Aggregate Exposure Assessment — An Approach

Working Draft

Prepared for:

Subcommittee on Aggregate Exposure Assessment,
Health and Environmental Sciences Institute, ILSI

by:

Edmund Crouch, Ph.D., and David Burmaster, Ph.D.
Cambridge Environmental Inc.
58 Charles Street, Cambridge, MA 02141
617-225-0810; FAX 617-225-0813
info@CambridgeEnvironmental.com

January 30, 1998

Contents

1 Introduction

1.1 The tasks — architecture, implementation, and databases

The Food Quality Protection Act of 1996 (FQPA) requires an aggregate exposure assessment for pesticides, but provides little guidance on what that is or how it is to be implemented. We here address the problem of an "aggregate exposure assessment," in the risk-assessment context in which it will be used. We address the following three tasks.

A. Define a methodology that can handle any level of information.

We shall outline a methodology that could be used in the ideal situation where all information is available, or in various less ideal situations. We shall evaluate what information would be the most useful (*e.g.* would it be ideal to have blood levels of various pesticides in the U.S. population for a known set of residue levels?). We shall specify a hierarchy of information needs for responding to the specifications demanded. We shall design the methodology to allow use of any level of information within that hierarchy (if this is possible), and specify what information is essential under all circumstances.

The object here is to provide the broad outlines of such a methodology, tailored to the required specifications, that can handle any level of information availability (perhaps even including a set of defaults that can be used in the absence of certain information). This is the "architecture" — the specification of how to obtain what is required.

B. Define an implementation of the methodology.

We shall outline an implementation method for the methodology proposed in the previous task. This will be done in the broadest possible terms that nevertheless show that the methodology is practical. This is an "implementation" — a demonstration (in principle) of one practical approach that will fulfill the requirements of the methodology.

This task may overlap somewhat with the previous task; it can be difficult to separate the architecture from the implementation in some cases.

C. Evaluation of current databases, and specification of others that may be useful.

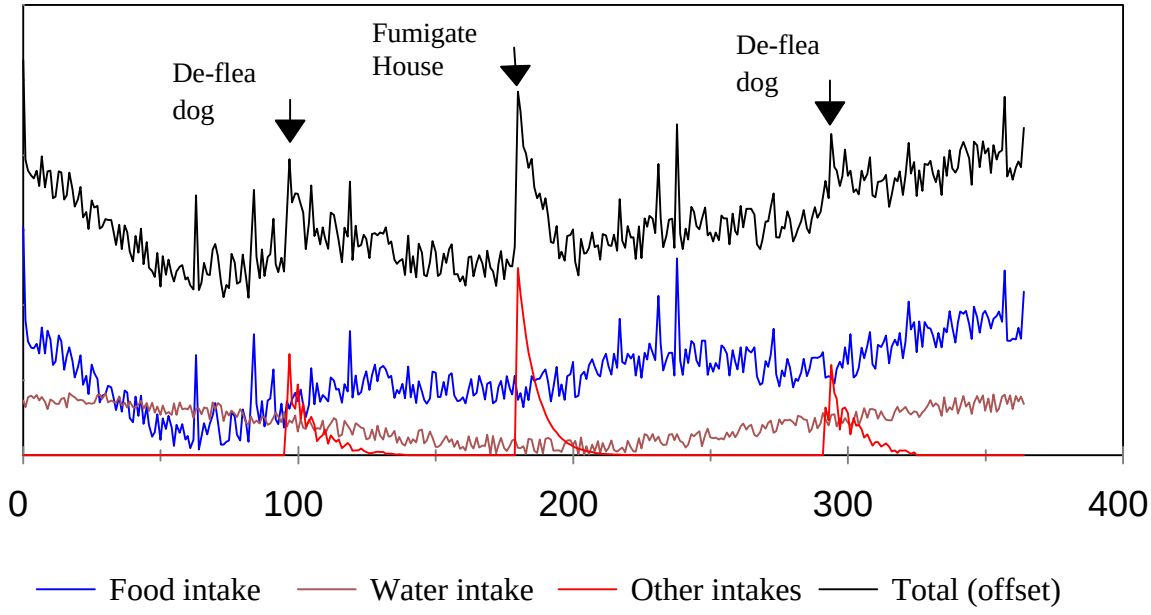
We shall examine the databases prepared for the current exercise, and evaluate how these fit into the hierarchy. We shall determine what further information or set of assumptions is required in order to directly use these databases, and what further information would add value to them. We

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

shall identify other databases that might be used to augment the information in the current set. Where necessary, we shall identify the techniques required to extrapolate from the current databases (*e.g.* what analysis or sampling techniques are available that maintain the correlation structure for all the non-detects?).

1.2 An outline of the implementation

Consider the expected pattern of day-to-day intake of pesticide — a year in the pesticide exposure of A.N. Other:



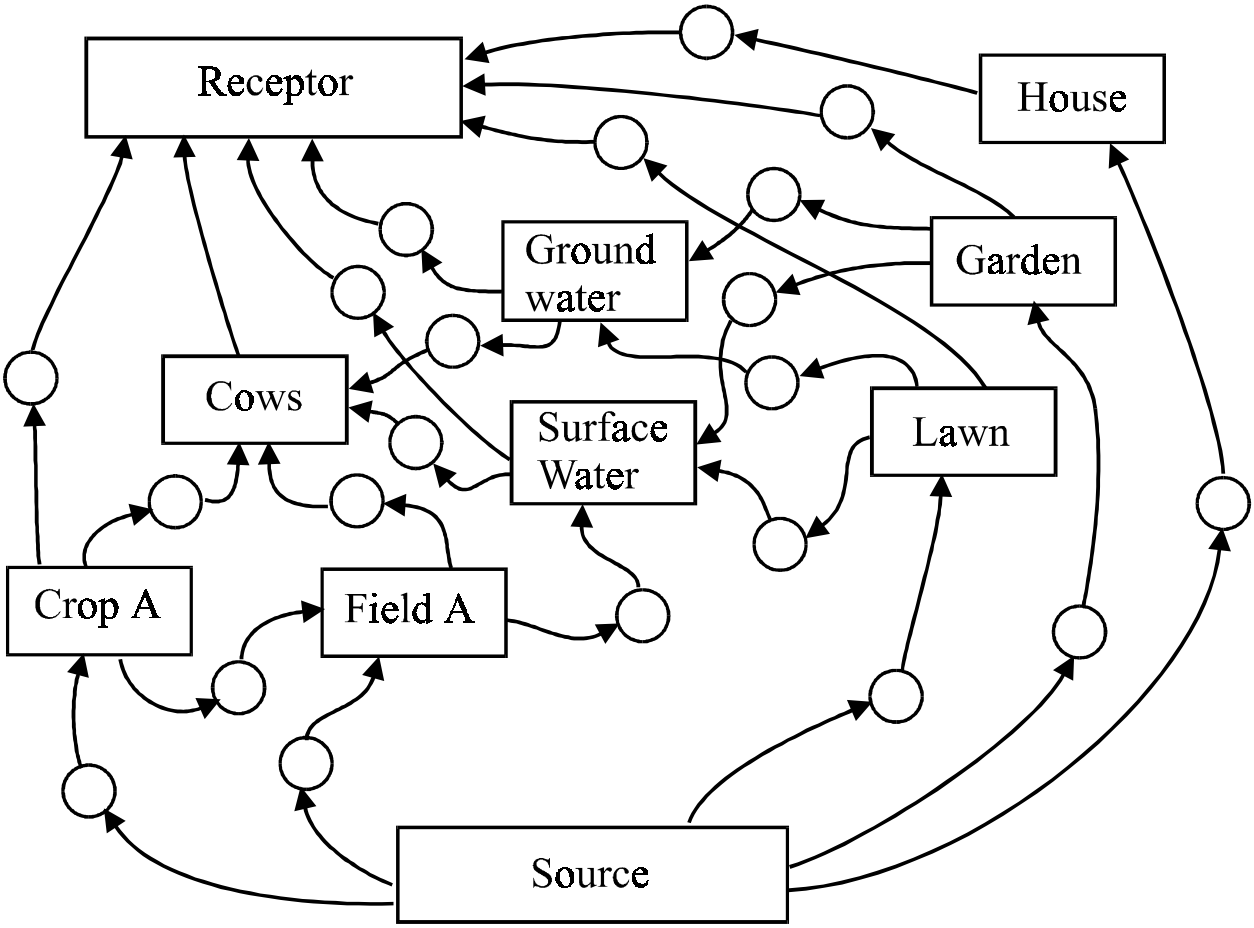
A.N. Other eats a fairly regular diet, but on weekends gets a craving (and every so often succumbs) for a food item that happens to have higher pesticide concentration. The regular diet has seasonal variations in pesticide concentrations, and random fluctuations from day-to-day. Similarly, the water has a seasonal variation, with random daily fluctuations (besides which, A.N. Other drinks different amounts of water from different locations each day). This particular year, the house was fumigated with this pesticide, and twice the dog was treated to remove fleas.

