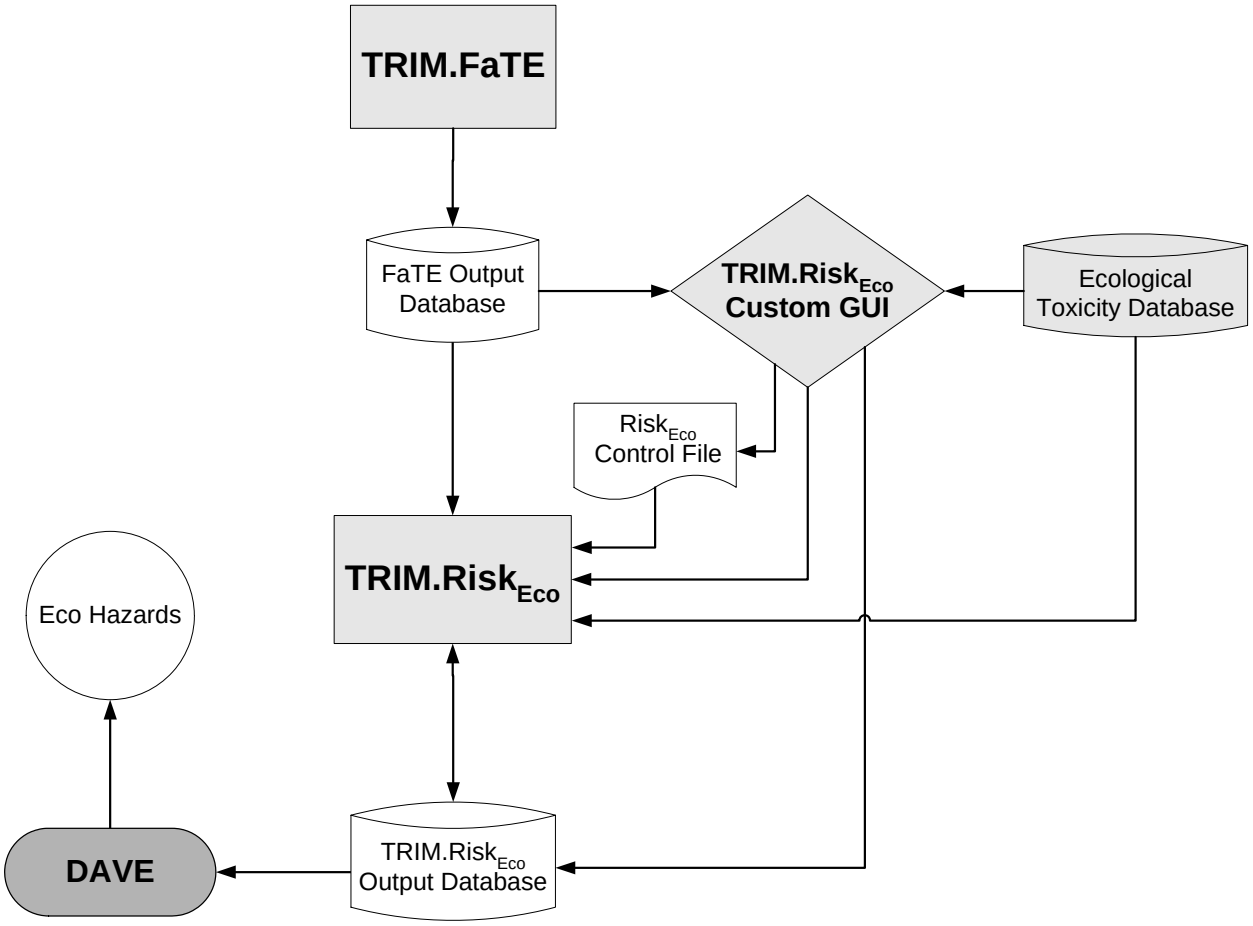


Exhibit A-2. Detailed Data Flow Diagram for the "Ecological risk assessment with TRIM.FaTE" TRIM.Risk Project



[This page intentionally left blank.]

Appendix B
Scenario and Module Instance Parameters

For each MIMS scenario and module instance, there is a set of parameters that specifies the location of model components, executables, log files and databases; details about the scenario or module instance; and options for the user to modify how the run is performed. This appendix provides a complete list of all parameters for the "Ecological risk assessment with TRIM.FaTE" TRIM.Risk project – for each scenario and module instance.

For each of the pre-packaged scenarios available with TRIM.Risk, the scenario and module instance parameters that are frequently changed by users are listed on tabs in the "Input Panels" pane of the scenario window. Lists and instructions for how to change these parameters on the "Input Panels" are provided in Chapter 3. The parameters on the "Input Panels" are included in the tables here because they are also found on the parameter tables within each scenario and module instance. Note that some parameters are in both scenario and module instance parameter tables. Parameters that are in multiple tables are linked so that if the parameter is changed in one table, it will change in all tables where it appears (e.g., if "FaTE Scenario" is changed in the "TRIM.FaTE w. TRIM.Risk" scenario parameter table, it will automatically change in the FaTE module instance table as well).

Scenario Parameters specify the location of model components, executables, and log files as well as parameters that are used for other MIMS processes (note that not all parameters are needed in every application). The full set of parameters are accessed by either selecting "Edit Parameter Values" from the "Scenario" drop-down menu or by checking the box next to "Parameter Table" in the "View" drop-down menu. If a required scenario parameter value is not set before the simulation is executed, there will be an error when the simulation is attempted. See Section 3.1.7 of Volume I of this User's Guide for further instructions on setting scenario parameters. Exhibit B-1 lists each scenario parameter in the "TRIM.FaTE w. TRIM.Risk" scenario, and Exhibit B-2 lists each scenario parameter in the "TRIM.Risk" scenario.

Module Instance Parameters contain information required to run a module through MIMS. The full set of module instance parameters are accessed by double-clicking on the module instance name in the graph view pane of the scenario window (e.g., click on ":FaTE" to open the TRIM.FaTE module instance parameters). The parameter values required to run the module instance within MIMS are specified via the "Parameters" view, which is accessed by clicking on the "Parameters" tab. There are no user inputs or interaction required in the "Module" view, which is accessed by clicking on the "Module" tab. Exhibit B-3 lists each module instance parameter for TRIM.FaTE; Exhibit B-4 lists each module instance parameter for $\text{TRIM.Risk}_{\text{Eco}}$; and Exhibit B-5 lists each module instance parameter for DAVE.

The tables in this appendix include the data type, default value, description, and instructions on how to set each parameter. Note that the parameters are in alphabetical order in these appendix tables, which may not be the same order as in the scenario or module instance.

Setting Parameters Using Environmental Variables

Values for parameters that specify file or directory names and locations can include “environmental variables” that allow the use of values of other parameters. In general, this would follow the pattern of “${str(“Previous Parameter, usually referring to a directory location”)\Specific File.” For example, the “TRIM Data Directory” parameter may have a value of “${str("TRIM Directory")}\data”. In this case, the parameter uses the “TRIM Directory” parameter, plus additional information. For more information on setting environmental variables in MIMS, refer to the Section 3.1.12 of Volume I.

Exhibit B-1. Scenario Parameters for “TRIM.FaTE w. TRIM.Risk” Scenario

“TRIM.FaTE w. TRIM.Risk” Scenario Main Parameters

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Default Directory	Directory name	${str("TRIM Directory")}\logs\FaTE WEcoRisk\${str("Run Name")}	Directory in which TRIM log files will be saved.	User-specified. If a value different from default is required, type directory name and location directly into the “Value” column or use “Browse” button to select the directory from your computer.
Eco Risk DB Name	Database name	–	Name of the output database from $TRIM.Risk_{Eco}$.	User-specified. Type database name directly into the “Value” column or use the “Browse” button to select the database on your computer.
Ecological Effects Database	Database name	–	Name of database with toxicity data.	User-specified. Type database name directly into the “Value” column or use the “Browse” button to select the database on your computer.

Exhibit B-1. Scenario Parameters for "TRIM.FaTE w. TRIM.Risk" Scenario

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Exit after running?	True/False	False	Check box which indicates whether the TRIM.FaTE GUI will close after the module instance execution is complete. If not checked, TRIM.FaTE will have to be closed before TRIM.Risk$_{Eco}$ will begin executing.	User-specified. Can be edited in the "Value" column.
FaTE Database Name	Database name	–	Name of output database from TRIM.FaTE simulation.	User-specified. Type database name directly into the "Value" column or use the "Browse" button to select the database on your computer.
FaTE Project File	Directory and file name	–	Name and location of TRIM.FaTE project file.	User-specified. Type file name and location directly into the "Value" column or use the "Browse" button to select the file on your computer.
FaTE Scenario	Text string	–	Name of TRIM.FaTE scenario.	User-specified. Type name of scenario in column.
File Management Parameters	Set of parameters	Set of 14	This parameter provides a link to a list of 14 File Management Parameters, none of which need to be set by the user.	Click the "Edit" button on the row containing the parameter (or click on the expand button). A Parameter Value window containing 14 parameters will appear, see Table below.
MySQL Password	Text string	Set by user during installation	Password used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. Type a value directly into the "Value" column.
MySQL User Name	Text string	Set by user during installation	User name used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. Type a value directly into the "Value" column.

Exhibit B-1. Scenario Parameters for "TRIM.FaTE w. TRIM.Risk" Scenario

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Properties File	Directory and file name	–	Properties import file for the TRIM.FaTE model run. This file could be used to specify a revised name for the TRIM.FaTE property outputDatabaseName if "Use Extra Properties for Run?" is set to true. The same name must be specified in the field "FaTE Database Name." Currently, a file name must be entered in this field even if "Use Extra Properties for Run?" is false and you do not intend to replace the database name. Enter a file name that is not the same as another file in your "Default Directory" for this scenario, and a file will be created but not used by the scenario.	User-specified. If a value different from default is required, type file name and location directly into the "Value" column or use the "Browse" button to select the file on your computer.
R Bin Directory	Directory location	Set by user during installation	Location of the bin directory for the R program.	User-specified. If a value different from default is required, type directory name and location directly into the "Value" column or use "Browse" button to select the directory from your computer.
Run Name	Text string	–	Name of the unique TRIM.Risk scenario run. The purpose is solely to help users keep track of TRIM.Risk scenarios and results.	User-specified. Type the name in the "Value" column.
Show GUI during execution?	True/False	True	Indicates whether the TRIM.FaTE GUI will be visible while the module instance is executing.	This property allows the user to specify whether the TRIM.FaTE GUI will open when the module is running.

Exhibit B-1. Scenario Parameters for "TRIM.FaTE w. TRIM.Risk" Scenario

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
TRIM Data Directory	Directory name	${str("TRIM Directory")}\data	Directory in which the TRIM results are stored.	Specified from the TRIM directory parameter by default. If a value different from default is required, type directory name and location directly into the "Value" column or use "Browse" button to select the directory from your computer.
TRIM Directory	Directory name	Set by user during installation	TRIM installation directory (i.e., the directory into which the TRIM.Risk installation packages were unzipped).	User-specified. If a value different from default is required, type directory name and location directly into the "Value" column or use "Browse" button to select the directory from your computer.
Use Extra Properties for Run?	True/False	False	Indicates whether an **optional** properties import file will be used to modify the TRIM.FaTE scenario before running it. If true, the properties file name must be specified by the Properties File parameter. Currently, the only TRIM.FaTE property that can be modified using this file is the outputDatabaseName.	User-specified. To select this option, click the check box in the "Value" column.

