OECD Aquatic Risk Indicators Computer System: User Guide

by Volkmar Gutsche (German BBA) and John Carley (United States EPA)

Table of Contents

1. Computer program structure

The program for the OECD Aquatic Risk Indicators REXTOX, ADSCOR, and SYSCOR uses an MS-Access database. The basic version of the indicator computer program consists of three main parts:

- model kernel
- results aggregation tool
- graphic presentation tool

Using the OECD indicator programs requires basic familiarity and facility with Microsoft-tools ACCESS, for the model kernel and results aggregation tool, and EXCEL for graphic presentation of results.

The **model kernel** serves

- to define the framework of the indicator approach, e.g.

    - What crops are considered?
    - What regions are distinguished?
    - What application methods are of interest?
    - What pesticide properties table should be linked to the program?

- to support input of the usage data

- to accept runtime parameters and calculate the indicators

- to store the basic results of indicator calculation

The kernel consists of: (see also figure 1)

- Five Access tables which together characterize each specific pesticide use comprehensively for all three indicators.

- One Access table that stores all pesticide properties relevant to the indicators.

- One Access table containing all specific pesticide usage information considered in calculating the indicators.

- Three Access macros to calculate the REXTOX, ADSCOR, and SYSCOR indicators.

- Three Access tables—one for each indicator—to store the basic results of indicator calculations as they are performed.

- One Access table to store the runtime parameters used in aggregation

- Two Access tables which store the breakpoints for the scoring models ADSCOR and SYSCOR, and one additional Access table that covers the penalties for SYSCOR.

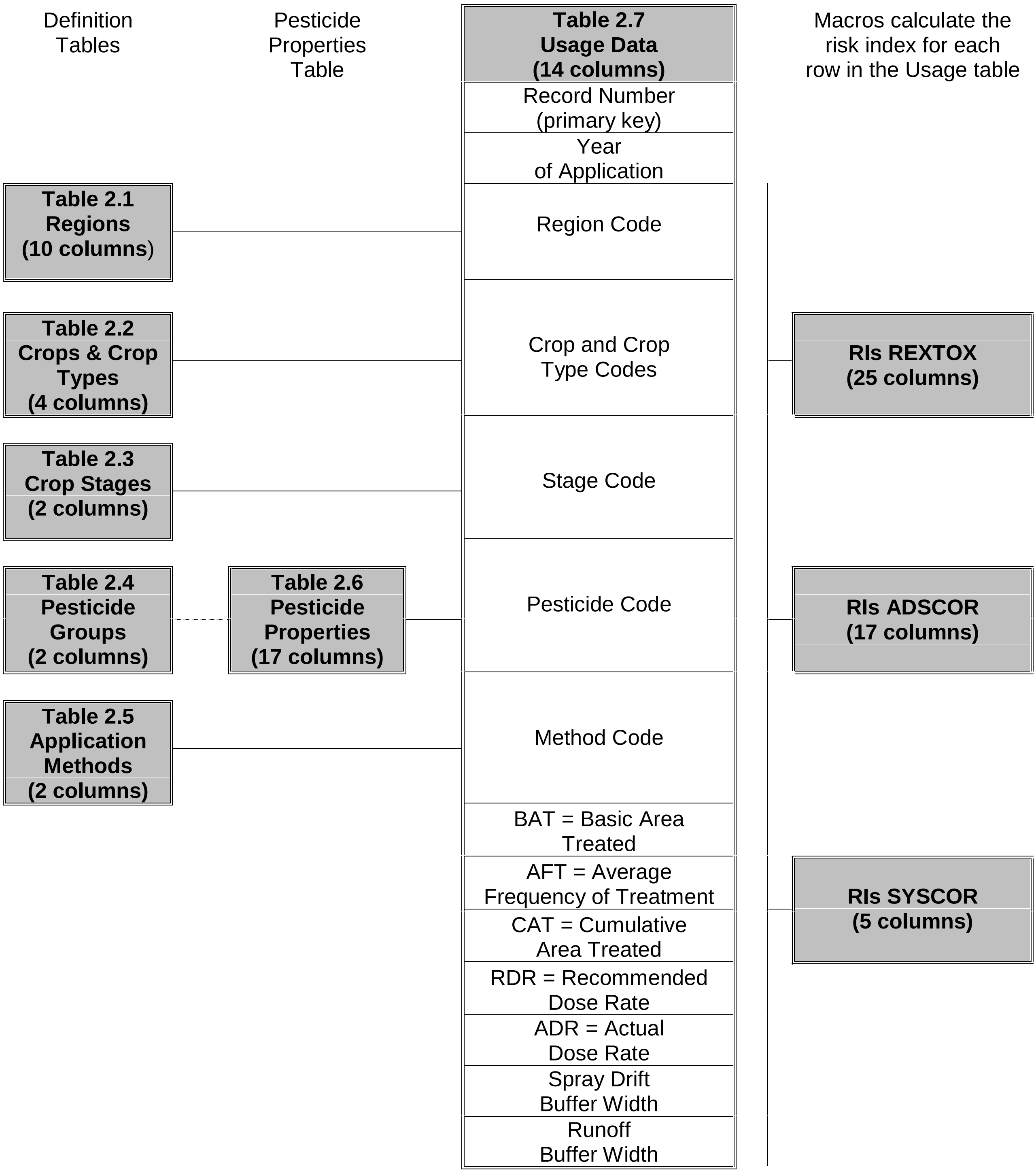
- Several Access forms to support use data input and indicator runs.

The three basic results tables are filled as calculations are performed on each new entry (row) in the usage table.

Result aggregations are performed using Access query tables. Only the most obvious possible aggregations are programmed in the initial pilot software, but additional query tables can easily be defined for other patterns of aggregation.

For **graphic presentation** of results, Excel-spreadsheets are linked with different tables of the indicator database. These spreadsheets include pre-prepared diagrams for graphic presentations. In the basic version of the computer program only a few basic diagrams are demonstrated, but they can easily be enhanced by users familiar with MS-Excel.

Figure 1: Structure of Model Kernel



2. Structure of Primary Data Tables

2.1 Regions Table

This first of the primary Access data tables includes the environmental factors that differ among various regions. If a user of the indicators does not wish to compare regions below the national level, then only one row in this table must be completed, containing estimated values for the country as a whole. If regional comparisons are desired, then each region must be described in a separate entry in this table.