The architecture and implementation must be able to track these sort of variations in intakes throughout a lifetime, for many randomly chosen individuals, in order to determine distributions of worst-case daily intake, worst-case weekly intake, etc. They therefore have to handle modeling of varying degrees of complexity, always using distributions of values. The distributions are derived from widely differing sources and may be represented in widely differing ways (analytic, tabular, sample selection, combinations, and others, some of which are

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

likely not yet known). Where there is inadequate data, we have to be able to insert default values with no difficulty. The approach must be readily modified and extensible, without making any fundamental changes. No particular models, distribution types, or data sources may be hard-wired — all must be changeable without difficulty.

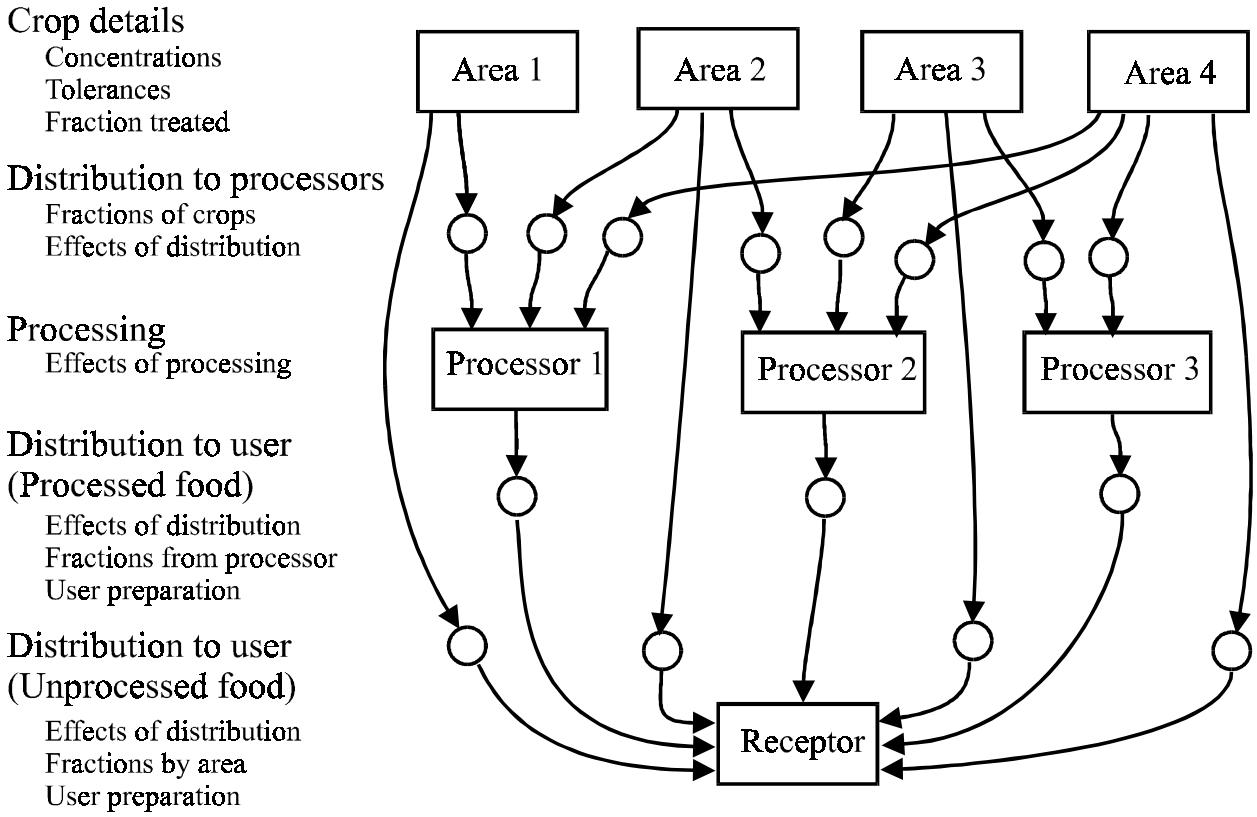
The approach taken is modular. Pathways of exposure may be diagrammatically represented by boxes = *pathlinks* (modeling real-world objects) and circles = *transporters* (modeling interactions between the real-world objects). Each box and circle translates into a computer object that may be programmed independently. Each computer box knows how it behaves or (circles) how two boxes interact — the boxes know how to calculate the concentrations of pesticide inside the corresponding real-world objects (given the inputs of pesticide to them), and the circles know how to calculate how much pesticide flows between the two real-world objects represented by the boxes they connect (given the concentrations in them).