[a] See text box on page B-2 about Environmental Variables; this and other parameters with the format ${str("parameter")}\file are using Environmental Variables to set the value.

Exhibit B-1. Scenario Parameters for "TRIM.FaTE w. TRIM.Risk" Scenario

"TRIM.FaTE w. TRIM.Risk" Scenario File Management Parameters

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Create Output Directories	True/False	True	No user interaction is required for these parameters. They are set correctly in the TRIM.Risk projects, or are not used at all.	
Delete Generated Batch Files	True/False	False		
Delete Generated Input Files	True/False	False		
Delete Intermediate Files When Done	True/False	False		
Delete Old Output Files	True/False	False		
Delete Unused Final Files	True/False	False		
Execution Method	Text string	Local Execution		
Execution Path	Directory name			
Keep Module Logs	Pick list	Yes		
Log Module Start and Finish	True/False	True		
Overwrite Log Files	True/False	False		
Overwrite Script Files	True/False	False		
Random Number Seed	Integer			
Saving Frequency	Integer	0		

Exhibit B-2. Scenario Parameters for "TRIM.Risk" Scenario

"TRIM.Risk" Scenario Main Parameters

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Default Directory	Directory name	${str("TRIM Directory")}\logs\EcoRisk\${str("Run Name")}[a]	Directory in which TRIM log files will be saved.	User-specified. If a value different from default is required, type directory name and location directly into the "Value" column or use "Browse" button to select the directory from your computer.
Eco Risk DB Name	Database name	–	Name of the output database from $TRIM.Risk_{Eco}$.	User-specified. Type database name directly into the "Value" column or use the "Browse" button to select the database on your computer.
Ecological Effects Database	Database name	–	Name of database with toxicity data.	User-specified. Type database name directly into the "Value" column or use the "Browse" button to select the database on your computer.
Fate Output Database	Database name	–	Name of output database from TRIM.FaTE simulation.	User-specified. Type database name directly into the "Value" column or use the "Browse" button to select the database on your computer.
File Management Parameters	Set of parameters	Set of 14	This parameter provides a link to a list of 14 File Management Parameters, none of which need to be set by the user.	Click the "Edit" button on the row containing the parameter (or click on the expand button). A Parameter Value window containing 14 parameters will appear, see table below.
MySQL Password	Text string	Set by user during installation	Password used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a different value is required, type the password in the "Value" column.

Exhibit B-2. Scenario Parameters for "TRIM.Risk" Scenario

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
MySQL User Name	Text string	Set by user during installation	User name used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a different value is required, type the name in the "Value" column.
R Bin Directory	Directory location	Set by user during installation	Location of the bin directory for the R program.	User-specified. If a value different from default is required, type directory name and location directly into the "Value" column or use "Browse" button to select the directory from your computer.
Run Name	Text string	–	Name of the unique TRIM.Risk scenario run. The purpose is solely to help users keep track of TRIM.Risk scenarios and results.	User-specified. Type the name in the "Value" column.
TRIM Data Directory	Directory name	${str("TRIM Directory")}\data	Directory in which the TRIM results are stored.	Specified from the TRIM directory parameter by default. If a value different from default is required, type directory name and location directly into the "Value" column or use "Browse" button to select the directory from your computer.
TRIM Directory	Directory name	Set by user during installation	TRIM installation directory (i.e., the directory into which the TRIM.Risk installation packages were unzipped).	User-specified. If a value different from default is required, type directory name and location directly into the "Value" column or use "Browse" button to select the directory from your computer.

[a] See text box on page B-2 about Environmental Variables; this and other parameters with the format ${str("parameter")}\file are using Environmental Variables to set the value.

Exhibit B-2. Scenario Parameters for "TRIM.Risk" Scenario

"TRIM.Risk" Scenario File Management Parameters

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Create Output Directories	True/False	True	No user interaction is required for these parameters. They are set correctly in the TRIM.Risk projects, or are not used at all.	
Delete Generated Batch Files	True/False	False		
Delete Generated Input Files	True/False	False		
Delete Intermediate Files When Done	True/False	False		
Delete Old Output Files	True/False	False		
Delete Unused Final Files	True/False	False		
Execution Method	Text string	Local Execution		
Execution Path	Directory name			
Keep Module Logs	Pick list	Yes		
Log Module Start and Finish	True/False	True		
Overwrite Log Files	True/False	False		
Overwrite Script Files	True/False	False		
Random Number Seed	Integer			
Saving Frequency	Integer	0		

Exhibit B-3. Module Instance Parameters for TRIM.FaTE

"TRIM.FaTE" Module Instance Main Parameters

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Exit after running?	True/False	False	Check box which indicates whether the TRIM.FaTE GUI will close after the module instance execution is complete. If not checked, TRIM.FaTE will have to be closed before TRIM.Risk$_{Eco}$ will begin executing.	User-specified. Can be edited in the "Value" column.
FaTE Executable Settings	Set of parameters	Set of 6	This parameter provides a link to a list of six parameters needed to run TRIM.FaTE.	Click the "Edit" button on the row containing the parameter. A Parameter Value window containing six parameters will appear. Provide values for these parameters as described below, if necessary.
FaTE Project File	Directory and file name	–	Name and location of the TRIM.FaTE project file.	User-specified. Type file name and location directly into the "Value" column or use the "Browse" button to select the file on your computer.
FaTE Scenario	Text string	–	Name of the TRIM.FaTE scenario.	User-specified. Type name of scenario in column.
FaTE Output Database	Database name	–	Name of output database from TRIM.FaTE simulation.	User-specified. Type database name and location directly into the "Value" column or use the "Browse" button to select the file on your computer.
MySQL Password	Text string	Set by user during installation	Password used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a value different from default is required, type the password in the "Value" column.
MySQL User Name	Text string	Set by user during installation	User name used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a value different from default is required, type the password in the "Value" column.

Exhibit B-3. Module Instance Parameters for TRIM.FaTE

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Preferences Path	File directory	–	Not required for this module instance.	No user interaction required.
Properties File	File name and location	–	Properties import file for the TRIM.FaTE model run. This file could be used to specify a revised name for the TRIM.FaTE property outputDatabaseName if "Use Extra Properties for Run?" is set to true. The same name must be specified in the field "FaTE Database Name." Currently, a file name must be entered in this field even if "Use Extra Properties for Run?" is false and you do not intend to replace the database name. Enter a file name that is not the same as another file in your "Default Directory" for this scenario, and a file will be created but not used by the scenario.	User-specified. If a value different from default is required, enter file name and location directly into the "Value" column. Alternatively, click "Browse" button to search your computer for the file, select the file in the Browse window, and click "Select."
Show GUI during execution?	True/False	True	This property specifies whether the TRIM.FaTE GUI will open when the module is running.	User-specified. To select this option, click the check box in the "Value" column.
Use Extra Properties for Run?	True/False	False	Indicates whether an **optional** properties import file will be used to modify the TRIM.FaTE scenario before running it. If true, the properties file name must be specified by the Properties File parameter. Currently, the only TRIM.FaTE property that can be modified using this file is the outputDatabaseName.	User-specified. To select this option, click the check box in the "Value" column.

Exhibit B-3. Module Instance Parameters for TRIM.FaTE

"TRIM.FaTE" Module Instance FaTE Executable Settings

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
CLASSPATH	Text string	See Footnote[a]	Specifies the classpath required to run TRIM.FaTE.	By default, this parameter is correctly configured in the scenario; therefore, no user input is required.
FaTE Executable	File name/location	${str("FaTE Executables Directory")}\trimfate.jar[b]	Name and location of the TRIM.FaTE executable JAR (Java archive) file. Typically a JAR file will contain the class files and auxiliary resources associated with applets and applications bundled into a single archive.	By default, this parameter is correctly configured in the scenario for the current version of TRIM.FaTE. If your version of TRIM.FaTE is updated after installing TRIM.Risk$_{Eco}$, the value of this parameter should be updated to correspond to the name and location of the new TRIM.FaTE executable JAR (Java archive) file.
FaTE Executables Directory	Directory name and location	${str("TRIM Directory")}\lib[b]	Directory containing the TRIM.FaTE executable JAR (Java archive) file.	By default, this parameter is correctly configured in the scenario; therefore, no user input is required.
Memory	Text string	-Xmx400m	Specifies the number of megabytes (MB) of memory (i.e., RAM) available for running TRIM.FaTE.	This parameter can be edited in the "Value" column. By default, it is set to "400", which should be sufficient for computers with at least 512 MB RAM. For computers with more than 512 MB RAM, users are unlikely to see benefits from increasing this value to larger than 400. For computers with less than 512 MB RAM, this value should be set to a number less than the total RAM on your computer. For example, users with 256MB RAM on their computers can safely set this parameter to "200".

Exhibit B-3. Module Instance Parameters for TRIM.FaTE

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Properties Option				No user interaction required.
Run Option	Text string	${ifElse("Exit after running?", "-runthenexit true", "-run true")}	Internal command for running TRIM.FaTE based on values set for other parameters.	No user interaction is required.

[a] ${str("FaTE Executable")};${str("FaTE Executables Directory")}\mims.jar;${str("FaTE Executables Directory")}\colt.jar;${str("FaTE Executables Directory")}\jgl3.1.0.jar;${str("FaTE Executables Directory")}\mparse.zip;${str("FaTE Executables Directory")}\junit.jar;${str("FaTE Executables Directory")}\jcchart.jar;${str("FaTE Executables Directory")}\metamatautils.zip;${str("FaTE Executables Directory")}\jbcl.jar;${str("FaTE Executables Directory")}\dx.jar;${str("FaTE Executables Directory")}\beandt.jar;${str("FaTE Executables Directory")}\jloox20.jar;${str("FaTE Executables Directory")}\jlm20.jar;${str("FaTE Executables Directory")}\jlayout20.jar;${str("FaTE Executables Directory")}\lxgis20.jar;${str("FaTE Executables Directory")}\or124.jar;${str("FaTE Executables Directory")}\drasys.jar;${str("FaTE Executables Directory")}\DistLib.jar;${str("FaTE Executables Directory")}\koml.zip;${str("FaTE Executables Directory")}\connectorJ.jar

[b] See text box on page B-2 about Environmental Variables; this and other parameters with the format ${str("parameter")}\file are using Environmental Variables to set the value.