Table 2.1: Regions

Region Code	Region Name	Soil Type	OC%	Water Depth	Water Index	Slope	Precipi-tation	Runoff Buffer Compliance	Spray drift Buffer Compliance
1-letter ID code	Full name of region	loamy or sandy	% organic carbon in soil	Avg depth in meters of water bodies in region	Defined in EG Report	Avg slope (in %) of fields in region	Assumed precip. in mm after each pesticide application	Assumed level of user compliance with buffer requirement	Assumed level of user compliance with buffer requirement
A	Region A	L	2.0	0.4	0.15	2	10	1	1
B	Region B	L	1.8	0.3	0.12	3	30	1	1
C	Region C	L	1.6	0.3	0.10	5	30	1	1
D	Region D	L	1.6	0.3	0.11	5	20	1	1
E	Region E	L	1.8	0.4	0.13	5	20	1	1
F	Region F	L	2.0	0.2	0.14	3	30	1	1

The last two columns of the Regions table include estimates of applicator compliance with label requirements for buffer zones to protect surface water from pesticide runoff and spray drift. A value of ‘1’ represents perfect or 100% compliance; a decimal fraction less than one can be thought of as the probability of compliance. Normally detailed compliance information is not available, so an estimate for each region must be made and included in this table. Note also that the sensitivity of each indicator to the effect of buffers can be assessed by varying these values.

2.2 Crops Table

All crops included in calculating the indicators must be declared in the second of the primary Access tables, the "Crops" table illustrated below.

Table 2.2: Crops

Crop Code	Crop Name	Crop Type Code	Crop Type
COR	Corn	CE	Cereals
OAT	Oats		
RYE	Winter Rye		
SPB	Spring barley		
TRI	Triticale		
WHE	Winter Wheat		
BAR	Winter barley		
SPB	Spring barley		
CLO	Clover	FP	Forage Plants
LUC	Lucerne		
APP	Apples	FT	Fruit and Soft Fruit
CHE	Cherries		
PLU	Plums		
PRS	Pears		
HOP	Hops	HO	Hops
OSR	Oilseed rape	OI	Oil Seeds
BEA	Beans	PB	Peas & Beans
PEA	Peas		
POT	Potatoes	RC	Root Crops
SUB	Sugar beet		
ONI	Onions	VE	Vegetables
GRA	Grapes	VI	Vines

Users must choose which crops to include in calculating the indicators, and then ensure that all included crops are in this table. The three-letter identifier for each crop is the primary key, and is used to link these values to usage records.

The "Crop Type" column permits grouping of the indicator results. Note that the **values "FT" , "HO", and "VI" in the last column are mandatory if fruits, hops, and vines are included in the crops table**. The REXTOX calculations will fail unless these crop type codes are used for fruit, hops and vines.

To illustrate its role in calculating and aggregating the indicators, Table 2.2 is shown here only as an example. The actual content of this table will depend heavily on user decisions about the scope of the pilot project.

2.3 Crop Stages Table

In this version of the software the Access table "Stages" is only a two column by two row table containing the two stage codes "1" and "2" and their explanation as "early" and "late", respectively. These stage codes cannot be changed without revising the Access module "update_new" used to calculate the indicators. Among other things, the calculation of spray drift in REXTOX depends on the crop stage code.

Table 2.3: Crop Stages

Stage Code	Crop Stage
1	Early
2	Late

2.4 Pesticide Groups Table

This table is linked via the two-letter Pesticide Group Code to the pesticide properties table, and permits selective aggregation of indicator results by the class of pesticide—such as herbicides, fungicides, or insecticides.

Table 2.4: Pesticide Groups

Pesticide Group Code	Pesticide Group Name
AC	Acaricide
DE	Dessicant
FU	Fungicide
GR	Growth regulator
HE	Herbicide
IN	Insecticide
MO	Molluscicide
NE	Nematicide
PP	Pruning paint
RE	Repellent
SS	Soil sterilant

These two-character "Pesticide Group Codes" are used by the Access macros to calculate the REXTOX indicator, and should not be changed unless corresponding changes are made in the module "update_new".

2.5 Application Methods Table

The Access table "Methods" lists all methods of pesticide application which figure in calculating the indicators:

Table 2.5: Application Method

Method Code	Method Name
AB	Air blast
AE	Aerial
GB	Granular broadcast
GI	Granular incorporated
GS	Foliar spray
PT	Pruning paint
SS	Soil sterilant
ST	Seed treatment
WA	Direct application to water

These two-character "Method codes" are used by the Access macros to calculate the indicators, and should not be changed unless corresponding changes are made in the module "update_new".

2.6 Pesticide Properties Table

Toxicological properties, environmental fate rate constants, and specific regulatory limitations of each pesticide for which usage data is reported must appear in the Pesticide Properties Table in order to calculate the risk indicators.

In the initial phase of the project, a table was compiled with the properties of 305 pesticide active ingredients used on arable and tree crops. To improve normalization of the database the last columns of the original table were modified as table 2.6 below. Note that the table on the next page represents only a single pesticide—i.e., only a single row of the actual pesticide properties table. Each row in the table corresponds to a column in the actual pesticide properties table; the presentation is rotated here in order to fit the many 'columns' in this table onto a single page.

Although the structure of this table is fixed, its content may change from one country to another for various reasons. A country might choose to apply the indicators to other crops than those selected for the initial pilot, and other pesticides might have been used on those other crops. A country might also choose to replace the values provided with preferred values from a national database of pesticide properties. **It is up to each user of the indicators to ensure that all pesticides used on the crops within the scope of the indicator are included, by adding a row to the table for each pesticide not included in the original list.**

The fourth column in this table, "Pesticide Type", is optional. It is not used to calculate the indicators, but it could be used to control aggregation of indicator results by chemical families. It could also contain a second unique pesticide identifier code that could be used to link the table entry to a country's primary pesticide information resources.

Table 2.6: Pesticide Properties

Column Heading	Column Content
Pesticide Code	Primary key, automatically assigned as the pesticide is added to table
Pesticide	(User defined) full name of pesticide
Pesticide Group Code	Link to the table “Pesticide groups“
Pesticide Type	Name of the chemical family of the pesticide, or user-defined link to alternative chemical information database
Algal Acute Tox	Acute LC50 to Algae in mg/l
Daphnia Acute Tox	Acute LC50 to Daphnia in mg/l
Fish Acute Tox	Acute LC50 to Fish in mg/l
Algal LongTerm Tox	21-day NOEC to Algae in mg/l
Daphnia Long Term Tox	21-day NOEC to Daphnia in mg/l
Fish Long Term Tox	21-day NOEC to Fish in mg/l
LogKow	Logarithm of the n-Octanol-water Partition Coefficient
Koc	Organic carbon-Water Partition Coefficient (not required)
DT50w	Half life of pesticide in water, measured in days
DT50s	Half life of pesticide in soil, measured in days
Photolysis	Photolysis of pesticide in water measured in days
Solubility	Solubility of pesticide in water measured in mg/l
Training Required	“0” if no requirement for specially trained users; “1” if a specially trained or certified user is required (used only used by SYSCOR)

Table 2.6a gives an example of the content of this important table. Here the highlighted codes in the fourth row (“Pesticide Type”) are Germany’s unique BBA-codes for active ingredients. If this element is defined as the primary key in this table, then the records in the “Usage” table can refer to that ‘pesticide type’ code instead of the ‘pesticide code’.