The pathway structure and level of detail is not pre-conceived or in any way constrained by the implementation. The box-and-circle pattern is set up by a control file that calls on the implementation to construct a particular pattern. Changes in that control file are easy to incorporate, so that, for example, in the above pathways, it would be straightforward (given

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

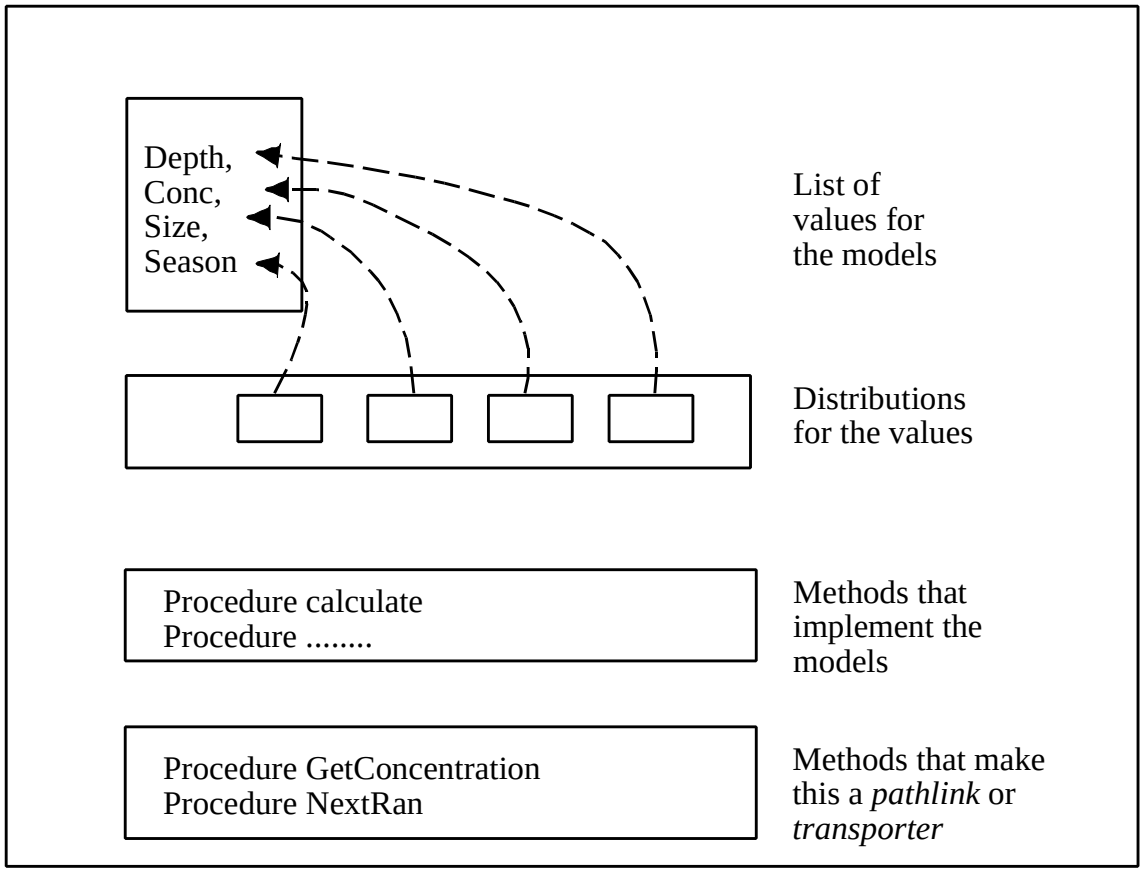
sufficient data) to expand the "Crop A" box, and the circle connecting it to the "Receptor" box to explicitly incorporate multiple geographic areas, together with distribution and processing steps:



Expanded food pathway, showing a possible breakdown into *pathlinks* and *transporters*, and the models or measurements handled by each component.

Where previously all the crop details were handled in a single "Crop A" box, now they are handled separately (allowing different representations in different areas, for example). Similarly, while the whole distribution and processing chain was incorporated into a single circle (*transporter*) previously, this chain is now explicitly separated into components to allow a more modular approach. Note that neither approach is "best" — the optimum in a given application will depend on the degree of detail available. While it would be possible to incorporate all the detail of the second (expanded) approach in the first, that might require a highly complex and very task-specific *transporter* object.

What are these objects? Basically, they are packets of data and subroutines that go together. Both *pathlinks* and *transporters* have the general form



All of these inherit certain properties from ancestor *pathlinks* and *transporters* — the functions that make these *pathlinks* or *transporters* — but they are free to override some of them. Thus a *pathlink* will usually override its inherited `GetConcentration` subroutine (method) in order to connect with the models associated with this particular *pathlink*.

There are several interesting consequences of the way this works. First, the implementation depends only on the existence of the subroutines (methods) that are inherited from the original ancestors — it does not know nor care what particular type of *pathlink* or *transporter* is used at any point. Consequently, when the implementation is defined, the particular pathlinks and transporters that are to be used in any particular application may not even have been written. The implementation is designed to allow one to add in new versions at any time, and use them as freely and in exactly the same way as those that are originally in place. Thus we can at any time add new models, or change old models. All that is required is to add them in and register their new names with the implementation (a process designed to be easy). Furthermore, much of the

behavior of new versions is already available — you simply inherit whatever is required from an already available version, and modify what needs changing.

Notice that all the values used in the models are associated with distributions. All the values may be updated by invoking the `Nextran` method of *pathlinks* or *transporters*. Actually, the implementation is much more subtle — only the values that are associated with particular types of distribution (*e.g.* variability versus true lack of knowledge) are updated when Nextran is updated, with the particular types of distribution that are updated is under control of the implementation's control structures.

Why do we need distributions? The implementation methodology does not require them — it can do calculations on fixed values just as easily (one option of the Monte Carlo control mechanisms can be to turn off all the distributions, and just use point estimates). But associated with every value is likely to be an uncertainty, and it may be important to know the uncertainty in the results. More important in the context of pesticide regulation, many relevant values are variable in the populations involved — different people eat different amounts of different foods. The results of any risk assessment for pesticides are thus going to be different for different population members, and we want to evaluate estimates for all members of the population — or at least find out by how must the estimates vary in the population . That makes two sorts of variability in values and results — true uncertainty, and differences between population members. However, it may be necessary to make finer distinctions between different sorts of uncertainty and/or variability, so that they can be handled differently in the implementation. A good example is afforded by EPA's treatment of the uncertainty and variability of toxicity reference values (reference doses, reference concentrations, unit risks, potency factors). At the moment, EPA treats these as fixed values, although everybody knows that there is huge uncertainty in them, and probably variability from person to person as well. The implementation described allows one to handle the uncertainty/variability of toxicity reference values separately from the uncertainties and variabilities of other values. Moreover, it also affords a way to evaluate the effects of such separate handling, and so to locate which uncertainties and/or variabilities are dominating the uncertainties/variabilities in the risk estimates.

The models that are used in all the *pathlinks* and *transporters* do not have to know anything about distributions. They just operate as usual on the fixed value fields in the pathlinks and transporters. The methods of the *pathlinks* and *transporters* are charged with implementing these models, but they can do so in any way they please — including writing out input data files for other programs, and reading the results of those other programs.