Exhibit B-4. TRIM.Risk$_{Eco}$ Module Instance Parameters

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
Control File	Name and location	${str("Default Directory")}\riskeco_control.txt[a]	File name of the control file.	By default, this parameter is correctly configured in the scenario; therefore, no user input is required.
Eco Risk Output Database	Text string	–	Name of the Ecological Risk output database created by the Risk-Eco GUI.	Set this parameter by clicking the "Set Name" button and typing a name or clicking the "Browse" button and browsing to a database.
Eco Risk DB Name	Text string	–		
Fate Output Database	Name and location	–	File name of the TRIM.FaTE output database file.	User-specified. Type database name directly into the "Value" column or use the "Browse" button to select the database on your computer.
Ecological Effects Database	Name	–	Name of database with toxicity data.	User-specified. Type database name directly into the "Value" column or use the "Browse" button to select the database on your computer.
MySQL User Name	Text string	Set by user during installation	User name used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a value different from default is required, type the name in the "Value" column.
MySQL Password	Text string	Set by user during installation	Password used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a value different from default is required, type the password in the "Value" column.
PATH	File directory	%PATH%;${str("TRIM Directory")}\dlls\Win32\Winnt[a]		By default, this parameter will be set correctly. No user interaction required.

[a] See text box on page B-2 about Environmental Variables; this and other parameters with the format ${str("parameter")}\file are using Environmental Variables to set the value.

Exhibit B-5. DAVE Module Instance Parameters

Parameter	Data Type	Default Value	Description	How Parameter Value is Set
DAVE Executable	Directory and Name	${str("TRIM Directory")}\lib\dave.jar	This property provides the location of the DAVE executable.	No user interaction required; value set for the default TRIM installation structure.
Database Name	Text string	${ifElse(str("TRIM Database Type") == "Ecological Risk",str("Eco Risk Output Database"),"null")}	Name of the Ecological Risk output database created by TRIM.Risk$_{Eco}$.	No user interaction; name is set through the next parameter, Eco Risk Output Database (which is set from TRIM.Risk$_{Eco}$).
Eco Risk Output Database	Text string	–	Name of the Ecological Risk output database created by TRIM.Risk$_{Eco}$.	This parameter is filled in through the TRIM.Risk$_{Eco}$ GUI. No user interaction is required in the parameter table.
MySQL User Name	Text string	Set by user during installation	User name used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a value different from default is required, type the name in the “Value” column.
MySQL Password	Text string	Set by user during installation	Password used for access to the MySQL databases created or used for the TRIM.Risk projects.	User-specified. If a value different from default is required, type the password in the “Value” column.
R Bin Directory	Directory location	Set by user during installation	Location of the bin directory for the R program.	User-specified. If a value different from default is required, type directory name and location directly into the “Value” column or use “Browse” button to select the directory from your computer.
TRIM Database Type	Text string	Ecological Risk	This defines the database type for DAVE.	Database type is set correctly for this project; no user interaction required.

[a] See text box on page B-1 about Environmental Variables; this and other parameters with the format ${str("parameter")}\file are using Environmental Variables to set the value.

[This page intentionally left blank.]

Appendix C
MySQL Database Schema

There are three MySQL databases that are used in the "Ecological risk assessment with TRIM.FaTE" TRIM.Risk project. The databases are:

- **TRIM.FaTE output database** – created by running TRIM.FaTE and exporting ecological exposure results to a database;

- **Ecological toxicity database** – contains ecological toxicity values for multiple Hazardous Air Pollutants. The most recent EPA version can be downloaded via the EPA FERA website (http://www.epa.gov/ttn/fera); and

- **TRIM.Risk$_{Eco}$ output database** – created by TRIM.Risk$_{Eco}$ from GUI inputs and calculations.

The database schemas are presented in this appendix for these three databases. The schema include a list of the tables in the databases and the fields in each table. For each field, the data type, units, whether it is a key, if it cannot be null, and a description are provided. For additional information on the TRIM.FaTE database, see the TRIM.FaTE documentation cited in Appendix F.1; for information on the Ecological Toxicity database, see documentation on the TRIM.Risk EPA website; and for information on TRIM.Risk$_{Eco}$, see Appendix F.2.

Exhibit C-1. TRIM.FaTE Output Database Schema *(current as of August 2005)*

TABLE: Results

Field	Data Type	Key	Not Null	Units	Description
RunID	integer			unitless	The run identification number for these results.
Time	date/time			unitless	The time at which the data on this line apply.
VarID	integer	foreign		unitless	The variable ID for this value.
Value	double			varies	The value for the given parameter.

TABLE: Variables

Field	Data Type	Key	Not Null	Units	Description
VarID	integer	primary		unitless	The name of the column in the Results table where this variable is stored.
ChemID	integer	foreign		unitless	The unique ID of the chemical for this variable.
VEID	integer	foreign		unitless	The volume element for this variable.
CompID	integer	foreign		unitless	The compartment for this variable.
Units	varchar(30)			unitless	The units for this variable (g/g, g/m3, g/L etc.).
Type	varchar(20)			unitless	The type of variable (conc, moles, mass).
LookupName	varchar(30)			unitless	The name that downstream pieces of TRIM will use to find data. (SurfSoil, T3fish, T4fish, mapped_air, wet_part_dep, dry_part_dep, wet_vap_dep, dry_vap_dep, wet_surfsoil, root_soil, diss_surf_water, tot_surf_water,dose_<biota ID>, body_burden_<biota ID>, sediment)
Duration	integer			hours	The duration in hours for this variable.
Avg_Inst	varchar(4)			unitless	Either "avg" if this is an averaged value,"inst" if instantaneous, or “ss” if it is steady-state.

Exhibit C-1. TRIM.FaTE Output Database Schema (*cont.*)

TABLE: Objects					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
ObjID	integer	primary		unitless	The unique identifier for this object in the database.
ObjName	varchar(100)			unitless	The name of the object in TRIM.FaTE.
ObjType	varchar(50)			unitless	The type of TRIM.FaTE object (Compartment, Chemical, etc)
ParentID	integer	foreign		unitless	The volume element that this object resides in.
Descr	varchar(255)			unitless	The description of this object.

TABLE: ObjectProps					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
PropID		primary		unitless	The unique identifier for this object in the database.
ObjID	integer	foreign		unitless	The unique identifier for this object in the database from the Objects table.
PropName	varchar(100)			unitless	The name of the property that this value is for.
DataType	varchar(50)			unitless	The type of value that this property has. (Double, Integer, Date, String, etc.)
ChemID	integer	foreign		unitless	The chemical ID this property applies too (could be null).
Value	varchar(100)			unitless	The value of the property.
Form	varchar(20)			unitless	The form of the property
DateChanged	DateTime			unitless	The date that this property was changed.
Descr	varchar(255)			unitless	The description for this property.

Exhibit C-1. TRIM.FaTE Output Database Schema (*cont.*)

TABLE: Chemicals					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
ChemID	integer	primary		unitless	Unique ID for this chemical.
CAS	varchar(50)			unitless	CAS number for this chemical.
Name	varchar(100)			unitless	Name for this chemical.

TABLE: ChemProps					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
ChemID	integer	Foreign		unitless	The unique ID for this chemical.
PropName	varchar(255)			unitless	Name of the property.
DataType	varchar(20)			unitless	The type of value that this property has (Double, Integer, Date, String, etc.)
Value	varchar(100)			unitless	Value of the property.
Form	varchar(20)			unitless	The form of the property.
DateChanged	DateTime			unitless	The date that this property was last changed.
Descr	varchar(255)			unitless	Description of this property.

Exhibit C-2. Ecological Toxicity Database Schema *(current as of August 2005)*

TABLE: Benchmarks					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
BenchmarkID	integer	primary	X	unitless	A unique numerical identifier for each benchmark.
AbbrevBM	varchar(5)		X	unitless	Text appreviation for the benchmark.
Receptor_ID	integer			unitless	Fixed ID for each receptor in the tbl_Receptor
Receptor_Name	varchar(30)			unitless	Receptor Name from tbl_Receptor
CAS	varchar(20)			unitless	ID number of each chemical in tbl_Chemicals
Chemical_Name	varchar(50)			unitless	Common name of chemical.
Effect	varchar(100)			unitless	Physiological effect on the receptor (e.g., mortality).
Endpoint	varchar(20)			unitless	Endpoint calculated or reported (e.g., NOAEL).
Benchmark_Value	double			varies	Numerical value of the benchmark (e.g., mg/L; mg/kg; mg/kg-d).
Test Species	varchar(20)			unitless	Test species reported in the study (e.g., rat, fathead minnow).
Exposure_Duration	varchar(30)			varies	Duration of exposure reported in the study (e.g., 7 day).
Chemical_Form	varchar(20)			unitless	Form of chemical reported in the study (e.g., arsenate).
Distribution_Type	varchar(20)			unitless	Statistical distribution for benchmark (e.g., Lognormal)
Dist_Parameter1	double			unitless	Statistical descriptor for distribution (e.g., mean, mode)
Dist_Parameter2	double			unitless	Statistical descriptor for distribution (e.g., mean, mode)
Dist_Parameter3	double			unitless	Statistical descriptor for distribution (e.g., mean, mode)
Dist_Parameter4	double			unitless	Statistical descriptor for distribution (e.g., mean, mode)
Dist_Parameter5	double			unitless	Statistical descriptor for distribution (e.g., mean, mode)
Doc_Reference	varchar(20)			unitless	Short version of the reference for the benchmarks (e.g., Suter and Tsao, 1996).

Exhibit C-2. Ecological Toxicity Database Schema (*cont.*)

TABLE: Benchmarks *(cont)*					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Reference_Type	varchar(20)			unitless	Indicates whether data were taken from a primary or secondary source.
Ref_ID	integer			unitless	Unique numerical identifier for reference.
Units	varchar(20)			unitless	Units for benchmark (e.g., mg/L; mg chem/kg BW/day).
TRIMUnits	varchar(20)			varies	Benchmark units that are used by TRIM (e.g., mg/kg-day)
IsRecommended Value	tinyint(1)			unitless	Designates whether a value is recommended by EPA for the default analysis option.
Comment	blob			unitless	Comment provides additional information on the benchmark (e.g., information on the test species used for the scaling equation).

TABLE: Chemicals					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
CAS	varchar(20)	primary	X	unitless	Chemical abstract number.
Chemical_Name	varchar(30)			unitless	Common name of chemical.
Log_Kow	double			unitless	Base 10 log of the octanol-water partition value.
Koc	double			L pore water/kg organic carbon	Organic carbon adsorption coefficient calculated for organics as 10^(0.00028+0.983*Log_Kow)); Koc is used to calculated the sediment concentrations for organic chemicals.

Exhibit C-2. Ecological Toxicity Database Schema (*cont.*)

TABLE: Receptor					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Receptor_id	integer	primary	X	unitless	Unique numerical identifier for each receptor.
Receptor_Name	varchar(30)			unitless	Common name for receptor (e.g., Red Tailed Hawk).
Receptor_TrimFate Name	varchar(40)			unitless	Name used by the TRIM.Fate model for the ecological receptor (e.g., hawk).
Receptor_Scientific Name	varchar(40)			unitless	Scientific name for the receptor (e.g., Buteo jamaicensis).
Receptor_Group	varchar(20)			unitless	Taxa group used to match receptors (e.g., mammal, bird).
BW_Female_g	double			kg	Body weight for female of receptor species.
BW_Male_g	double			kg	Body weight for male of receptor species.
BW_Both_g	double			kg	Average body weight for male and female of receptor species.