Table 2.6a: Examples of Pesticide Properties

Pesticide Code	310	311
Pesticide	Epoxiconazol	Atrazin
Pesticide Group Code	FU	HE
Pesticide Type	**0875**	**0006**
Algal Acute Tox	2.3	0.043
Daphnid Acute Tox	8.69	115
Fish Acute Tox	6.7	15.7
Algal LongTerm Tox	0.0018	0.004
Daphnid LongTerm Tox	500	0.12
Fish LongTerm Tox	0.05	0.06
LogKow	3.4	2.34
Koc		
DT50w	140	124
DT50s	403	100
Photolysis	150	2.6
Solubility	7.05	33
Training Required		

2.7 Usage table

If all codes are well defined the input of usage data into the ACCESS table "Usage" is straight-forward, but it requires an immense expenditure of time. The structure of this ACCESS table is provided in table 2.7 and shown graphically in Figure 1. Again the table here has been rotated, so that each row represents a column in the actual Access table.

Table 2.7 Usage Data

Columns	Desciption
Record	Record number of this particular application, automatically provided by the Access System when the usage data are entered. (Primary key)
Year	Year of application
Region Code	Region code, corresponding to the table " Regions "
Crop Code	Crop code, corresponding to the table " Crops"
Stage Code	"1" for early, "2" for late, corresponding to the table " Stages"
Pesticide Code	Pesticide code, corresponding to the table " Pesticides "
Method Code	Method code, corresponding to the table "Methods"
BAT	Basic Area Treated (defined in Indicator Report)
AFT	Average Frequency of Treatment
CAT	Cumulated area treated (defined in Indicator Report)
RDR	Recommended Dose Rate
ADR	Actual Dose Rate
Spray drift Buffer Width	Spray drift buffer width for this application method as required on the label
Runoff Buffer Width	Runoff buffer width if required on the label

Each row in the table is identified by a unique record number, and corresponds to a specific report of usage of a particular pesticide. The record number serves as the primary key, and links to the three result tables "RIs REXTOX", "RIs ADSCOR", and "RIs SYSCOR". All codes within the records in Table 2.7 correspond to their definitions in Tables 2.1 through 2.5 explained above.

The Basic Area Treated (BAT) is the area within the region treated at least once with the particular pesticide. For example, if there are 1000 hectares planted to potatoes in a region, and half this area is treated at least once with the same fungicide against *Phytophthora inf.*, then the BAT for the use of this pesticide on potatoes in this region is 500 hectares.) The unit of measurement is up to the user but should be the same for each record in this table (e.g., hectares or acres or thousand hectares or thousand acres.)

The Average Frequency of Treatment (AFT) is the average number of applications per season to the basic area treated. Since it only applies to the BAT, which by definition has been treated at least once, the value of AFT must be >= 1. For example, if the 500 hectares in the example above for BAT were treated an average of three times, the AFT for the use of this pesticide on potatoes in this region is 3.0.)

The Cumulative Area Treated (CAT) need not be separately entered. It is calculated by the macros according to the formula CAT = BAT * AFT.

The Recommended Dose Rate (RDR) is intended to accommodate cases where no actual pesticide usage data are collected, and usage must be estimated from sales data. In that case RDR reflects an estimated average of the label dose rates for all possible applications of the pesticide in that particular crop at that particular crop stage. When dose rates are estimated, the value of RDR is duplicated in the field for ADR.

The Actual Dose Rate (ADR) is based on actual usage data, and is strongly preferred when the usage data are available.

Practically speaking, the two dose rates need not be distinguished. When the REXTOX indicator is used to calculate the "Specific Risk Potential" as an unscaled theoretical index, it uses the RDR, but in general the Indicators use either dose rate estimations or a measured dose rate. The next version of the software should put either actual or recommended dose rates into a single place and name it 'Dose Rate' (DOR). This change requires some modifications of the macros and the corresponding result tables and has not yet been made.

2.8 Breakpoint tables

Table 2.8a: Breakp_ADSCOR

Score	BP_ADR	BP_AFT	BP_WI	BP_DT50W	BP_DT50S	BP_PHLW	BP_logKow	BP_logKoc
Case:	Is >	Is >	Is >	Is >	Is >	Is >	Is >	Is <
4	10	1E+12	1E+12	1E+12	1E+12	1E+12	1E+12	-1E+12
3	3	1E+12	1E+12	1E+12	1E+12	1E+12	1E+12	-1E+12
2	1	3	1E+12	183	183	1E+12	3	1.2957
1	0.1	1.1	0.2	60	60	5	2	2.3109
0	0	0	0	0	0	0	0	1E+12

Table 2.8b: Breakp_SYSCOR

Score	BP_ADR	BP_CAT	BP_WI	BP_SOL	BP_DT50W	BP_DT50S	BP_logKd
Case:	Is >	Is >	Is >	Is >	Is >	Is >	Is <
2	0.5	3500	1E+12	100	60	183	1
1	0.1	450	0.2	1	30	60	4
0	0	0	0	0	0	0	1E+12

These two tables show the breakpoints for each variable used in the scoring models. For example, the second column of the breakpoint table for ADSCOR shows the breakpoints for the Average Dose Rate (ADR). It is read in the following way:

Score = 4 if ADR > 10
Score = 3 if 3 < ADR ≤ 10
Score = 2 if 1 < ADR ≤ 3
Score = 1 if 0.1 < ADR ≤ 1
Score = 0 if ADR ≤ 0.1

Please note that the direction of comparison is reversed in the last column in both tables. The extreme values 1E+12 and –1E+12 are artificial breakpoints, provided to complete the tables with real numbers. They will never be reached by the real parameters to be scored.

Indicator users must change the values of the breakpoints—especially those for dose rates and areas and water indices--to suit the distribution of their own data. The maximum number of breakpoints is limited to 5 for ADSCOR and to 3 for SYSCOR (including the score 0). To add more permissible scores would require changes in the programme module "update_new".

3. Running the Indicators

The indicators are activated by the corresponding Access-form "Running indicators" which is linked with three macros. These macros use the same "update_new" module, written in Visual Basic, as the essential kernel of all indicator calculations. Its source code is provided in the appendix. This module links to two specific virtual query tables ("qUsage Ratio", used for REXTOX, and "qUsage Scores", used for both ADSCOR and SYSCOR), which provide all the variables needed for the calculations and store the calculated indices resulting from execution of the procedure.