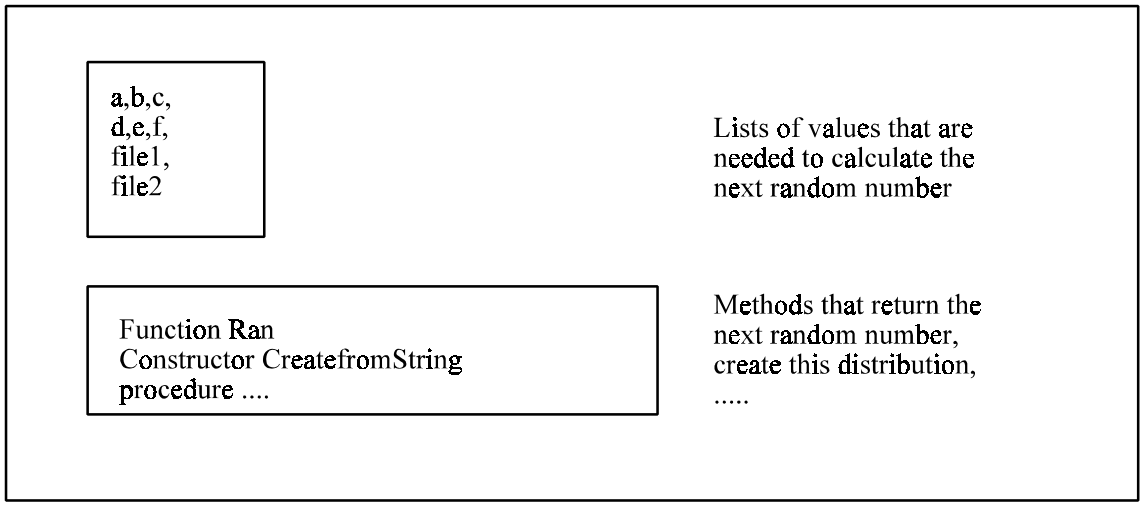
What are distributions here? The *pathlinks* and *transporters* neither know nor care what the distributions actually are in any particular case. All that matters is that these particular distributions are descended from an ancestor distribution object, and know how to return a random value when there `Ran` method is invoked.

Once again, the way in which particular distributions are realized is completely irrelevant to the implementation. When the implementation is put together, the distributions that will ultimately

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

be used may not even have been thought of, let alone coded. We can add distributions into the implementations in just the same way as *pathlinks* and *transporters* — once their name is registered (and the implementation is set up to do this), they are useable just as freely as any originally-defined distribution.

A distribution basically looks like (as usual, there are other components):



58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

The distribution has two essential attributes — it knows how to return the next random number from itself, and it knows how to create itself from a piece of text saying what it is. It is entirely up to the distribution how it obtains its next random value — the main program implementation never asks any questions about that. The distribution may be an analytic one — so that it can be specified by something like:

```
lognormal      1.5      0.7
```

meaning a lognormal distribution with median exp(1.5) and standard deviation of logarithms equal to 0.7. Provided a distribution with an external name "lognormal" and expecting two parameters has been registered with the implementation, the implementation, when presented with this line of text, will create an example of a lognormal distribution with median exp(1.5) and standard deviation of logarithms equal to 0.7.

Much more complicated distributions can be introduced just as easily. For example, it may be necessary to sample from a set of empirical data. Suppose we write a distribution that knows about this, so that its `Ran` method can handle all the empirical data, and it knows how to set itself up from an external file of empirical data in a certain format. Then if we register this distribution with the implementation with the external name "empirical_dist_A1" (for example), all that is required to get the implementation to use such a distribution is to give it a line of text looking like

```
empirical_dist_A1     extern_file.dat
```

where "extern_file.dat" is an external file of the empirical data that "empirical_dist_A1" knows how to read in order to set itself up.

Now it is easy to see how to set up *pathlinks* and *transporters*. They are defined by files that specify which distributions to associate with which values. For example, a particular *pathlink* (representing a cow) might be defined by a pathlink definition file containing:

```
Forage_consumption      lognormal   2.5   0.3   ! ln(kg/day)
Soil_consumption        Point       0.5         ! kg/day
Water_consumption       uniform     45    55    ! l/day
Forage_fraction         uniform     0.6   0.7   ! fraction of food from forage
Rel_bioavailability     triangular  0.5   0.65  0.8
Milk_fat_fractraction   tabular     milkfat.dat
```

where the initial entries on each line represent value fields of the *pathlink*, and the following entries the associated distributions. Evidently, substituting default values simply requires substituting the default distributions (which may be point values) in these definition files, or substituting a complete default file of definitions for the *pathlink* definition file (*i.e.* replacing the whole *pathlink* by a default object). Similar methods and comments apply for *transporters*.

It is just as easy to set up complete pathways for risk assessments. For example, the complex-looking pathways shown pictorially above may be set up by a pathway definition file containing:

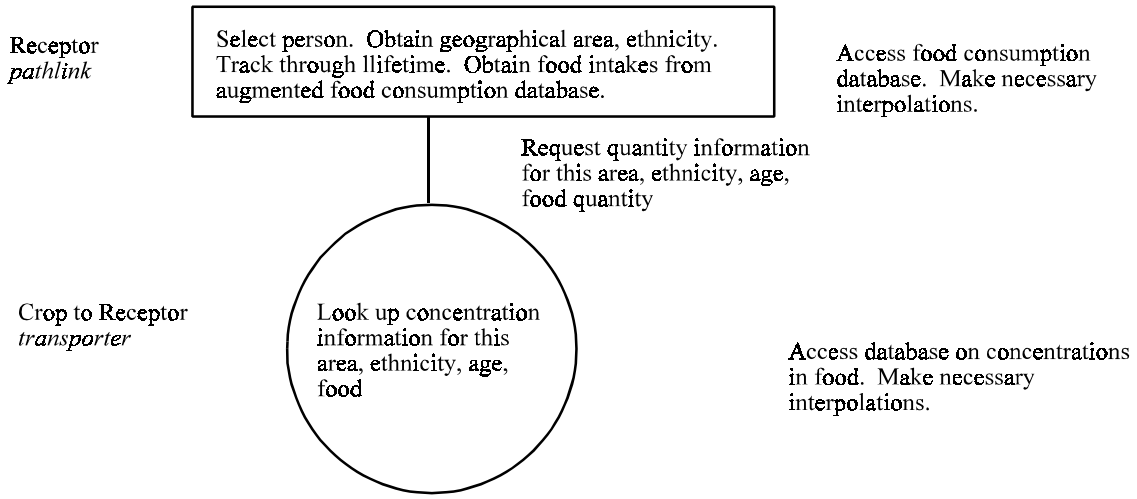
```
PATHLINKS          ; Keyword

; Name       Type         Definition              Comment

Receptor     Person       PersonA.rec             ; The receptor
Source       Manuf        Manufacturer.src        ; The manufacturers
CropA        Crop         CropA.lnk               ; Crop A
FieldA       Fields       FieldA.lnk              ; Fields assoc. w. crop A
Cows         Cow          Cows.lnk                ; Milk cows on fields
Gwater       Groundwat    Groundwater.lnk         ; Groundwater supplies
Swater       Surfacwat    Surfacewater.lnk        ; Surfacewater supplies
Backgrass    Grass        Backgrass.lnk           ; Backyard grass
Backgarden   Soil         Backgard.lnk            ; Backyard garden
House        House        Housing.lnk             ; Household


TRANSPORTERS       ; Keyword

;From        To           Type              Definition

Source       CropA        supply            cropA.trn
CropA        Receptor     vegsupply         vegsupply.trn
Source       FieldA       drift             cropAdrift.trn
CropA        FieldA       washoff           cropA.trn
FieldA       GWater       Soakin            Fieldsoak.trn
GWater       Receptor     Watersupply       Watersupply.trn
FieldA       SWater       Runoff            runoff.trn
SWater       Receptor     Watersupply       Watersupply.trn
SWater       Gwater       Soakin            Riversoak.trn
FieldA       Cows         Coweat            coweat.trn
Cows         Receptor     milksupply        milksupply.trn
Source       Backgrass    supply            bgrass.trn
Backgrass    Gwater       soakin            bsoak.trn
Backgrass    Swater       runoff            runoff.trn
Bgrass       Receptor     contact           grasscontact.trn
Source       Backgarden   supply            bgarden.trn
Backgarden   Gwater       soakin            bsoak.trn
Backgarden   Swater       runoff            runoff.trn
Backgarden   Receptor     Gardening         gardening.trn
Source       House        supply            house.trn
House        Receptor     indoor            indoor.trn
```