TABLE: Database Date					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Recnum	integer	primary	X	unitless	Unique ID.
Db_date	date		X	unitless	Time date stamp.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(current as of August 2005)*

TABLE: input_databases ***(created based on inputs to the Risk-Eco GUI)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
db_type	varchar			unitless	Database type input to Risk-Eco GUI.
db_name	varchar			unitless	Database name entered in GUI.

TABLE: tblER_VE ***(created based on inputs to the Risk-Eco GUI)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
VEID	integer			unitless	Volume element ID number from TRIM.FaTE.
VEName	varchar(100)			unitless	Volume element name (if available).
CompID	integer			unitless	Compartment ID that defines the TRIM.FaTE data that the benchmark will be run against.
PAC	varchar(50)			unitless	Primary abiotic compartment name for the volume element from TRIM.FaTE

TABLE: tblER_Comp ***(created based on inputs to the Risk-Eco GUI)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
BenchmarkID	integer			unitless	Benchmark ID number from Ecological Toxicity database or user-defined.
AbbrevBM	varchar(5)			unitless	Benchmark type abbreviation. The combination of BenchmarkID and AbbrevBM should be unique.
CompID	integer			unitless	Compartment ID that defines the TRIM.FaTE data that the benchmark will be run against.
ReceptorID	integer			unitless	Receptor ID number from Ecological Toxicity database that corresponds to the benchmark.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: tblER_Benchmarks ***(created based on inputs to the Risk-Eco GUI)***

Field	Data Type	Key	Not Null	Units	Description
BenchmarkID	integer			unitless	The ID number for the benchmark, from the Ecological Toxicity database.
AbbrevBM	varchar(5)			unitless	Benchmark type abbreviation. The combination of BenchmarkID and AbbrevBM should be unique.
BenchmarkName	varchar(50)			unitless	The name for the type of benchmark (Soil concentration, Sediment concentration, Water concentration - total, Water concentration - dissolved, Oral Intake, Body burden)
Receptor_ID	integer			unitless	Receptor ID number from Ecological Toxicity database or user-defined.
CAS	varchar(20)			unitless	Chemical CAS number.
Effect	varchar(50)			unitless	Benchmark effect from the Ecological Toxicity database or user-defined.
EndPoint	varchar(50)			unitless	Benchmark endpoint from the Ecological Toxicity database or user-defined.
Benchmark	double			varies	The value for the benchmark from the Ecological Toxicity database or user-defined (used in deterministic mode calculations).
Distribution_Type	varchar(50)			unitless	The type of distribution for the benchmark (text description).
Dist_Parameter_1	double			unitless	The first distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_2	double			unitless	The second distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_3	double			unitless	The third distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_4	double			unitless	The fourth distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_5	double			unitless	The fifth distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: tblER_Benchmarks ***(cont)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Units	varchar(50)			varies	The units for the benchmark from the Ecological Toxicity database.
IsUserDefined	tinyint(1)			unitless	A logical flag indicating whether the benchmark value used in the calculation was defined by the user (true) or was selected from the Ecological Toxicity database (false).

TABLE: tblER_Timesteps ***(created based on inputs to the Risk-Eco GUI)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
AbbrevBM	varchar(10)	primary		unitless	Benchmark abbreviation.
TimeStepType	varchar(20)			unitless	Type of timestep chosen by user (all, max, last, selected, or single).
BeginTimeStep	datetime			unitless	Beginning timestep time/date.
EndTimeStep	datetime			unitless	Ending timestep time/date.
TimeStepFreq	integer			unitless	An integer defining the user-selected time step sampling frequency when the user chooses "Selected Time Steps".
Duration	integer			hours	Duration of the timestep (set to zero for steady-state results).
TimeBasis	varchar(4)			unitless	Indicator of whether the data value is average, instantaneous, or steady-state.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: tblgui_comp_rec_chem_bm ***(table combining inputs to the Risk-Eco GUI)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
RecordID	integer	Primary	X	unitless	ID number for unique match.
BenchmarkID	integer			unitless	The ID number for the benchmark, from the Ecological Toxicity database.
AbbrevBM	varchar(5)			unitless	Benchmark type abbreviation. The combination of BenchmarkID and AbbrevBM should be unique.
CompID	integer			unitless	Compartment ID that defines the TRIM.FaTE data that the benchmark will be run against.
CompName	varchar(50)			unitless	TRIM.FaTE Compartment name.
IsPAC	boolean			unitless	Defines whether the compartment is the primary abiotic compartment in the TRIM.FaTE volume element.
Category	varchar(50)			unitless	Hierarchical TRIM.FaTE classification for the compartment.
Taxon	varchar(30)			unitless	If TRIM.FaTE compartment is an animal, this category defines the Taxon (or the highest-level in the category field), otherwise the value is “none.”
VEID	integer			unitless	Volume element ID number from TRIM.FaTE.
VEName	varchar(100)			unitless	Volume element name.
RecID	integer			unitless	Unique numerical identifier for receptor from Ecological Toxicity Database.
RecName	varchar(30)			unitless	Common name for receptor from Ecological Toxicity Database.
ChemID	integer			unitless	The unique ID for this chemical used in TRIM.FaTE.
ChemName	varchar(30)			unitless	Common name of chemical.
IsExactMatch	boolean			unitless	Indicates whether the CompName from TRIM.FaTE and RecName from the Ecological Toxicity Database match for the benchmark.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: tblgui_comp_rec_chem_bm *(cont)*					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
IsMatchByTaxon	boolean			unitless	Indicates whether the match between the TRIM.FaTE compartment and the benchmark is based on the Taxon category instead of the exact name of the receptor and compartment.
IsMatchByMedia	boolean			unitless	Indicates whether match between TRIM.FaTE compartment and benchmark is based on the abotic media type of the TRIM.FaTE compartment.
IsShowable	boolean			unitless	Indicates whether the match will show in the Eco GUI.
IsSelected	boolean			unitless	Indicates whether the match was selected by the user in the Eco GUI.

TABLE: tblgui_ve_comp *(table combining inputs to the Risk-Eco GUI)*					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
VEID	integer			unitless	Volume element ID number from TRIM.FaTE.
VEName	varchar(100)			unitless	Volume element name.
CompID	integer			unitless	Compartment ID that defines the TRIM.FaTE data that the benchmark will be run against.
CompName	varchar(50)			unitless	TRIM.FaTE compartment name.
IsPAC	boolean			unitless	Defines whether the compartment is the primary abiotic compartment in the TRIM.FaTE volume element.
IsSelected	boolean			unitless	Indicates whether the match was selected by the user in the Eco GUI.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: Times ***(output from Risk-Eco run)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
recnum	smallint(5) unsigned	primary		unitless	Table record number.
BegTime	datetime			unitless	Time that the run began.
EndTime	datetime			unitless	Time that the run ended.

TABLE: tblER_VENames ***(output from Risk-Eco run)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
VEID	integer	primary		unitless	TRIM.FaTE volume element ID number.
VEName	varchar(50)			unitless	Volume element name from TRIM.FaTE.
PAC	varchar(50)			unitless	Primary abiotic compartment name from TRIM.FaTE.

TABLE: tblER_CompNames ***(output from Risk-Eco run)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
CompID	integer	primary		unitless	TRIM.FaTE compartment ID number.
CompName	varchar(50)			unitless	Compartment name from TRIM.FaTE.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: tblER_ReceptorNames ***(output from Risk-Eco run)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Receptor_ID	integer	primary		unitless	TRIM.FaTE Receptor ID number.
Receptor_Name	varchar(30)			unitless	Receptor name from Eco Effects DB.
Receptor_Group	varchar(20)			unitless	Receptor group name from Eco Effects DB.

TABLE: tblER_ChemicalNames ***(output from Risk-Eco run)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
ChemID	integer	primary		unitless	Chemical ID number from TRIM.FaTE.
CAS	varchar(50)			unitless	TRIM.FaTE CAS number.
Name	varchar(100)			unitless	Chemical name from TRIM.FaTE.

TABLE: tblER_BMNames ***(output from Risk-Eco run)***					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Benchmark	double			varies	The value for the benchmark used in the calculation.
BenchmarkName	varchar(50)			unitless	The name of the benchmark used in the calculation (e.g., body burden, dose, air concentration, etc.)
BenchmarkID	integer	primary		unitless	The benchmark ID number from the Eco Effects database (or user-defined).
Units	varchar(50)			unitless	The units for the benchmark.
AbbrevBM	varchar(5)	primary		unitless	The abbreviation for the benchmark from the Eco Effects database. A combination of the benchmark abbreviation and the benchmark ID number make a unique index for identifying the information in the Ecological Toxicity database associated with the benchmark.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: tblER_BMNames *(cont)*					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Endpoint	varchar(50)			unitless	The text description of the end point for the selected benchmark from the Ecological Toxicity database.
Effect	varchar(50)			unitless	The text description of the effects for the selected benchmark from the Ecological Toxicity database.
IsUserDefined	boolean			unitless	A logical flag indicating whether the benchmark value used in the calculation was defined by the user (true) or was selected from the Ecological Toxicity database (false).
Dist_Parameter_1	double			unitless	The first distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_2	double			unitless	The second distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_3	double			unitless	The third distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_4	double			unitless	The fourth distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).
Dist_Parameter_5	double			unitless	The fifth distribution parameter for the benchmark from the Eco Effects database or user-defined (used in probabilistic analysis mode, Version 1.1).

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: Results ***(output from Risk-Eco run)***

Field	Data Type	Key	Not Null	Units	Description
recnum	integer	primary	X	unitless	Result record number.
RunID	bigint(20) unsigned			unitless	Run ID number from TRIM.FaTE results table.
VEID	integer			unitless	TRIM.FaTE volume element ID number.
CompID	integer			unitless	Compartment ID number from TRIM database
Receptor_ID	integer			unitless	Eco Effects database Receptor ID number.
CAS	varchar(50)			unitless	The text representation of the chemical's chemical abstracts registry number.
Time	datetime			unitless	The time step associated with the result.
AbbrevBM	varchar(5)			unitless	Abbreviation for the benchmark type
BenchmarkID	integer			unitless	The benchmark ID number from the Eco Effects database.
TimeBasis	varchar(4)			unitless	Indicator of whether the data value is average, instantaneous, or steady-state.
HQ	double			unitless	The HQ value calculated by the ER executable.

TABLE: Control ***(output from Risk-Eco run)***

Field	Data Type	Key	Not Null	Units	Description
control_info	varchar(50)	primary		unitless	Control file parameter.
control_value	varchar(20)			unitless	Control file parameter value.

Exhibit C-3. TRIM.Risk$_{Eco}$ Output Database Schema *(cont)*

TABLE: Selected Times [This is a temporary table used to manage the timestep selection by the EcoRisk model]					
Field	**Data Type**	**Key**	**Not Null**	**Units**	**Description**
Time	datetime			unitless	Timestep time

[This page intentionally left blank.]

Appendix D
Troubleshooting

This appendix includes troubleshooting tips for common problems encountered using the "Ecological risk assessment with TRIM.FaTE" TRIM.Risk project. Some of the tips describe known bugs. Permanent fixes to these bugs will be incorporated into a later version of TRIM.Risk, but for now these tips will aid users in getting around the bug. This appendix is separated into three sections: General Scenario Troubleshooting, TRIM.Risk$_{Eco}$ Troubleshooting, and DAVE Troubleshooting.