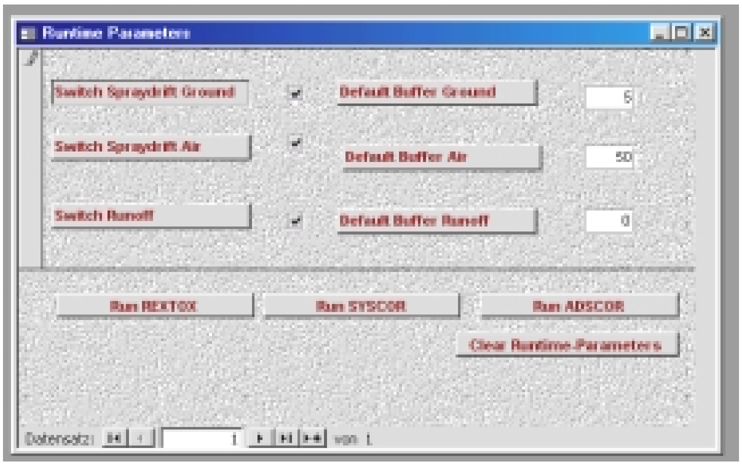
The input variables found in the two virtual tables are listed below.

Table 3: Virtual Query Table Input Variables

qUsage Ratio	qUsage Scores
Record	Record
Crop Type	
Crop Code	
Stage Code	
Method Code	Method Code
BAT	BAT
AFT	AFT
RDR	
ADR	ADR
CAT	CAT
Spraydrift Buffer	Spraydrift Buffer
Used Spraydrift Buffer	Used Spraydrift Buffer
SB Compliance	SB Compliance
Runoff Buffer	Runoff Buffer
Used Runoff Buffer	Used Runoff Buffer
RB Compliance	RB Compliance
LogKow	LogKow
Pesticide Group Code	
DT50w	DT50w
DT50s	DT50s
	Photolysis
	Solubility
	Training Required
Precipitation	
Water Index	Water Index
OC%	OC%
Soil Type	
Water Depth	
Slope	
Algal Acute Tox	Algal Acute Tox
Daphnid Acute Tox	Daphnid Acute Tox
Fish Acute Tox	Fish Acute Tox
Algal LongTerm Tox	Algal LongTerm Tox
Daphnid LongTerm Tox	Daphnid LongTerm Tox
Fish LongTerm Tox	Fish LongTerm Tox

The form "Running indicators" is used to enter run-time parameters based on general assumptions on spray drift and runoff buffer sizes. This can provide valuable insight into the sensitivity of the models to buffer size. Note that run-time parameters do not overwrite any data on actual buffer sizes entered in the Usage Table. A user may decide to calculate the indices using either actual values for buffers or temporary values; this decision is recorded by means of switches within the "Running indicators" form. Figure 2 shows the design of the form "Running indicators". In this example all three switches are set "on", with a default spray drift buffer width of 5 meters for ground spraying, 50 meters for aerial or air blast spraying, and no runoff buffer.

Figure 2: Setting runtime parameters within form "Running indicators"



4. Aggregation of results

The basic results of indicator calculation are stored in three Access tables. Each record in the Usage Table corresponds with one record in each result table (assuming all three indicators have been activated.) The terms of the indicator indices are given in the table below.

Table 4: Result Tables

	RIs REXTOX	RIs ADSCOR	RIs SYSCOR
Record number (primary key)	Record*	Record*	
Specific Acute Risk Potential	SARpot Algae SARpot Daphnia SARpot Fish		
Specific LongTerm Risk Potential	SLRpot Algae SLRpot Daphnia SLRpot Fish		
Specific Acute Risk Intensity (unscaled index)	SARint Algae SARint Daphnia SARint Fish	AAUA** AAUD AAUF	
Specific LongTerm Risk Intensity (unscaled index)	SLRint Algae SLRint Daphnia SLRint Fish	ALUA ALUD ALUF	
Acute Risk Index (scaled index)	ARI Algae ARI Daphnia ARI Fish	AASA AASD AASF	SASA*** SASD SASF
LongTerm Risk Index (scaled index)	LRI Algae LRI Daphnia LRI Fish	ALSA ALSD ALSF	
Loss via Spray Drift	LSD		
Loss via Run-Off	LRO		
Short-term Exposure Unscaled	SEU	ASEU****	
Short-term Exposure Scaled	SES	ASES	SSES*****
Long-term Exposure Unscaled	LEU	ALEU	
Long-term Exposure Scaled	LES	ALES	

* Record number corresponding with the record number in the Usage Table
** AAUD stands for: Additive Acute Unscaled Algae etc.
*** SASA stands for: SIRIS Acute Scaled Algae etc.
**** ASEU stands for: Additive Short-term Exposure Unscaled etc.
***** SSES stands for SIRIS Short-term Exposure Scaled

Unscaled risk indices are aggregated by calculating the mean of all included individual values; scaled indices are aggregated by calculating the sum of individual values.

The basic version of the indicator software provides for two types of aggregation:

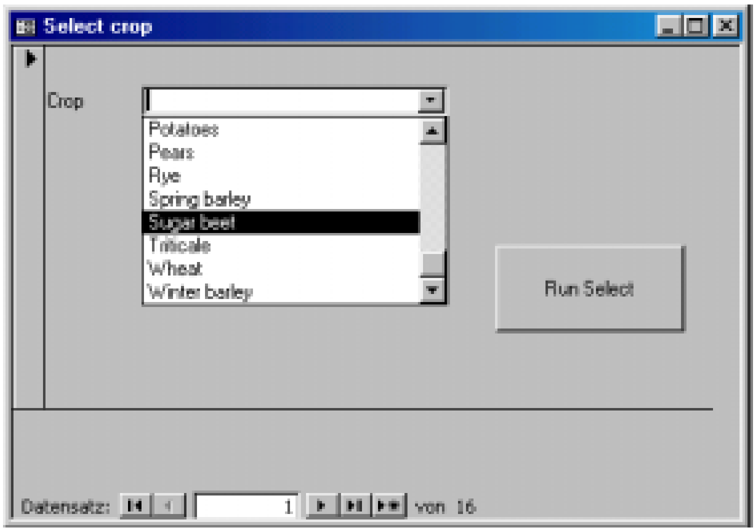
- Aggregation by year and crop group (the default setting: arable crops, fruit)
- Aggregation by year and selected crop

The results of the default aggregation can be found in the query tables:

- Results REXTOX by year_arable crops
- Results ADSCOR by year_arable crops
- Results SYSCOR by year_arable crops
- Results REXTOX by year_fruit
- Results ADSCOR by year_fruit
- Results SYSCOR by year_fruit

To aggregate by a specific crop the user must activate the Access-form "Select_crop". This form is linked to the crop table with a macro to allow a selection from a menu bar (see figure 3).

Figure 3: Selecting crop for aggregations within form "Select_crop"



The aggregated indices calculated for individual crops are stored in tables:

- REXTOX result subset 1
- ADSCOR result subset 1
- SYSCOR result subset 1

4. Presentation of results

Indicator results are displayed using Excel spreadsheets directly linked to the Access tables that store the aggregated indices. This allows easy use of the graphic tools of Excel, and easy export of the resulting diagrams to word processing systems for inclusion in reports. Only a few examples from among the wide range of possible graphic presentations have been included in the basic software, but users may add others by following the principles sketched below.