Here, the *pathlinks* are first named, they are then associated with types of *pathlinks*, and their definition files are given. Then the *transporter*s are identified by specifying which *pathlinks* they join, what type of *transporter* they are, and where they are defined (in definition files that look similar to *pathlink* definition files). Once again, at this higher level of abstraction, there is every opportunity to substitute default values for missing data. All that is required is selection of a default type of *pathlink* and/or *transporters*, with associated default values in its/their definition file(s).

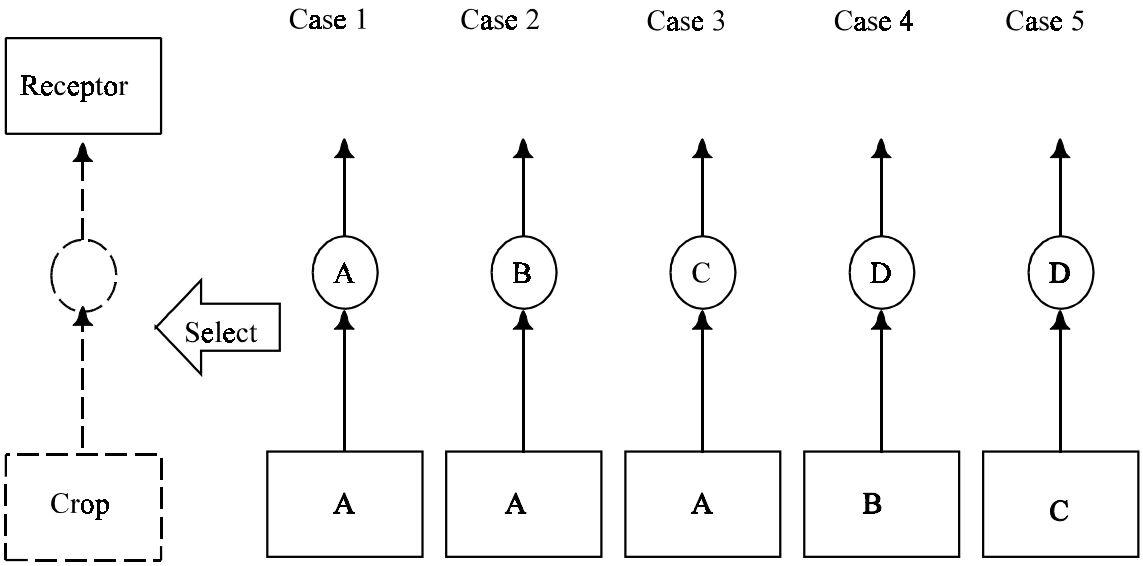
Thus this implementation has the "graceful degradation" property. At every level of abstraction, (individual parameter values, object and interaction definitions, and whole pathways), default values may be substituted if and when data are lacking. No change in the implementation is required beyond substituting the default values where required in the input files.

How to apply all this? Examine the food intake pathway under various conditions.

First, the "ideal" case. A huge amount of data is available on pesticide concentration in prepared food. Then the entire food pathway can be collapsed to just the receptor *pathlink* and the crop to receptor *transporter* (the crop *pathlink* becomes a placeholder only). The Monte Carlo procedure selects a receptor, and tracks that receptor through age. The food consumption of the receptor is followed throughout life by sampling from the database of food consumption (note that this requires some default assumptions, since the available databases do not track food consumption over age). The receptor requests the quantities of pesticides transferred in from the transporter (by informing it of the amount of food at each age). The transporter looks up the concentrations in the foods in its huge database, computes the quantities of pesticide, and returns that information. The Receptor accumulates doses, hazard indexes, lifetime risk contributions, or other toxicity indices, over the relevant time periods, and hands the results to the Monte Carlo control structures. The Monte Carlo control structures accumulate uncertainty and variability distributions by ethnicity, geographic area, and any other required characteristic, by repeating this procedure multiple times.



Now what happens with fewer data? We plug in alternate *transporters* and/or *pathlinks* that include more modeling and extrapolation from further back in the food supply chain to where there is data available.



Case	Crop to Receptor Transporter	Crop pathlink
1. Prepared food measurements	A. Sample out of measurement database or analytic distributions fitted to database measurements	A. Do nothing
2 Market basket measurements	B. Sample out of measurements or analytic fits. Apply models of food preparation.	A. Do nothing
3. Survey/composite market measurements	C. Sample out of measurements or analytic fits. Add uncertainty (and variability if known) to estimate single measurements. Apply models of food preparation.	A. Do nothing
4. Field measurements	D. Apply models of food distribution and processing. Apply models for food preparation.	B. Sample out of measurements or analytic fits. Apply extrapolation models to alternate areas.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

5. Tolerances only	D. Apply models of food distribution and processing. Apply models for food preparation.	C. Supply tolerances. Apply models for expected values going into food distribution. Apply extrapolation models to alternate areas.

Similarly for other pathways of exposure. The implementation allows one to choose the complexity of analysis methods independently in different pathways.

2 The Architecture

2.1 Some initial thoughts

We have to think about meeting certain risk management criteria for risk estimates from exposure to pesticides. So we immediately have to think about:

1. Exposures
2. Doses, dose rates
3. Toxicity
4. Risk management criteria that might be applied.

2.1.1 Toxicity

Toxicity is not currently under consideration here, except that we have to be aware of the measures of toxicity currently used to specify toxicity values, and how they may be changed later. There is clearly a need to consider total exposures, exposure rate, total doses, dose rates, time periods, and age. The current U.S. EPA approach is to select point estimates for toxicity values that are conservative (low) for dose rates or total doses over specified periods, possibly for persons of specified ages. The toxicity values are often very generalized (like RfDs, cancer potency slopes). Measures of risk (toxicity indices) are ratios of dose or dose rate to toxicity value (hazard indices, margins of exposure), or products of dose or dose rate and the toxicity value (risk estimates).

2.1.2 Risk management criteria

Toxicity indices are generally comparisons of an exposure/exposure rate/dose/dose-rate estimate with a toxicity value in some defined way (ratio or product). The resulting toxicity index for some defined population under some defined conditions (a scenario) must then meet certain risk management criteria to be considered acceptable.