D.1 General Scenario Troubleshooting

Why did my scenario fail when I tried to run it?

Information to help you diagnose the reason a run failed is available in the logs produced during the simulation. The logs may include information such as a property or setting that needs modification, the name of a missing input file, or the process that was running when the scenario failed. In many cases, the error will be associated with a specific module instance. To view errors associated with a module instance, right-click on the module instance that failed, evidenced by a red X (⊗) and select one or more of the following options:

- View Run Script, which opens a text editor showing the batch script files for the MIMS module instance;
- View Output Log, which opens a text editor showing the output logs for the MIMS module instance;
- View Error Log, which opens a text editor showing the error log for the MIMS module instance (the module instance error log provides a detailed message regarding the specific error encountered); and/or
- View Error Message, which re-opens the initial pop-up error message generated when the module failed.

To view errors associated with running the scenario, select one or more of the following options presented under the "View Systems Logs..." menu of the "File" menu:

- Error Log, which opens a text editor showing the error log for the MIMS scenario (the scenario error log provides a more detailed message regarding the specific error encountered); and/or
- Standard Log, which opens a text editor showing status messages, including the start and completion time of simulations.

Under the "Scenario" menu, select:

- View Execution Log, which opens the file that recorded starts and completions of the module instance(s).

One or more of these logs may contain useful information about the problem causing the scenario to fail.

I was able to execute my scenario successfully before, why did it fail when I tried to run it a second time?

To re-run a previously executed scenario, you must select "Reset All" under the "Scenario" menu. This returns all processes that have been executed to a ready-to-execute state. Individual module instances can be reset by highlighting and right-clicking on the module instance in the "Graph View" pane and selecting "Reset" from the drop-down menu. Also, to clear all previous selections made in the TRIM.Risk$_{Eco}$ GUI, reload the GUI by right-clicking on the module instance and selecting "Reload Custom GUI" from the drop-down menu before re-running the scenario.

D.2 TRIM.Risk$_{Eco}$ Troubleshooting

The TRIM.Risk$_{Eco}$ GUI window disappeared from my screen.

If you open another program in front of the TRIM.Risk$_{Eco}$ GUI window, you have to minimize the other program in order to view the GUI again. You cannot "maximize" the GUI window from the Windows Taskbar.

I selected receptors for analyses with oral intake using option 3b "Select Other Matches" on Tab 3, but none of these selected receptors show up on the Toxicity Values table accessed through Tab 8.

This is a known bug with the TRIM.Risk$_{Eco}$ GUI. After selecting receptors using option 3b on Tab 3, it is required that the user review the selections by clicking on the "Review" button at the bottom of Tab 3 before clicking the "Done" button. Return to Tab 3 of the GUI, click the "Review" button, and then proceed through the GUI making your other selections again.

I set the TRIM.FaTE scenario property exportBiotaIntake to false, but I still have the option to select animal receptors for oral intake analyses in the TRIM.Risk$_{Eco}$ GUI on Tab 3 or 4.

If animals were included in the TRIM.FaTE run, but the TRIM.FaTE scenario property exportBiotaIntake was set to false, the user will still be able to view the animals through the Receptor Assistant, but will ***not*** be able to select toxicity values to calculate HQs on Tab 8. Do not select any oral intake receptors on Tabs 3 or 4, and proceed through the GUI.

I want to calculate a HQ for a biotic receptor in a specific TRIM.FaTE volume element. How do I do that?

In the current version of the TRIM.Risk$_{Eco}$ GUI, if a biotic receptor is selected in Tab 3 or 4, the HQ results will be calculated for the receptor in ***every*** volume element that it was modeled in TRIM.FaTE. The user can only select specific volume elements for abiotic receptors selected via Tab 6.

The HQs calculated based on oral intake exposure measures from my steady-state TRIM.FaTE simulation are all zero in my TRIM.Risk results.

Oral intake rates cannot be calculated for a steady-state TRIM.FaTE simulation because they are mass flux values. Since a steady-state run does not have a temporal aspect to it, mass flux outputs (in mg/kg-day) are set to zero (see note in Section 4.1.1). Thus, ecological HQs cannot be calculated based on oral intake rates from a steady-state simulation. Instead, use media concentration exposure estimates (selected on Tab 5) to calculate ecological risks for a steady-state run.

It takes a long time to make selections in the TRIM.Risk$_{Eco}$ GUI, and it is taking a long time for it to run.

Depending on the size of the input exposure database (from TRIM.FaTE), it may take awhile to make selections in, and to run the GUI. (For example, using a TRIM.FaTE database generated from a one year simulation of three chemicals with moles, mass, and concentration results exported every two hours, it took 20 minutes for TRIM.Risk$_{Eco}$ to run when all exact receptor and media receptor matches were selected and HQ values were calculated for one instantaneous time step on a one gigaHertz Pentium 3 computer.) To be sure that TRIM.Risk$_{Eco}$ is actually running, go to Task Manager (opened by holding down Ctrl+Alt+Delete keys and clicking the "Task Manager" button), and check on the Processes tab to see what program, if any, is utilizing the computer's resources. If System Idle or any program other than a JAVA or MySQL executable is utilizing the majority of the computer's resources, TRIM.Risk$_{Eco}$ is not running properly and should be manually terminated. Highlight the TRIM.Risk$_{Eco}$ module instance and select the "stop" button ( ) at the top of the scenario window. If TRIM.Risk$_{Eco}$ was not running correctly, check TRIM.Risk logs and input files to identify the problem.

D.3 DAVE Troubleshooting

My database does not show up in the DAVE Database Selector window list.

The Database Type at the top of the window must be set to "Ecological Risk" to view the results from TRIM.Risk$_{Eco}$. If the database still is not in the list, be sure that TRIM.Risk$_{Eco}$ executed successfully and double-check the property "Eco Risk DB Name" on the "Input Panels" tab.

[This page intentionally left blank.]

Appendix E
TRIM.FaTE Library Settings Required for TRIM.Risk$_{Eco}$

TRIM.FaTE is designed to provide inputs directly to TRIM.Risk$_{Eco}$. To create outputs in the right format and units for TRIM.Risk$_{Eco}$, certain library and scenario properties must be set appropriately in TRIM.FaTE. The Public Reference Library available with TRIM.FaTE has the correct library settings for use with TRIM.Risk$_{Eco}$. However, if the Public Reference Library is modified for an application, or a different library is used, this appendix should be referenced to check that the TRIM.FaTE library settings are set correctly for TRIM.Risk$_{Eco}$. The TRIM.FaTE scenario properties that are needed for creation of TRIM.Risk$_{Eco}$ inputs are described in Section 4.1.1 of this volume of the User's Guide.

Configure the TRIM.FaTE library prior to creating the TRIM.FaTE scenario,[1] as described in the steps below:

1. Specify the "CAS" property for each modeled chemical. The value for this property must exactly match the CAS number provided (including any formatting) for the chemicals in the ecological toxicity MySQL database. This matching allows for the appropriate chemical benchmarks to be used from the ecological toxicity database.

2. Set the "category" properties for the following abiotic compartments such that they start with the text strings in the table below. These settings allow for concentrations in abiotic compartments to be matched to benchmarks based on abiotic concentrations.

Compartment	Category
Soil - Surface	Abiotic | Soil | Surface Soil
Soil - Root Zone	Abiotic | Soil | Root Zone
Surface Water	Abiotic | Surface Water
Sediment	Abiotic | Sediment
Air	Abiotic | Air

It is important to note that the category value in the library can have additional category information after the category provided in the table above. For example, it is acceptable for the Sediment compartment to have a category of "Abiotic | Sediment | Sediment - default."

[1] Note that users can alternatively perform this series of steps directly on a TRIM.FaTE scenario by completing the steps for each chemical and relevant compartment in the scenario (e.g., for Step 2, the user would set the "category" property for all abiotic compartments in the scenario). This process is more time-consuming because it requires making the same changes to multiple compartments of the same type, but may be preferable when configuring an existing scenario to create TRIM.Risk$_{Eco}$ inputs.

3. Set the values for the "concentrationOutputUnits" and "concentrationOutputFactor" properties for the following abiotic compartments to the values in the table below. The "concentrationOutputFactor" is multiplied by the result from TRIM.FaTE to produce results in the correct "concentrationOutputUnits" to match with the units for benchmarks in the ecological toxicity database. For instructions on setting output concentration units, refer to the TRIM.FaTE User's Guide.

Compartment Type	concentrationOutputUnits	concentrationOutputFactor
Air	ug/m^3	1E06
Soil - Surface	ug/g dry weight (or mg/kg dry weight)	1000 / (Compartment.rho * Compartment.VolumeFraction_Solid)
Soil - Root Zone	ug/g dry weight (or mg/kg dry weight)	1000 / (compartment.rho * compartment.VolumeFraction_Solid)
Surface Water	mg/L	1000
Sediment	ug/g dry weight (or mg/kg dry weight)	1000/(compartment.VolumeFraction_Solid * compartment.rho)

4. For each of the biotic compartments for which biota intake and body burden data will be output to the database, set the value for the "concentrationOutputUnits" property to "mg/kg wet weight" (or "ug/g wet weight") and the value for the "concentrationOutputFactor" property to "1000." These settings are designed to match the units of benchmarks in the ecological toxicity database. For instructions on setting output concentration units, refer to the TRIM.FaTE User's Guide.

5. Set the property "wetConcOutputFactor" in the sediment compartment to the following value. This factor is used to produce estimates of sediment compartment concentrations in units of wet weight from the concentration outputs for sediment (specified above) in dry weight. If the property does not exist in the sediment compartment, add it to the compartment.

   (compartment.VolumeFraction_Solid * compartment.rho) /
   ((compartment.VolumeFraction_Solid * compartment.rho) +
   (compartment.VolumeFraction_Liquid * Constants.kg_per_m3_Water))

6. Set the property "wetConcOutputUnits" in the sediment compartment to "ug/g wet weight" (or "mg/kg wet weight"). This property indicates the units of the sediment compartment concentrations in units of wet weight, as calculated from the dry weight units and the "wetConcOutputFactor" property specified above.

Appendix F
Documentation of Ecological Risk Assessment Component Executables

The "Ecological risk assessment with TRIM.FaTE" project has three component executables: TRIM.FaTE, TRIM.Risk$_{Eco}$, and DAVE. This appendix provides documentation to, or references for, these executables.

F.1 TRIM.FaTE

TRIM.FaTE produces ecological exposure estimates for TRIM.Risk$_{Eco}$. Extensive documentation of TRIM.FaTE is available on EPA's TRIM.FaTE website (http://www.epa.gov/ttn/fera/trim_fate.html). The documents which will provide the most assistance in understanding the FaTE executable are listed in the table below.