The key principles are the direct linkage of EXCEL tables to corresponding ACCESS tables and the pre-defined linkage of each EXCEL table to appropriate graphic outputs. A user need only click on the execution button provided by the Excel system to update all data and graphics after a new run of the indicators or a new selection of results of a particular crop.

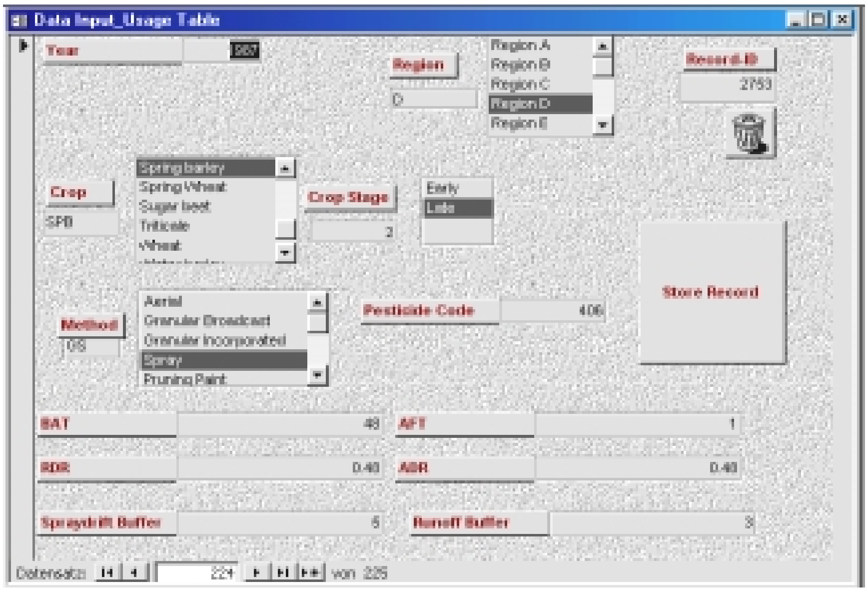
The examples given here address all three indicators. The Excel-spreadsheets “REXTOX_arable crops.xls”, “ADSCOR_arable crops” and “SYSCOR_arable crops” are linked with the corresponding result query tables of aggregation level “ … by year_arable crops”.

The second type of spreadsheets “REXTOX_particular crop.xls”, “ADSCOR_particular crop.xls” and “SYSCOR_particular crop.xls” corresponds with the results of a selected crop that are stored in the related tables “ … result subset 1”.

5. Usage Data Input

The form “Data Input_Usage Table” was developed to support input of usage data.

Figure 4: Usage Data Input Form

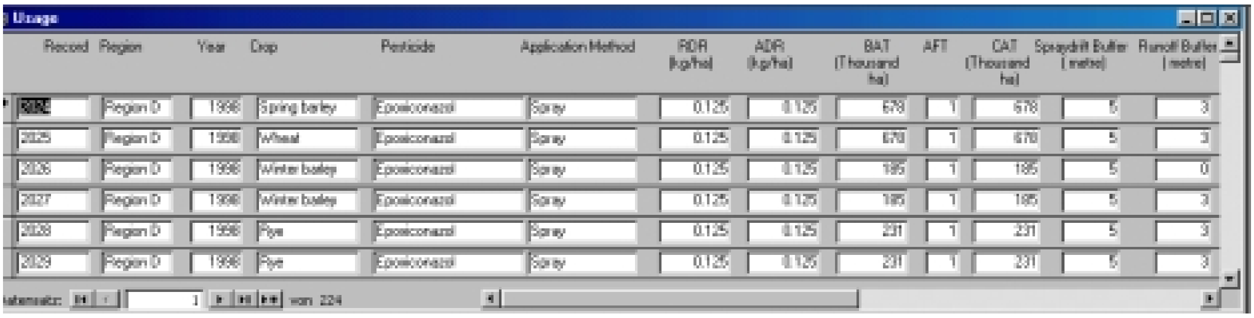


This form permits a manual record-by-record input of the pesticide usage data. For some elements the form prompts a selection from a menu bar (Region, Crop, Crop Stage, Method); for others the numerical value must be entered (Year, Pesticide Code, BAT, AFT, RDR, ADR, Buffer widths). Please note that the usage data entry form does **not** include checks for plausibility.

When data entry is complete, the user pushes the “Store Record” button and the data are posted to the usage table. The same form can also be used to delete a previously entered usage record. When a usage record is deleted from the table, any indicator results already calculated for that record are also automatically deleted.

Finally, the form “View_Usage Table” makes it easy to inspect the data in the usage table.

Figure 5: View Usage Data Form



Usage

Record	Region	Year	Crop	Pesticide	Application Method	RDR (kg/ha)	ADR (kg/ha)	BAT (Thousand ha)	AFT	CAT (Thousand ha)	Spraydrift Buffer (metre)	Runoff Buffer (metre)
[illegible]	Region D	1998	Spring barley	Epoxiconazol	Spray	0.125	0.125	678	1	678	5	3
2025	Region D	1998	Wheat	Epoxiconazol	Spray	0.125	0.125	678	1	678	5	3
2026	Region D	1998	Winter barley	Epoxiconazol	Spray	0.125	0.125	185	1	185	5	0
2027	Region D	1998	Winter barley	Epoxiconazol	Spray	0.125	0.125	185	1	185	5	3
2028	Region D	1998	Rye	Epoxiconazol	Spray	0.125	0.125	231	1	231	5	3
2029	Region D	1998	Rye	Epoxiconazol	Spray	0.125	0.125	231	1	231	5	3