Here are some possible simple examples of risk management criteria that might be applied:

Worst-case estimates of hazard index (using 1-day, 7-day, 1- month metrics of exposure and toxicity values) must be less than 1, or lifetime cancer risk estimates must be below some point in the range 1e–4 to 1e–7.

- The toxicity indices for reasonably maximally exposed individuals must be similarly limited (hazard index less than 1, risk estimate below the range 1e–4 to 1e–7).
- Some percentile of a variability and/or uncertainty distribution, or a combined specification involving percentiles of both distributions, must be evaluated and found to fall below acceptable limits.
- The frequency/magnitude variability distribution of risk estimates at a particular percentile of the uncertainty distribution must lie below a specified frequency/magnitude distribution.
- Toxicity indices in particular sub-populations (pregnant women, children, people with particular conditions/characteristics) must meet certain conditions.
- Logical combinations of the above criteria — for example, the 90th percentile of the variability distribution for lifetime risk at the 65th percentile of the uncertainty distribution must fall below 1e–6, and (simultaneously) the 99.99th percentile of the variability distribution for lifetime risk must be below 1e–3 at the 50th percentile of its uncertainty distribution.

We observe that all the criteria so far specified are expressible as relationships between functionals on distributions that maintain a separability between variability and uncertainty, and we hypothesize that all the criteria likely to be used will similarly be expressible as functionals on distributions that maintain such separability.

As a slight generalization, we will require the methodology to support analysis of distributions with an arbitrary number of distinguishable types of variation. We would expect just variability and uncertainty to be separated, but EPA might want to add extra distinguishing factors — for example, between the variability and uncertainty in human exposure parameters and those in environmental modeling parameters, as was done in HWIR. All criteria will thus be expressible as relationships between functionals of distributions. Practically, we will limit the functionals available — obvious examples are: extremes of a distribution with finite range, percentiles, mean, and generalized functional means (*i.e.* the mean of a function evaluated over the distribution).

Each of the criteria listed involves a defined population group (*e.g.* the whole population, children, asthmatics), and a risk measures (*e.g.* lifetime risk, hazard index, margin of exposure). Implicit within the risk measure is a time frame for its application that depends on its definition (*e.g.* hazard index computed for 1-day exposures, for 7-day exposures, for exposures lasting months or for a lifetime; or lifetime risks relative to some defined standard). Criteria may in principle also be calendar-time-dependent — that is, may depend on the calendar date, rather than on the age of any individual. For example, EPA might change a toxicity value, or EPA or Congress may change an acceptable risk level, or require that all assessments after some (future) calendar data should use a different acceptable risk level.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

2.1.3 Exposures

In an ideal world, what could one do to measure exposure? Suppose we took the whole population and measured everyone's exposures to pesticides. How would this be done? By measuring separately concentrations and intakes and any other characteristics (*e.g.* age, membership in sub-populations) required for matching with the toxicity values and criteria. Would such an approach be adequate for the FQPA? It would not! One cannot use that approach directly as a prediction tool, since one has only a snapshot at one time, and cannot evaluate pesticides not in use.

The minimum hypothesis required to use such an ideal approach is: current measurements are predictive. Then we could use distribution of results obtained by measurement as a prediction tool (for both current and unused pesticides). Measured exposures in the whole population give a distribution of exposures (and hence toxicity indices) in the population now, with full age-dependence and whatever else is needed to evaluate whatever criteria are selected. The hypothesis allows the extrapolation from the finite observed population distribution (but with uncertainties of measurement) to the hypothetical underlying distribution of which current measurements are a sample (with measurement error also).

The implication: we don't need to measure the whole population — a sample or set of samples will be just as good, provided such samples are correctly selected. Taking practical considerations into account, the sample may even be better than measurements in the whole population.

Is this minimum hypothesis plausible? It is not, since it does not take account of calendar time variation in the behaviors of the population. It might be possible to extrapolate from previous measurements using time as a predictor variable, but such an approach is inadequate. Time variations may be due to some physical effects (*e.g.* changing concentrations, dietary patterns) that may not be smooth in calendar time. The alternative is to use a causal hypothesis: calendar time dependence has a cause. To apply such a hypothesis, it is necessary to define the causal links — to examine the causal relations between concentrations in environmental media, exposures of humans to those media, and the resultant doses to humans.

This definition requires introduction of causal models, which (i) explain the concentrations found in environmental media, then (ii) explain exposure to environmental media by linking behavior with exposure, and finally (iii) explain exposure/dose relationships with models of humans. Such causal models may be used to link calendar-time dependence of measurements in populations with underlying time-dependent factors, such as use of pesticides, behavioral changes in various ages, and changes in humans. They also introduce the possibility of including other factors such as geographical area.

With the causal hypotheses come measurement and modeling requirements. It is necessary to measure or hypothesize the underlying time-dependent factors, such as behavior patterns,

geographical pesticide use patterns, change in use of bottled water, change in foods, and change in agriculture.

2.1.4 Dose and dose rate

Exposure and dose must be distinguished — there is an intervening physiological and/or behavioral step. However, many current toxicity values are expressed in terms of exposure, so that the implementation of the models has to be able to handle both exposure and dose. Moreover, the implementation must be flexible enough to allow the inclusion and/or subsequent addition of models for the interconversion between exposures and doses (*e.g.* pharmacokinetic models).

2.2Some architectural considerations

The methodology clearly has to::

Calculate with distributions
Account for calendar time
Account for age
Calculate exposures and/or doses
Handle geographical areas
Handle uncertainty and variability (or better, handle any finite set of distinguishable types of distributions)
Degrade gracefully — come up with the best available answer for the available information
Be extensible

It would be desirable if the methodology also:

Can tell you what extra information would be most valuable.
Can tell you the extent to which any particular piece of extra information would be of value.

2.3Architectural approach

The underlying idea is to track the exposures and doses of randomly chosen individuals through time. The random selection of the individuals is governed by the criteria that are to be imposed (*e.g.* particular age groups, particular ethnicities, particular geographical areas, particular habits), as is the length of time that individuals are tracked. Exposures, doses, and/or toxicity indices are computed for the individuals as they are tracked. By tracking enough individuals, the distributions required to evaluate the criteria may be generated.

The underlying approach may be specified by a pseudo-code (note: in the following pseudo-code, we use the abbreviation ARBC for As Required By Criteria)

```
Request criteria to be evaluated
Request (combination of) pesticide(s) to be evaluated
Enumerate limitations implied by criteria (ARBC)
Set up (by request or from data) use pattern for pesticide(s)
Set up environment(s) (ARBC)
Repeat
    Obtain initial estimates of, or re-estimate, tolerance levels
    Repeat (nested to the degree required by the distinguishable types of distribution implied by
                the criteria)
        Select an individual meeting specified limitations (ARBC)
        Set initial calendar time and age (ARBC)
        Repeat
                Update individual (age, location, other parameters)
                For selected individual, obtain activity
                Compute exposure(s) associated with the activity
                If required, calculate toxicity measure and toxicity index
                Cumulate exposure(s) or toxicity index (ARBC)
                Step time to end of activity
        until ending time/age ARBC
        ARBC, obtain relevant toxicity measure
        ARBC, compute a relevant toxicity index from cumulative exposure
    until enough samples obtained to evaluate criteria to required accuracy.
    Evaluate relevant criteria
until criteria are met or no re-estimation of tolerance levels is required
Display result(s)
```

The individual pieces of this pseudo-code are explained in more detail below.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

2.3.1 Request criteria to be evaluated

The likely required criteria have not been specified in full detail. We can expect the criteria to introduce the listed limitations, although other such limitations or specifications are possible.