TRIM.FaTE Documentation

Document	Description	Reference
TRIM.FaTE User's Guide	User's guide for the TRIM.FaTE module; includes instructions for creating the inputs required by the TRIM.Risk ecological risk assessment application.	U.S. Environmental Protection Agency (EPA). 2003. TRIM.FaTE User's Guide. Office of Air Quality Planning and Standards. March.
TRIM.FaTE Technical Support Document. Volume 1: Description of Module	Provides a description of terminology, model framework, and functionality of TRIM.FaTE.	EPA. 2002. Total Risk Integrated Methodology: TRIM.FaTE Technical Support Document. Volume I: Description of Module. EPA-453/R-02-011a. Office of Air Quality Planning and Standards. September.
TRIM.FaTE Technical Support Document. Volume 2: Description of Chemical Transport and Transformation Algorithms	Presents a detailed description of the algorithms used in TRIM.FaTE.	EPA. 2002. Total Risk Integrated Methodology: TRIM.FaTE Technical Support Document. Volume II: Description of Chemical Transport and Transformation Algorithms. EPA-453/R-02-011b. Office of Air Quality Planning and Standards. September.

F.2 TRIM.Risk$_{Eco}$

1. Introduction

The TRIM risk characterization module for ecological risk (TRIM.Risk$_{Eco}$) is a component of the TRIM multimedia risk analysis system and is used to calculate hazard quotients (HQs) for ecological receptors exposed to chemicals in the environment. TRIM.Risk$_{Eco}$ includes a Graphical User Interface (GUI) which is designed to provide an environment for the TRIM user to interact with the TRIM.Risk$_{Eco}$ module. The GUI for TRIM.Risk$_{Eco}$ stores certain data relating to the user's selections in an ASCII file called the "control" file, and the remaining data in the TRIM.Risk$_{Eco}$ output database (Eco Risk DB, see Appendix C for the database schema). TRIM.Risk$_{Eco}$ uses the data in the control file and the Eco Risk DB to determine the combinations of receptors, chemicals, benchmarks, and timesteps for which calculations are needed, and then calculates HQs and writes the results (along with information about the receptors, chemicals, benchmarks, and timesteps) into the Eco Risk DB. The results table in the Eco Risk DB can be queried directly for results or can be used as the input into a data visualization tool.

This appendix provides a description of TRIM.Risk$_{Eco}$. Section 2 provides details on the capabilities of the model; Section 3 lists the processes that are modeled (or will be modeled in future versions) with TRIM.Risk$_{Eco}$; Section 4 describes how the main program was implemented; Section 5 describes the files and tables that contain the inputs to and outputs from TRIM.Risk$_{Eco}$ and provides a brief description of the data; and Section 6 provides a description of the classes used in TRIM.Risk$_{Eco}$. Section 6 is intended to provide information that may be used by programmers to understand the structure of the TRIM.Risk$_{Eco}$ model code and to modify that code in the future.

2. TRIM.Risk$_{Eco}$ Capabilities

TRIM.Risk$_{Eco}$ is designed to calculate hazard quotient (HQ) values using benchmarks selected by the user for ecological receptors selected by the user. The hazard quotient calculation is simply the ratio of the chemical concentration in the media of interest (surface water–total or dissolved, soil, sediment, air) or the chemical exposure (dose, body burden) to the selected benchmark. There are seven possible types of benchmarks, and each must be paired with the appropriate data to generate a meaningful HQ result. The possible benchmark types are biota oral intake, body burden, soil, sediment, air, total surface water, and dissolved surface water. Total surface water benchmarks are compared to the total concentration of chemical in the water (including chemical adsorbed to suspended solids), while dissolved surface water benchmarks are compared to the concentration of chemical dissolved in the water. Benchmarks may be developed from a variety of information and may relate to specific end points and effects of interest (e.g. a lowest observable adverse effects level [LOAEL] for reproductive success).

Prior to the execution of the TRIM.Risk$_{Eco}$ model, the user will run TRIM.FaTE to simulate contaminant fate and transport in the environment. The physical system is considered to be composed of volume elements. A volume elements is essentially a discrete volume section of the system being modeled. Associated with each volume element are compartments, which may be "abiotic" such as environmental media (air, surface water, soil, sediment) or "biotic"

such as specific receptors (mouse, duck, weasel, etc.). Each volume element has one associated "primary" abiotic compartment.

While it is convenient to think of biotic compartments in TRIM.FaTE like receptors, the biotic compartment definition and label may not be identical with those used in the ecological toxicity database. It is up to the user to identify which benchmarks are appropriate to use for HQ calculations for the given biotic compartment. So, for example, it is possible that the user has specified a "duck" as the TRIM.FaTE compartment name and bird as the taxon, whereas from the ecological toxicity database the user selected benchmarks for both a wood duck and a mallard duck (which are both in the "bird" receptor group) for the HQ calculations.

In addition to the spatial component in TRIM.FaTE, there is also a temporal component. Output from a given TRIM.FaTE steady-state simulation includes steady-state estimates, and from a dynamic simulation includes instantaneous results at user-specified frequency. The dynamic simulation output may additionally include a set of average values which are the averages of the instantaneous values for a user-specified time duration (e.g., annual, monthly, etc.). TRIM.FaTE does not have the capability currently to calculate multiple averaging times within the same run, but that feature may be implemented in the future.

In executing the TRIM.Risk$_{Eco}$ model, the user will specify the type of temporal outputs from TRIM.FaTE (e.g., instantaneous or averaged) that they wish to use. This could be for all time points in the TRIM.FaTE output database, a selected portion, or for a single time point. If selecting a portion of the data, the user selects the beginning and ending time. For a single time point, the user will select either the last time point, the time point that produces the maximum HQ value (i.e. the one with the highest concentration, wildlife oral intake, or body burden), or a specific time point. Different time points can be specified for each type of exposure estimates.

The user will use the TRIM.Risk$_{Eco}$ GUI to specify the exposure data locations (e.g., TRIM.FaTE volume elements and compartments) and the toxicity values against which to compare those outputs from the ecological toxicity database. The user will also specify the time points for which calculations will be conducted. Based on this the GUI will generate a set of data in the TRIM.Risk$_{Eco}$ output database.

3. TRIM.Risk$_{Eco}$ Processes

TRIM.FaTE models a site as a set of volume elements, each of which has certain properties relating to the media associated with it, the biotic compartments it contains, the chemicals for which data are available, and the concentrations of the chemical in the various media. It is expected that the user may wish to run TRIM.Risk$_{Eco}$ for everything available in TRIM.FaTE (i.e. all volume elements, all biotic compartments associated with those volume elements, and all time points), for selected compartments and volume elements, for particular benchmarks and benchmark values of interest, and/or for selected time points. The TRIM.Risk$_{Eco}$ GUI provides the mechanism for the user to make these choices. The TRIM.Risk$_{Eco}$ model implements the calculations for the selected cases designated by the user through the TRIM.Risk$_{Eco}$ GUI.

The following list is a summary of the processes that TRIM.Risk$_{Eco}$ performs.

- Read user input data generated through the user's session with the TRIM.Risk$_{Eco}$ GUI. These data are in two places: the Eco Risk control file, and data tables constructed by the TRIM.Risk$_{Eco}$ GUI based on the user's selections and placed in the Eco Risk DB.
- Read the name of the control file from the command line (first argument) or assume by default the name of the control file is "control.txt" and that the control file is located in the same directory as the executing program.
- Send messages from the program to the console. In general, only error messages are sent. More detailed messages useful for program debugging are sent to the messages file by running the program in "verbose" mode. Verbose mode is activated from the command line through the use of a "-v" switch.
- The control file is a plain text file. All data in the control file are provided in the control file in the format "Data Element Name=Value". Lines beginning with a hash mark (#) are considered comments and are ignored.
- Create a table in the Eco Risk DB called "times" and enter into it the time at the beginning of program execution and the time at the end of program execution.
- Load the control file data elements and their values into a table in the Eco Risk DB called "control."
- Put results from running TRIM.Risk$_{Eco}$ into the Eco Risk DB in a table called "results." Select input data regarding the receptors, volume elements, chemicals, benchmarks, effects, and end points to use in the analysis from the Eco Risk DB tables generated by TRIM.Risk$_{Eco}$ GUI.
- Based on the user's selection, run the model in one of three modes (only one mode, the screening mode, is active in the initial 2005 release (i.e., Version 1.0)):
    - Default - the default mode is a deterministic analysis in which a pre-determined set of benchmarks and time steps are used. The user may still choose the volume elements and receptors of interest. Within the TRIM.Risk$_{Eco}$ model, operation in default mode is the same as for screening mode.
    - Screening - the screening mode is a deterministic analysis in which the user has selected the volume elements, compartments, benchmarks, and time steps for the analysis.
    - Probabilistic - to be determined.
- Conduct deterministic analyses using time series concentration/dose/body burden data from TRIM.FaTE. Based on user selections, time series data can be used in one of five ways. In the "selected" mode, the user may define a beginning and an ending time step, or may use the entire time series.:
    - All - all time series data is used (this is the default).
    - Max - the time step with the highest value for concentration/oral intake/body burden is used.
    - Last - the last time step in the data series is used.
    - Selected - time series data are limited to a user-defined beginning and ending timestep.
    - Single - the user has selected a single time point to use.
- The user can vary time series selections by benchmark type.
- Calculate HQ values for all volume elements, compartments, chemicals, benchmarks, effects, end points, and time steps designated by the user.
- Output the results to the Eco Risk DB into a table called "results" that includes the calculated HQs and volume element, compartment, chemical, time, and benchmark

information.

- Create an index on the results table to speed querying.
- Do reasonable amounts of error trapping to ensure that the program exits "gracefully" and provides a message when error conditions are encountered.