Datensatz: 1 von 224

APPENDIX
Source Code of Module ‘ update_new’

```
Option Compare Database
Option Explicit

Public Function UpdateRIs_Rextox()

Dim db As Database, U As Recordset, V As Recordset
Dim LSD#, LRO#, Q#, P#, f1#, f2#, S#, OC#, Kd#, SBC#, RBC#
Dim SARpot1#, SLRpot1#, AAR#, BATWI#, CUM#, SB#, RB#
Dim SwitchG#, SwitchA#, SwitchR#, SBG#, SBA#, RBU#
Dim Loss#, EXpot#, EXunsca#, EXsca#, LTF#
Dim alpha#, kmax#, PA#

DoCmd.SetWarnings False
DoCmd.OpenQuery "Delete RIs REXTOX"
DoCmd.SetWarnings True

Set db = CurrentDb()
Set U = db.OpenRecordset("qUsage Ratio")
Set V = db.OpenRecordset("Runtime Parameters")
If (V.BOF And V.EOF) <> True Then
V.MoveLast
SwitchG = V![Switch Spraydrift Ground]
SwitchA = V![Switch Spraydrift Air]
SwitchR = V![Switch Runoff Buffer]
Else
SwitchG = 1
SwitchA = 1
SwitchR = 1
End If




U.MoveFirst
Do Until U.EOF


 'Calculate LSD:
   SBC = U![SB Compliance]
   Select Case U![Method Code]
      Case "GS"
      SB = U![Spraydrift Buffer]
      If SwitchG = -1 Then 'in this case user want to take an default buffer size for all ground applications
      SB = V![Default Buffer Ground]
      End If

      If SB < 0.5 Then SB = 0.5 'the minimum distance between the last nozzle and the water
      'body is set to 0.5 m

         Select Case U![Stage Code]
            Case "1"
               LSD = (((0.32397 + (1.73651 / SB)) ^ 2) * SBC) + (14.42 * (1 - SBC))
            Case "2"
               LSD = ((Exp((1.58074 - (1.17817 * Log(SB))))) * SBC) + (10.99 * (1 - SBC))
            Case Else
               LSD = 0
         End Select
```

```
    Case "AB"

    SB = U![Spraydrift Buffer]
    If SwitchA = -1 Then 'in this case user want to take an default buffer size for all air blast applications
    SB = V![Default Buffer Air]
    End If

    If SB < 0.5 Then SB = 0.5 'see above

      Select Case U![Crop Type] & U![Stage Code]
        Case "FT1"
          LSD = (((7.08298 - (1.68712 * Log(SB))) ^ 2) * SBC) + (68.1 * (1 - SBC))
        Case "FT2"
          LSD = ((1 / (0.03597 + (0.00179 * (SB ^ 2)))) * SBC) + (27.46 * (1 - SBC))
        Case "VI1"
          LSD = (((0.25614 + (4.79664 / SB)) ^ 2) * SBC) + (97.02 * (1 - SBC))
        Case "VI2"
          LSD = (((4.21807 - (1.20411 * Log(SB))) ^ 2) * SBC) + (25.53 * (1 - SBC))
        Case "HO1"
          LSD = ((Exp(5.26822 - 1.15262 * Sqr(SB))) * SBC) + (85.9 * (1 - SBC))
        Case "HO2"
          LSD = ((Exp(3.11768 - 0.09246 * SB)) * SBC) + (21.57 * (1 - SBC))

        Case Else
          Select Case U![Stage Code]
            Case "1"
              LSD = (((0.32397 + (1.73651 / SB)) ^ 2) * SBC) + (14.42 * (1 - SBC))
            Case "2"
              LSD = ((Exp((1.58074 - (1.17817 * Log(SB))))) * SBC) + (10.99 * (1 - SBC))
            Case Else
              LSD = 0
        End Select

      End Select

    Case "AE"

          LSD = 100


    Case Else
      LSD = 0
  End Select



'Calculate LRO:
    RB = U![Runoff Buffer]
    If SwitchR = -1 Then 'in this case users want to take an default run-off protection buffer for all applications
    RB = V![Default Buffer Runoff]
    End If




  Select Case U![Method Code]
    Case "GS", "AB", "AE", "GB"
      P = U![precipitation]
      Select Case U![Pesticide Group Code]
        Case "HE", "MO", "NE", "SS"
          f2 = 0
```

```
                alpha = 0.03475
                Select Case U![Soil Type]
                    Case "Loamy": kmax = 0.93
                    Case "Sandy": kmax = 0.71
                End Select
            Case Else
                Select Case U![Stage Code]
                    Case "1"
                        f2 = 0.2
                        alpha = 0.03451
                    Case "2"
                        f2 = 0.7
                        alpha = 0.01035
                End Select

                Select Case U![Crop Type]
                    Case "CE", "OI", "FP"
                        Select Case U![Soil Type]
                            Case "Loamy": kmax = 0.85
                            Case "Sandy": kmax = 0.54
                        End Select
                    Case Else
                        Select Case U![Soil Type]
                            Case "Loamy": kmax = 0.93
                            Case "Sandy": kmax = 0.62
                        End Select
                End Select
        End Select


        PA = 7.62 * (1 / kmax - 1)
        Q = (P - PA) * kmax + (kmax / alpha) * (Exp(-alpha * (P - PA)) - 1)
        S = U![slope]
        If S < 20 Then
            f1 = 0.02153 * S + 0.001423 * S ^ 2
        ElseIf S >= 20 Then
            f1 = 1
        End If
        OC = U![OC%]
        Kd = (10 ^ (1.09 + (U!logkow * 0.47))) * OC / 100 'estimated Kd
        LRO = Q / P * f1 * f2 * Exp(-3 * Log(2) / U![DT50S]) * 100 / (1 + Kd) 'first step without f3
        LRO = LRO * (U![RB Compliance] * (0.83 ^ RB - 1) + 1) 'include f3 and compliance
    Case Else
        LRO = 0
End Select


Loss = (LSD + LRO) / 100 / U![Water Depth]
EXpot = U![RDR] * Loss * U![Water Index]
EXunsca = U![ADR] * U![AFT] * Loss * U![Water Index] 'this calculation is not quite correct. '
'To get a correct transformation from dose rate, measured in kg/hectare, into concentration,