Age range of interest
: Probably nursing infant, infant, child, teen, adult, whole life or something similar. Assume that an age range is available.

Calendar times of interest
: Expected to be now and for any foreseeable time. Assume that an initial time is given, and have a default calendar period over which any extrapolations included can be considered reliable.

Locations of interest
: Anywhere in the US. Inner City, Urban, Suburban, Rural. Particular geographical areas of the US. Combinations.

Measures of interest
: Minimum margin of exposure, hazard index, cancer risk. Worst case, a percentile of a distribution, an uncertainty percentile of a variability percentile, an uncertainty percentile on a mean variability, an uncertainty mean on a variability percentile. Allow for an arbitrary percentile or defined function on each of a finite set of distinct distribution types.

Time spans of interest
: The measures of interest should incorporate a time-span for the exposures of interest. Typically these will be 1-day exposures, 7-day exposures, 1-year exposures, and lifetime exposures, or a similar set. Typically, what will be of interest are the worst cases of these exposures over the age-ranges of interest. More complicated specifications are also possible.

Specification of specific sub-groups
: Particular ethnic group, subsistence farmer, subsistence fisher, home gardener, deep-sea-fisher, freshwater fisher.

2.3.2 Request (combination of) pesticide(s) to be evaluated

Expect to be able to select from a database of current pesticides, in order to know their properties and to gain access to any databases of measurements on such pesticides. Alternatively, or in addition, provide the characteristics (physical, chemical, toxicological, environmental properties) of any new pesticide, and provide access to any available databases of measurements.

2.3.3 Set up environment(s) (ARBC)

Perform any initial calculations necessary to set up the program to allow it to obtain values for environmental exposure concentrations for the pesticide(s) of interest, under the conditions implied by the criteria that will be used.

As the simplest example, set up links to databases of measurements, so that subsequently the program can sample the measured concentrations in the environments relevant to the individuals who will be tracked through time. Examples include measurements in indoor air, in water supplies, on indoor surfaces after pesticide applications, and in as-bought food items.

In more complicated cases, and in many cases for new pesticides, this step will entail running (or setting up) models for the transport of the pesticide through the environment under the envisioned conditions of use. Thus, if residue measurements are available at the farm gate, it may be necessary to track these concentrations through the food distribution chain to estimate the concentrations in foods as bought. Such models can obviously be of varying complexity. They may take account of expected patterns of use on various crops, geographical distributions of crops, yields of crops, the distribution network, and fractions of crops used locally and distributed more widely. In addition, the modeling may take account of modifications to concentrations of the pesticide (and possible production of decay or breakdown products) during distribution, and of transfer of the pesticide through other foods — *e.g.* uptake into milk and meat, and the subsequent distribution of these foods.

The implementation must be flexible enough to incorporate any combination of sets of measurements and modeling approaches, in order to allow probabilistic sampling of the concentrations in the environmental (including available foods and drinks) around the selected individuals.

2.3.4 Obtain initial estimates of, or re-estimate, tolerance levels

The object of the program is to obtain estimates of tolerance levels that will be acceptable according to the specified criteria. The repeat loop involving re-calculation may be necessary in order to perform this computation — we have to make initial estimates of tolerance levels and then recursively modify those estimates to take account of calculations with the previous estimates. The method of modification will depend on what is being calculated and also (if multiple tolerances are being simultaneously evaluated) on the approach taken to parceling out the available leeway to the individual tolerances.

If a single tolerance level for a single pesticide is under evaluation, then there is a unique way of selecting the tolerance level to meet a given set of criteria. In such a case, the step simply

consists of performing a calculation for an initial estimate, then converging on that unique result. If all the relevant exposure or dose estimates are linear in the tolerance level, it is necessary only to perform a single calculation of the risk estimates. If the relationship between tolerance levels and exposure or dose estimates is non-linear, a recursive approach will be required (hence the outermost repeat loop).

For multiple tolerances, multiple pesticides, or both, some method of parceling out the available risk or of trading off among tolerances must be specified in order to make estimates of tolerance levels that will just satisfy the given criteria. Without such a method, there are likely to be multiple sets of tolerances that will meet the risk criteria. The simplest case to illustrate the problem is for tolerances of the same pesticide on two crops. One set of two tolerances that would meet a given risk criterion would be for one tolerance to be zero, with the second at some finite value. A second set would be with the first tolerance at a finite value, and the second at zero. Then there is a curve of one tolerance versus the other tolerance connecting these two combinations, and all points below and on the curve would produce exposures that satisfy the risk criteria. The curve is a straight line if the risk estimates are linear in the tolerance levels. Selection of a single pair of values requires an extra specification of the trade-off between the two tolerances that selects a particular point on that line.

With more tolerances, and/or multiple pesticides, the same problem occurs, and again some method of trade-off must be specified in order to obtain unique solutions for tolerances. With such a trade-off method specified, this step consists of taking the results of the risk estimates obtained with the previous estimates for tolerance levels, and estimating an improved set of tolerance levels that both satisfy the risk criteria and also satisfy the trade-off requirements. If the risk estimates are linear functions of tolerance levels, this step may be carried out by a matrix inversion without the repeat loop over risk estimation, although satisfying the trade-off requirements may require some iteration internal to the step. Otherwise, repeated calculation of the risk estimates will be necessary using the improved tolerance level estimates.

2.3.5 Nested repeat loop over types of variation

The next level of repetition is over selection of individuals to track. This has to be performed enough times to generate the required distributions to the desired degree of accuracy. The simplest way is as sketched here — over each distribution to be evaluated, draw enough samples to generate the distribution to the accuracy required. Once the criteria are known, it is fairly straightforward to formulate stopping rules to determine when the distribution is known to the required accuracy (an accuracy that has to be specified in the implementation).

Various techniques may speed up these repetitions. For example, it may be possible to prove in some circumstances that certain of the repeat loops are merely inefficiently calculating averages, in which circumstances those averages may be pre-calculated.

2.3.6 Select an individual meeting specified limitations (ARBC)

The criteria may specify various pieces of information about the population — for example, the criteria may be limited to children, or to people in a particular census region, or to fish eaters only. This step randomly selects from the population individuals who meet the limitations specified in the criteria. The selection criteria may include the calendar time (in case the population has changed over time) and the initial age of the individual.

2.3.7 Calculations for the individual — the innermost repeat loop

This section covers

Repeat
    Update individual (age, location, other parameters)
    For selected individual, obtain activity
    Compute exposure(s) associated with the activity
    If required, calculate toxicity measure and toxicity index
    Cumulate exposure(s) or toxicity index (ARBC)
    Step time to end of activity
until ending time/age ARBC

in the pseudo-code.