4. Description of the Main Program

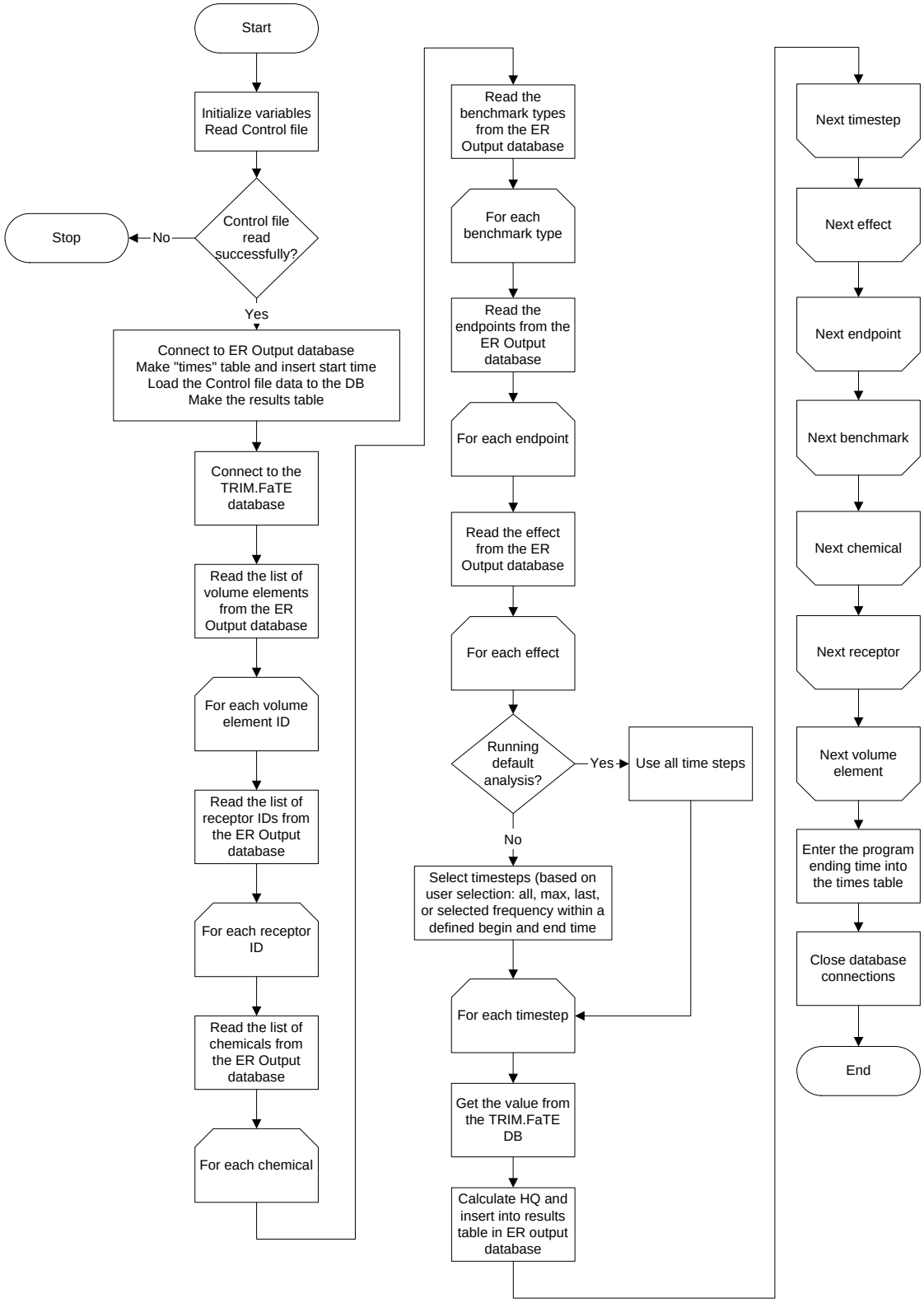
The TRIM.Risk$_{Eco}$ model is designed to calculate HQ values for the compartments and chemicals designated by the user. HQ values are calculated as the ratio of a media concentration, body burden, or wildlife oral intake to an ecological toxicity value (i.e., benchmark). The following factors determine the data used in the calculation:

- *Volume element.* The TRIM.FaTE volume element is for a particular type of media (e.g., air, water, soil).
- *Compartment.* The user specifies which compartments (biotic and abiotic) will be used for each volume element as a basis for HQ calculations.
- *Chemical*. The combination of receptor and chemical determines the health benchmark data that are available.
- *Benchmark type*. The benchmark type determines which TRIM.FaTE output variable (e.g., type of exposure measure) is to be used in the HQ calculation. Choices are dose (wildlife oral intake), body burden, surface water (total or dissolved), air, soil, or sediment.
- *Effect*. The effect describes the health outcome that the benchmark relates to, such as morbidity or reproductive success.
- *Endpoint*. The endpoint (termed toxicity metric in GUI screens) describes the nature of the benchmark, such as a No Observable Adverse Effects Level (NOAEL).
- *Timestep*. Output from TRIM.FaTE may be steady-state or instantaneous estimates through time. Additionally, it may also include averaged estimate through time. TRIM.Risk$_{Eco}$ can use different time points for the different types of benchmarks (and exposure estimates). For example, body burden values may be analyzed using the highest annual average output, while surface water concentrations may be analyzed using the a range of instantaneous results.

The combination of receptor, chemical, benchmark type, effect, and endpoint defines the specific value of the benchmark that is used in the denominator of the HQ calculation. The combination of the volume element, compartment, chemical, benchmark type, and timestep identifies the exposure estimate used in the numerator of the HQ calculation.

The user expects the program to generate HQ values for the volume elements, receptors, chemicals, benchmarks, and timesteps they have selected through their interactive session with the TRIM.Risk$_{Eco}$ GUI. To accomplish this, the TRIM.Risk$_{Eco}$ model operates in default or screening mode according to the general pseudocode provided below. A detailed flow chart of the operation of the main program is depicted in Exhibit F-1.

Exhibit F-1. TRIM.Risk$_{Eco}$ Executable Flowchart[a]

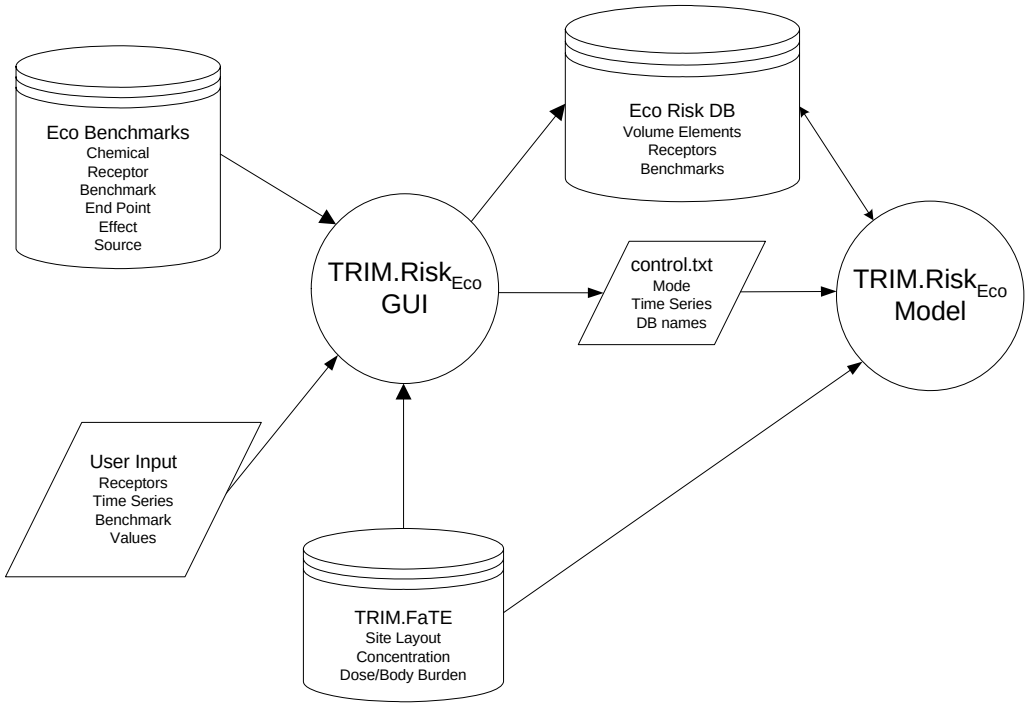


[a] This flowchart may not reflect final edits to the initial release executable with regard to capabilities for steady-state values.

5. Program Inputs and Outputs

TRIM.Risk$_{Eco}$ inputs and outputs are depicted in Exhibit F-2. TRIM.Risk$_{Eco}$ receives data from several sources. These include the Eco Risk control file, selected tables from the Eco Risk DB, and selected tables from the TRIM.FaTE database. The TRIM.Risk$_{Eco}$ model writes results to the Eco Risk DB into selected tables. The details of the input and output files and databases are described in the following sections.

Exhibit F-2. TRIM.Risk$_{Eco}$ Inputs and Outputs



Program Inputs

The TRIM.Risk$_{Eco}$ model receives inputs from the Eco Risk control file created by the TRIM.Risk$_{Eco}$ GUI, and from the following database tables:

Eco Risk DB:

- tblER_VE - this table contains information about user-selected volume elements and receptors.
- tblER_Comp - this table contains information about the combinations of compartments, benchmarks, and receptors that the user has selected.
- tblER_Benchmarks - this table contains the benchmarks data the user has elected to use for each combination of receptor and chemical.
- tblER_Timesteps - this table contains the user's time point selections for each benchmark type.

TRIM.FaTE database:

- Variables - this table contains information about the variables available in TRIM.FaTE. The table is used to identify the variable ID numbers of interest for getting the needed TRIM.FaTE results.
- Results - this table contains the time series of results for each TRIM.FaTE variable. The data are accessed via the variable ID from the variables table.
- Chemicals - this table contains the list of chemicals for which data are available in TRIM.FaTE. The table is used to map the chemical ID number in the variables table to a CAS number.
- Objects - this table contains the list of object identification numbers and the associated object names.

A complete description of the fields in each of these database tables is available in Appendix C with the database schema. In addition, the inputs in the Eco Risk control file are listed here:

Eco Risk Control File, created by the TRIM.Risk$_{Eco}$ GUI:

Control File Data Element Name	**Description**
Ecological Risk Output Database	The name of the database that will be used to store the results from the TRIM.Risk$_{Eco}$ runs.
FATE Database	The name of the TRIM.FaTE database that will be used in the analysis.
Ecological Effects Database	The name of the database containing ecological effects (i.e., ecological toxicity) data.
Default Analysis	Logical flag indicating whether the analysis to be conducted is the "default" analysis (possible values are "True" and "False"). If this element is "True", the "Screening Analysis" and "Probabilistic Analysis" elements must be "False." The default analysis is a deterministic analysis.
Screening Analysis	Logical flag indicating whether the analysis to be conducted is the "screening" analysis (possible values are "True" and "False"). If this element is "True", the "Default Analysis" and "Probabilistic Analysis" elements must be "False." A screening analysis is a deterministic analysis.
Probabilistic Analysis	Logical flag indicating whether the analysis to be conducted is the "probabilistic" analysis (possible values are "True" and "False"). If this element is "True", the "Default Analysis" and "Screening Analysis" elements must be "False."
Number of Realizations	An integer defining the number of Monte Carlo iterations to run when the "Probabilistic Analysis" is true (future version).
Randomization Seed	An integer defining the starting "seed" that will be used in the randomization function in the program. This defines the sequence of numbers that will be used in selecting the Monte Carlo iterations for the analysis (future version).

Program Outputs

TRIM.Risk$_{Eco}$ outputs information to the Eco Risk DB relating to results and the time it took to run the program. It also outputs to the Eco Risk DB the control file data from the ASCII file created by TRIM.Risk$_{Eco}$ GUI. The results output to the Eco Risk DB can be accessed using either the Data Analysis and Visualization Engine (DAVE), MySQL queries, or the MySQL Query Browser.

6. Description of Classes in TRIM.Risk$_{Eco}$ Module

This section, which is primarily intended for programmers, describes the classes used in the TRIM.Risk$_{Eco}$ module, including their purpose, properties, and functions.

Class CBenchmarks

Files Containing Definition:
Benchmarks.cpp, Benchmarks.h

Purpose:
This class is designed to get, hold, and return the various parameter values associated with the current benchmark.