'measured in mg/liter, the term has to be divided by 10 !! But to stay in accord with the formulae
'given in the indicator-report a correction was not made here.

EXsca = EXunsca * U![BAT]
If U![DT50w] > 0 Then
    LTF = U![DT50w] / 14.5560907917589 * (1 - Exp(-14.5560907917589 / U![DT50w]))
Else
    LTF = 0
End If

CUM = U![BAT] * U![AFT]
```

```
With U
   .Edit
   ![SARpot Algae] = EXpot / U![Algal Acute Tox]
   ![SARpot Daphnia] = EXpot / U![Daphnid Acute Tox]
   ![SARpot Fish] = EXpot / U![Fish Acute Tox]

   ![SARint Algae] = EXunsca / U![Algal Acute Tox]
   ![SARint Daphnia] = EXunsca / U![Daphnid Acute Tox]
   ![SARint Fish] = EXunsca / U![Fish Acute Tox]

   ![ARI Algae] = EXsca / U![Algal Acute Tox]
   ![ARI Daphnia] = EXsca / U![Daphnid Acute Tox]
   ![ARI Fish] = EXsca / U![Fish Acute Tox]

   ![SLRpot Algae] = (EXpot * LTF) / U![Algal LongTerm Tox]
   ![SLRpot Daphnia] = (EXpot * LTF) / U![Daphnid LongTerm Tox]
   ![SLRpot Fish] = (EXpot * LTF) / U![Fish LongTerm Tox]

   ![SLRint Algae] = (EXunsca * LTF) / U![Algal LongTerm Tox]
   ![SLRint Daphnia] = (EXunsca * LTF) / U![Daphnid LongTerm Tox]
   ![SLRint Fish] = (EXunsca * LTF) / U![Fish LongTerm Tox]

   ![LRI Algae] = (EXsca * LTF) / U![Algal LongTerm Tox]
   ![LRI Daphnia] = (EXsca * LTF) / U![Daphnid LongTerm Tox]
   ![LRI Fish] = (EXsca * LTF) / U![Fish LongTerm Tox]

   If LTF = 0 Then
      ![SLRpot Algae] = Empty
      ![SLRpot Daphnia] = Empty '(EXpot * LTF) / U![Daphnid LongTerm Tox]
      ![SLRpot Fish] = Empty '(EXpot * LTF) / U![Fish LongTerm Tox]

      ![SLRint Algae] = Empty '(EXunsca * LTF) / U![Algal LongTerm Tox]
      ![SLRint Daphnia] = Empty '(EXunsca * LTF) / U![Daphnid LongTerm Tox]
      ![SLRint Fish] = Empty '(EXunsca * LTF) / U![Fish LongTerm Tox]

      ![LRI Algae] = Empty '(EXsca * LTF) / U![Algal LongTerm Tox]
      ![LRI Daphnia] = Empty '(EXsca * LTF) / U![Daphnid LongTerm Tox]
      ![LRI Fish] = Empty '(EXsca * LTF) / U![Fish LongTerm Tox]
   End If
   'this if loop was introduced to handle missing data in the pesticide properties table
   'So the indices that need these values for calculation also are empty in the result tables

   ![LSD] = LSD
   ![LRO] = LRO
   ![CAT] = CUM
   ![Used Spraydrift Buffer] = SB
   ![Used Runoff Buffer] = RB

   'exposure indices
   ![SEU] = EXunsca
   ![SES] = EXsca
   ![LEU] = EXunsca * LTF
   ![LES] = EXsca * LTF

   If LTF = 0 Then
      ![LEU] = Empty 'EXunsca * LTF
      ![LES] = Empty 'EXsca * LTF
   End If
   'see above


   .Update
```

```
        .MoveNext
    End With
Loop

End Function


Public Function UpdateRIs_adscor()

Dim db As Database, U As Recordset, V As Recordset, B As Recordset
Dim i%, j%, SPA&(4860)
Dim AAUES#, ALUES#, CUM#, SB#, RB#
Dim SwitchG#, SwitchA#, SwitchR#, SBG#, SBA#, RBU#

Dim BP#(5, 9), WI#
Dim MC$, AFT#, AG#, AT#, BAT#, OC#, LogKd#, logkow#

DoCmd.SetWarnings False
DoCmd.OpenQuery "Delete RIs ADSCOR"

DoCmd.SetWarnings True

Set db = CurrentDb()



'reading in break points
Set B = db.OpenRecordset("Breakp_Adscor")
B.MoveFirst
For i = 1 To 5
    BP(i, 0) = B!Score
    BP(i, 1) = B!BP_ADR
    BP(i, 2) = B!BP_AFT
    BP(i, 3) = B!BP_WI
    BP(i, 4) = B!BP_DT50W
    BP(i, 5) = B!BP_DT50S
    BP(i, 6) = B!BP_PHLW
    BP(i, 7) = B!BP_logKow
    BP(i, 8) = B!BP_logKoc
    B.MoveNext
Next i
B.Close

Set U = db.OpenRecordset("qUsage Scores")
Set V = db.OpenRecordset("Runtime Parameters")
If (V.BOF And V.EOF) <> True Then
V.MoveLast
SwitchG = V![Switch Spraydrift Ground]
SwitchA = V![Switch Spraydrift Air]
SwitchR = V![Switch Runoff Buffer]
Else
SwitchG = 1
SwitchA = 1
SwitchR = 1
End If



U.MoveFirst
Do Until U.EOF

'Calculate the Scores:
```

```
MC = U![Method Code]
SB = U![Spraydrift Buffer]
        If SwitchG = -1 Then
        SB = V![Default Buffer Ground]
        End If
RB = U![Runoff Buffer]
        If SwitchR = -1 Then
        RB = V![Default Buffer Runoff]
        End If

Select Case MC
    Case "ST", "BC", "PT": AAUES = 0
    Case "SS", "GI":  AAUES = 1
    Case "GB":  AAUES = 2
    Case "GS":  AAUES = 3
    Case "AB":  AAUES = 4
    Case "AE":  AAUES = 4
    Case "WA":  AAUES = 4
    Case Else: MsgBox "Unmatched Method Code"
End Select

If SB > 1 Then

    Select Case MC
        Case "GS": AAUES = AAUES - U![SB Compliance]
        Case "AB", "AE": AAUES = AAUES - (2 * U![SB Compliance])
    End Select
End If

If RB > 0 Then
    Select Case MC
        Case "GB", "GS", "AB", "AE": AAUES = AAUES - U![RB Compliance]
    End Select
End If

Select Case U![ADR]
    Case Is > BP(1, 1):  AAUES = AAUES + 4
    Case Is > BP(2, 1):  AAUES = AAUES + 3
    Case Is > BP(3, 1):  AAUES = AAUES + 2
    Case Is > BP(4, 1):  AAUES = AAUES + 1
End Select

BAT = U![BAT]
AFT = U![AFT]


Select Case AFT
    Case Is > BP(1, 2):  AAUES = AAUES + 4
    Case Is > BP(2, 2):  AAUES = AAUES + 3
    Case Is > BP(3, 2):  AAUES = AAUES + 2
    Case Is > BP(4, 2):  AAUES = AAUES + 1
End Select
WI = U![Water Index]
Select Case WI
    Case Is > BP(1, 3):  AAUES = AAUES + 4
    Case Is > BP(2, 3):  AAUES = AAUES + 3
    Case Is > BP(3, 3):  AAUES = AAUES + 2
    Case Is > BP(4, 3):  AAUES = AAUES + 1
End Select
```

```
AAUES = AAUES + 1
ALUES = AAUES

logkow = U!logkow
'Bioaccumulation
Select Case logkow
   Case Is > BP(1, 7):  ALUES = ALUES + 4