Enumerated constants:

NUMPARAMS: the maximum number of parameters used to define the benchmark distribution

STRLEN: the maximum length of a string (used to create character arrays for containing strings)

Enumerated types:[2]

type: the type of benchmark. Possible values are: none, dose, bodyburden, airconc, sedconc, soilconc, swconcdiss, swconctot

Public Functions:
CBenchmarks() - Class constructor
~CBenchmarks() - Class destructor
double CalcValue() - calculates benchmark value based on a random number, the distribution type, and the distribution parameters. Used for probabilistic calculations only. Included in Version 1.1.
type GetType() - returns the value for the type of benchmark
char * GetAbbrev() - returns the abbreviation for the benchmark
void PrintVals() - prints the values for the distribution parameters to cerr.
bool GetUnits(char * , char * , char *, char *) - puts into the first char * parameter the portion of the query string relating to the benchmark type and units
bool GetBenchmarkVal(char *, DBWriteObject *) - queries tblER_Benchmarks to get the data

[2] Dose here refers to wildlife oral intake values.

on the current benchmark and put the values into the appropriate CBenchmarks class variables

void InitVars() - initializes the variables used in the CBenchmarks class.

Public Variables:
double Value - value of the benchmark
double DistParams[NUMPARAMS+1] - array containing the distribution parameters read in from tblER_Benchmarks.
int ID - the benchmark ID
char Name[STRLEN] - the benchmark name
char Units[STRLEN] - the benchmark units
char Abbrev[6] - the benchmark abbreviation
char Lookup[STRLEN] - Lookupname for the associated compartment
bool IsUserDefined - boolean indicating whether the benchmark is a user-defined value (true if the value is user-defined)
int recid - receptor ID for the receptor associated with the benchmark
char cas[STRLEN] - CAS number for the chemical associated with the benchmark
type Type - the benchmark type

Class CCtrlFile

Files Containing Definition:
CtrlFile.cpp, CtrlFile.h

Purpose:
This class gets and handles data associated with the Control File generated by the TRIM.Risk$_{Eco}$ GUI.

Enumerated Constants:
MAXPATH - maximum length for the file path string
MAXSTRLEN - maximum length for strings
MAXDBNAMELEN - maximum length for the database name

Public Functions:
CCtrlFile() - class constructor
~CctrlFile() - class destructor
bool ReadControlFileData(char *) - reads data from the designated control file and puts values in the class variables based on the data read from the control file.
char * GetERDB_Name( ) - returns the name stored in the Eco Risk DB variable.
char * GetTRIMDB_Name( ) - returns the name stored in the TRIM database variable.
char * GetCtrlFileName( ) - returns the name stored in the control file name variable.
int GetNumRealizations() - returns the value in the variable that stores the number of realizations.
bool GetDefault() - returns value for variable used to store whether model is being run in default mode.
bool GetScreening() - returns value for variable used to store whether model is being run in screening mode.

bool GetProb() - returns value for variable used to store whether model is being run in probabilistic mode.
int GetSeed() - returns value for random number seed.

Public Variables:
static bool verbose - verbose mode flag

Private Variables:
char ERDB_Name[MAXDBNAMELEN] - Eco Risk DB name
char TRIMDB_Name[MAXDBNAMELEN] - TRIM database name
char CtrlFileName[MAXSTRLEN] -control file name
char SessionName[MAXSTRLEN] - session name
bool probabilistic - true if running in probabilistic mode
bool screening - true if running in screening mode
bool default1 - true if running in default mode
int seed - random number generator seed
int NumRealizations - number of realizations (when running in probabilistic mode - Version 1.1)

Class CTimestep

Files Containing Definition:
Timestep.cpp, Timestep.h

Purpose:
This class handles actions associated with getting and implementing the user-designated time steps used in hazard quotient calculations for each benchmark.

Enumerated Constants:
MAXSTRLEN - maximum string length.

Enumerated Types:
timesteps - user timestep selection (possible values are all, last, max, and selected).

Public Functions:
Ctimestep() - Class constructor
~Ctimestep() - Class desctructor
timesteps GetTimestep() - returns the timestep variable value
bool ReadTimestep(DBWriteObject * , char *) - reads the timestep info from the database.
char * GetBeginstep() - returns the value of the beginning timestep value
char * GetEndstep() - returns the value of the last timestep value
int GetFrequency() - returns the value of the timestep sampling frequency
bool CreateQueryString(DBObject * , DBWriteObject * , CCtrlFile * , char * , char * , char * , char *) - create the query strin used to pull the appropriate timesteps from the TRIM database, based on the user's selections

Public Variables:
timesteps timestep - timestep value

char beginstep[50] - beginning timestep
char endstep[50] - ending timestep
int frequency - sampling frequency
int duration - duration of the averaging time
char avg_or_inst[5] - text indicating whether the TRIM time value is an average or instantaneous one

Class DBObject

Files Containing Definition:
DBObject.cpp, DBObject.h

Purpose:
Class for handling connections and querying of mysql databases to get data. DBObject only reads from a database; it will not write to it.

Enumerated constants:
MAXPATH - maximum length for the path name variable.
MAXSTRLEN - maximum string length
MAXDBNAMELEN - maximum database name length

Public Functions:
DBObject() - Class constructor
~DBObject() - Class constructor
int Connect(char *) - connects to a database with the name passed in the char * parameter.
int ConnectExists(char *) - if database exists, it is dropped prior to connection and a connection to a new database is established.
bool GetData(char *) - gets data using a query based on the content of the char * parameter.
bool Get2Values(char *, int &, const char *, int &, const char *) - queries the database based on the char * parameter to get a value from the 2 fields specified in the const char * parameters and puts them into the int & parameters.
bool DoQuery(char *) - runs a query based on the char * parameter on the database.
bool GetCharValue(char * , char * , const char *) - gets a string value from a table using the query defined in the first character string and located in the field named in the third character string and puts it into the variable in the second character string.
bool GetValues(char * , Row &) - gets a row of values based on the query in the character string
template <class T> bool GetValue(char * , T & , const char *) - gets a value from a table using the query defined in the first character string and located in the field named in the second character string and puts it into a variable of the appropriate class T.

Public Variables:
Result qry_res - Result object for query
char dbName[MAXSTRLEN] - name of the database associated with the DBObject object.

Protected Variables:
Connection con1 - database connection for connecting to mysql database

Query query1 - query variable for executing a query on the mysql database

Class DBWriteObject: public DBObject

Files Containing Definition:
DBWriteObject.cpp, DBWriteObject.h

Purpose:
Object for working with mysql databases that must also be written to as well as read from. This class is derived from the DBObject class.

Public Functions:
DBWriteObject() - Class constructor
DBWriteObject(bool) - Class constructor - if boolean is true, this is for the Eco Risk DB. This is used as a check to make sure the control file is not loaded into a database other than the Eco Risk DB.
~DBWriteObject() - Class destructor.
int MakeTable(char *, char *) - Makes a table with a name in the first char* parameter and the table field definitions in the second char* parameter.
int InsertRow(char *, char *) - Inserts a new record into a table with a name in the first char * parameter and a query string portion in the second char * parameter that defines the data to be inserted.
bool InsertUniqueRow(const char * , char *) - inserts a row into a table only if the row is not already in the table
bool LoadCtrltoDb(char *) - Loads the data from the control file (control.txt or other name specified in the char * parameter) into the database.

Private Variables:
bool EcoOutDBFlag - Boolean, set to “true” if the specific object created is for the Eco Risk DB.

Class FieldInfo

Files containing definition: Datatables.h

Purpose: The FieldInfo class contains information relating to a single field in a table, including the FieldName, DataType, and Value. A collection of fields makes up a table.

Enumerated Constants
MAXSTRLEN - maximum string length

Public functions:
FieldInfo() - Class constructor.
FieldInfo(const char * , const char * , const char *) - Class constructor that sets the fieldname, data type, and value variables.
FieldInfo(const char * fn, const char * dt) - Class constructor that sets the fieldname and datatype variables.
void Add2String(char *) - Creates portion of query string for creating field with the given field

name and data type and puts it in the character string
bool Add2String2(char * strng) - Creates portion of query string for setting value of field with the given field name and value and puts it in the character string

Public Variables:
char FieldName[MAXSTRLEN] - The name of the field
char DataType[MAXSTRLEN] - The data type for each field
char Value[MAXSTRLEN] - The value for the field

Class RECORD

Files containing definition: Datatables.h

Purpose: Type definition for a collection of FieldInfo objects that would make up a record to be entered into a database table.

typedef vector< FieldInfo > RECORD

Class TABLE: public RECORD

Files containing definition: Datatables.h

Purpose: To help manage tables and data written to the Eco Risk DB.

Enumerated constants:
MAXSTRLEN - maximum string length

Public Functions:
TABLE() - Class constructor
~TABLE() - Class destructor
void Copy(const char * , const char *) - Copies a string data value (first character string) into the "Value" variable associated with the field (second character string)
void Copy(const char * , const int) - Copies an integer data value (first character string) into the "Value" variable associated with the field (second character string)
void Copy(const char * , const double) - Copies a double data value (first character string) into the "Value" variable associated with the field (second character string)
void Copy2Field(const char * , const char *) - Looks up the Field (first character string) in the FieldInfo list and assigns the value (second character string) to the associated Value variable
char * LookupString(char * , char *) - Creates the query string (return value) for looking up data associated with the primary key value (second character string) from the table in the database (first character string) containing the source data
char * CreateTableString() - Function returning a string containing the query to create the table associated with the TABLE object
char * AddRecordString() - Function returning a string with the query to add a single record to the table associated with the TABLE object
char * AddRecordsString() - Function returning a string containing the query to populate the

table associated with the TABLE object

char * AddRecordsString(char *) - Function returning a string containing the query to populate the table associated with the TABLE object using data from a table located in the database identified by the character string

void CopyAll( ) - Copies all the data values from a row object to the associated Value variable for that field in the TABLE object

Public Variables:

char TableName[MAXSTRLEN] - The name of the table in the Eco Risk DB that the TABLE object is associated with

char PrimaryKey[MAXSTRLEN] - Primary key field(s) for the associated table

char LookUpFields[MAXSTRLEN] - String containing the names of fields in the table used to populate the lookup table when the names of the fields in the lookup table differ from those in the populating table. E.g. the compartments table has a CompID and CompName fields which are populated from the objID and objname fields in the objects table of TRIM

char LookUpCondition[MAXSTRLEN] - String containing the "where" condition for populating the lookup table

char LookUpTableName[MAXSTRLEN] - Name of the table used to populate the lookup table in the Eco Risk DB associated with the TABLE object

char lookup_querystring[MAXSTRLEN] - Query string used to look up the data associated with a particular primary key value

char querystring[MAXSTRLEN] - Query string used for creating the table associated with the TABLE object

Row qry_row - record to go into the table

Global Functions

bool InitVars() - initializes program variables and TABLE objects.

bool ReadControlFile(CCtrlFile &, int, char **, int) - reads the control file based on either the standard control file name or a file name designated in the command line string.

GetList(char * querystring, DBWriteObject *, char (* &)[MAXSTRLEN], const char *) - Gets a list of items from a table based on a query string piece in the first char* parameter, using the database object defined in the parameter list and putting the values into the char (*&) array based on the field name in the const char*.

void MakeResultsString(Row &, int, char *, int, CBenchmarks *, double) - constructs a query string for inserting a row into the results table.

int main(int, char*) - main program.

F.3 DAVE

The Data Analysis and Visualization Engine (DAVE) is a tool for analyzing the TRIM.Risk results that is included with the “Ecological risk assessment with TRIM.FaTE” project. Documentation of DAVE is included with the TRIM.Risk Installation in the “docs” folder. This documentation should be referenced for complete information on the executable.