   Case Is > BP(2, 7):  ALUES = ALUES + 3
   Case Is > BP(3, 7):  ALUES = ALUES + 2
   Case Is > BP(4, 7):  ALUES = ALUES + 1
End Select

Select Case U![DT50w]
   Case Is > BP(1, 4):  ALUES = ALUES + 4
   Case Is > BP(2, 4):  ALUES = ALUES + 3
   Case Is > BP(3, 4):  ALUES = ALUES + 2
   Case Is > BP(4, 4):  ALUES = ALUES + 1
End Select


Select Case U![DT50S]
   Case Is > BP(1, 5):  ALUES = ALUES + 4
   Case Is > BP(2, 5):  ALUES = ALUES + 3
   Case Is > BP(3, 5):  ALUES = ALUES + 2
   Case Is > BP(4, 5):  ALUES = ALUES + 1
End Select


Select Case U![Photolysis]
   Case Is > BP(1, 6):  ALUES = ALUES + 4
   Case Is > BP(2, 6):  ALUES = ALUES + 3
   Case Is > BP(3, 6):  ALUES = ALUES + 2
   Case Is > BP(4, 6):  ALUES = ALUES + 1
End Select

'Run-off potential, attention the direction of comparison did change here from > to < !
Select Case logkow
   Case Is < BP(1, 8):  ALUES = ALUES + 4
   Case Is < BP(2, 8):  ALUES = ALUES + 3
   Case Is < BP(3, 8):  ALUES = ALUES + 2
   Case Is < BP(4, 8):  ALUES = ALUES + 1
End Select



CUM = U![BAT] * U![AFT]

With U
   .Edit
   ![AAUA] = AAUES / ![Algal Acute Tox]
   ![AAUD] = AAUES / ![Daphnid Acute Tox]
   ![AAUF] = AAUES / ![Fish Acute Tox]
   ![AASA] = ![AAUA] * BAT
   ![AASD] = ![AAUD] * BAT
   ![AASF] = ![AAUF] * BAT
   ![ALUA] = ALUES / ![Algal LongTerm Tox]
   ![ALUD] = ALUES / ![Daphnid LongTerm Tox]
   ![ALUF] = ALUES / ![Fish LongTerm Tox]
   ![ALSA] = ![ALUA] * BAT
   ![ALSD] = ![ALUD] * BAT
```

```
    ![ALSF] = ![ALUF] * BAT
    ![CAT] = CUM
    ![Used Spraydrift Buffer] = SB
    ![Used Runoff Buffer] = RB
    ![Used Spraydrift Buf] = SB
    ![Used Runoff Buf] = RB
    'Exposure indices
    ![ASEU] = AAUES
    ![ASES] = AAUES * BAT
    ![ALEU] = ALUES
    ![ALES] = ALUES * BAT

    .Update
    .MoveNext
End With

Loop

End Function



Public Function UpdateRIs_Syscor()

Dim db As Database, U As Recordset, rs As Recordset, V As Recordset, B As Recordset
Dim S%(7, 2), i%, j%, SPA&(4860)
Dim SASES&, CUM#, SB#, RB#
Dim SwitchG#, SwitchA#, SwitchR#, SBG#, SBA#, RBU#

Dim BP#(5, 8), WI#
Dim MC$, AFT#, AG#, AT#, BAT#, OC#, LogKd#, logkow#
Dim Rank&, Penalty&

DoCmd.SetWarnings False
DoCmd.OpenQuery "Delete RIs SYSCOR"
DoCmd.SetWarnings True

Set db = CurrentDb()

Set rs = db.OpenRecordset("SIRIS Penalties")
rs.MoveFirst
For i = 1 To 4860
    SPA(i) = rs!Penalty
    rs.MoveNext
Next i
rs.Close

S(1, 1) = 324
S(2, 1) = 162
S(3, 1) = 54
S(4, 1) = 18
S(5, 1) = 6
S(6, 1) = 2
S(7, 1) = 1

'reading in break points
Set B = db.OpenRecordset("Breakp_Syscor")
B.MoveFirst
For i = 1 To 3
    BP(i, 0) = B!Score
    BP(i, 1) = B!BP_ADR
    BP(i, 2) = B!BP_CAT
```

```
    BP(i, 3) = B!BP_WI
    BP(i, 4) = B!BP_SOL
    BP(i, 5) = B!BP_DT50W
    BP(i, 6) = B!BP_DT50S
    BP(i, 7) = B!BP_logKd
    B.MoveNext
Next i
B.Close

Set U = db.OpenRecordset("qUsage Scores")
Set V = db.OpenRecordset("Runtime Parameters")
If (V.BOF And V.EOF) <> True Then
V.MoveLast
SwitchG = V![Switch Spraydrift Ground]
SwitchA = V![Switch Spraydrift Air]
SwitchR = V![Switch Runoff Buffer]
Else
SwitchG = 1
SwitchA = 1
SwitchR = 1
End If



U.MoveFirst
Do Until U.EOF

'Calculate the Scores:

For i = 1 To 7
    S(i, 2) = 0
Next

MC = U![Method Code]
SB = U![Spraydrift Buffer]
        If SwitchG = -1 Then
        SB = V![Default Buffer Ground]
        End If
RB = U![Runoff Buffer]
        If SwitchR = -1 Then
        RB = V![Default Buffer Runoff]
        End If

Select Case MC

    Case "SS", "GI": S(1, 2) = 2
    Case "GB": S(1, 2) = 4
    Case "GS": S(1, 2) = 6
    Case "AB": S(1, 2) = 8
    Case "AE": S(1, 2) = 8
    Case "WA": S(1, 2) = 10
    Case Else: MsgBox "Unmatched Method Code"
End Select

If SB > 1 Then
    S(1, 2) = 2 'equiv of old category, but now going up in twos

End If



Select Case U![ADR]
```

```
    Case Is > BP(1, 1): S(1, 2) = S(1, 2) + 2
    Case Is > BP(2, 1): S(1, 2) = S(1, 2) + 1
End Select

BAT = U![BAT]
AFT = U![AFT]
Select Case BAT * AFT
    Case Is > BP(1, 2): S(1, 2) = S(1, 2) + 2
    Case Is > BP(2, 2): S(1, 2) = S(1, 2) + 1
End Select

WI = U![Water Index]

Select Case WI
    Case Is > BP(1, 3): S(2, 2) = S(2, 2) + 2
    Case Is > BP(2, 3): S(2, 2) = S(2, 2) + 1
End Select


Select Case U![Solubility]
    Case Is > BP(1, 4): S(3, 2) = 2
    Case Is > BP(2, 4): S(3, 2) = 1
End Select


logkow = U!logkow


Select Case U![DT50w]

    Case Is > BP(1, 5): S(4, 2) = 2
    Case Is > BP(2, 5): S(4, 2) = 1

End Select

OC = U![OC%]

LogKd = 1.09 + (0.47 * logkow) + (Log(OC / 100) / Log(10)) 'estimated LogKd
Select Case LogKd
    Case Is < BP(1, 7): S(5, 2) = 2
    Case Is < BP(2, 7): S(5, 2) = 1
End Select

Select Case U![DT50S]
    Case Is > BP(1, 6): S(6, 2) = 2
    Case Is > BP(2, 6): S(6, 2) = 1
End Select

If U![Training Required] = 0 Then S(7, 2) = 1



Rank = 1
For i = 1 To 7
    Rank = Rank + (S(i, 1) * S(i, 2))
Next
SASES = SPA(Rank)
CUM = U![BAT] * U![AFT]

With U
    .Edit
```

```
    ![SASA] = SASES / ![Algal Acute Tox]
    ![SASD] = SASES / ![Daphnid Acute Tox]
    ![SASF] = SASES / ![Fish Acute Tox]
    ![CAT] = CUM
    ![Used Spraydrift Buffer] = SB
    ![Used Runoff Buffer] = RB
    ![Used Spraydrift Buf] = SB
    ![Used Runoff Buf] = RB
    'Exposure indices

    ![SSES] = SASES

    .Update
    .MoveNext
End With

Loop

End Function
```