The innermost repeat loop in the program requires calculating risk estimates for the selected individual. This first requires estimating exposures or doses by tracking the individual through time (and hence also through age). Each activity of the individual involves contact rates with the environment, and hence with environmental concentrations of the pesticide(s) under study (these have previously been evaluated under the required conditions, see Section 2.3.3). Here one has to use various interaction models to estimate exposures and doses from the combination of activity and environmental concentrations. Such models may be relatively simple for food and water intake (eating and drinking), but may get substantially more complex for other contacts (for example, estimation of the quantity of pesticide dermally absorbed during playing on a floor). Even for eating and drinking, more complex models might be used if they are available and if the criteria allow — pharmacokinetic models could be used for instance. For each such model, however, there are some fairly simple (conservative) default values that may be adopted.

With the exposure or dose estimate obtained for each activity, the measures relevant to the criteria under study may be cumulated — for example, lifetime dose estimates, maximum dose rates (averaged over 1 day, the preceding 7 days, the preceding month, and so on), minimum margins of exposure (similarly averaged), or perhaps body burden, blood concentration, or a more complex measure, if one is using a pharmacokinetic model

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

The innermost repeat loop consists of making these estimates for each activity, and stepping through time from activity to activity, for as long as required by the relevant risk management criteria. The object of stepping through time is to take account of exposures that are occur within short intervals (for example, multiple simultaneous or near-simultaneous activities all involving pesticide exposure), while also incorporating longer term variations (such as seasonal effects). In some cases it may be possible to short-circuit the process of examining each activity separately in time. For example, if the criteria call for examination of mean values over longer periods than covered by each activity, as would occur if very short-term variations in dose or exposure rate were unimportant, the mean values may be pre-computed.

2.3.8 Computing risk estimates

This section covers the entries:

ARBC, obtain relevant toxicity measure
ARBC, compute a relevant toxicity index from cumulative exposure
until enough samples obtained to evaluate criteria to required accuracy.
Compute relevant criteria
until criteria are met or no re-estimation of tolerance levels is required
Display result(s)

in the pseudo code.

For each individual, the cumulated exposure, dose, or toxicity index has to be converted to the toxicity measure (*e.g.* a hazard index, lifetime risk estimate, margin of exposure) appropriate to the risk management criteria. This toxicity measure must then be converted to the relevant toxicity index (*e.g.* the ratio of hazard index to unity; the ratio of lifetime risk to a fixed value of 1e–6; the minimum margin of exposure) for comparison with the risk management criteria. Finally, the toxicity index may be compared with the risk management criteria to determine the adequacy of the current tolerance level estimate(s).

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

3 An Implementation Approach

In this chapter, we describe one possible approach that can be used to implement the architecture outlined in Chapter 2, using object-oriented methods that match very closely with the requirements of the architecture, and with the real world. Much of this chapter will appear rather abstract, with little initial connection with the problem at hand. Unfortunately, this seems inevitable — the data structures that are used are themselves abstract entities that can be used in a wide variety of situations. We deliberately build an implementation that uses elements that are as abstract and general as possible, but make it simple to specialize to our requirements. This results in an implementation that is much easier to modify to take account of changes (in data, methods, and policies) than a more specialized structure. It also turns out to be easier to construct the highly abstract implementation than a more specialized one, and the match to the architecture is better.

In addition to the abstractness, much of the chapter is also written in rather technical language, in order to be concise and precise. The technical language used is that of object-oriented and other computer programming, and following it requires a working knowledge of these disciplines. Those not familiar with classes, objects, instantiations, linked lists, pointers, and other such esoterica, should consider just reading Section 3.1 and skipping the remainder of the chapter. For the cognoscenti, specific terms in what follows are as defined in Borland Pascal and Delphi. However, similar constructions appear, or can be constructed, in all object-oriented languages, and translation between them is straightforward.

3.1 Elementary introduction to the techniques

The architecture described requires two fundamentals properties for setting tolerances under the FQPA:

1 complete accounting for variability and uncertainty, and
2. accounting for multiple ways of being exposed.

The first requires the handling of distributions of values, rather than point estimates, while the second requires more flexibility than current approaches (deterministic or probabilistic) allow. We propose the use of a programming technique, object-oriented programming, that simplifies both requirements, in that distributions are handled automatically and arbitrary pathways of exposure can be readily constructed and incorporated.

In our approach, all values entering calculations are chosen from probability distributions. Using object-oriented programming, we have made this as simple to implement as the use of ordinary scalar values. All probability distributions are defined to be descendants of a fundamental *distribution* object type (or class), and all risk assessment code handles this fundamental *distribution* type. Thus any distribution (even those not yet defined) can be handled by any part of the code without further intervention by the programmer. As currently implemented, all distributions may be initialized from data files by using keywords to define what type of distribution is required, together with the parameters to define that particular distribution. New distributions can be added at any time, and the system incorporates a registration procedure that automatically adds the keywords associated with such new distributions. Currently, the definitions of all distributions also include a "preferred value" that may be used with a simple switch to evaluate the standard point estimates commonly generated in risk assessments.

Risk assessments for pesticides involve physical objects such as sources of the pesticide, pathways of exposure, the receptors, and sets of exposure conditions. The object-oriented approach associates with each such physical object an instantiation of some computer object type (class) with parameters appropriate to the particular object represented. Pathways of exposure are represented by a sequence of such *pathlinks*, each representing part of the physical world (*e.g.* the pesticide source, the air, human foods, plants in a field, the field itself, the watershed containing the field, the river draining the watershed, cows, a human receptor). Each *pathlink* is represented by an instantiation of a particular *pathlink* type from a limited set. For example, different human foodstuffs may all be represented by a *food* pathlink type, but with different values for the parameters for each foodstuff (if such different parameters are required). In the simplest case, for example, *milk, bread,* and *lettuce* might be three different instantiations of the *food* pathlink, but each would have different distributions for pesticide concentrations. Insofar as there are processes common to all *food* pathlinks, they may be defined for *food* and automatically apply for all the instantiations of different foods.

As a more complex example, a source of soil contamination, a field, and a watershed, may all be represented by a *soil pathlink* type, with different values for the parameters in each case. The soil type, area, and erosion characteristics (among others) would be different for the three different instantiations of the *soil pathlink* representing these physical objects, but all would have a common set of parameters and behave in the same way — since the physical, chemical, and other environmental processes are similar in all cases. The physical, chemical, and environmental behaviors of the *pathlinks* are defined by subroutines (methods) that are associated with that *pathlink* type. These methods correspond to the usual types of models used in risk assessments to relate contaminant inputs to concentrations or other required output information — examples are mixing and erosion models for soils, bio-transfer factors for cows and cattle, and risk estimates for human receptors.

This technique also easily allows for exceptions or improvements. A generic *human* pathlink, for example, may have an associated method that keeps track of the body-burden of a pesticide, by accumulating inputs of the pesticide and utilizing a simple decay constant. For some pesticides, much better pharmacokinetic models might be available. Then all that need be done is to define

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

a descendant type *P_human* from the *human* that overrides the body-burden method with the better pharmacokinetic model (but retains all the other properties and methods of the *human* type); the *P_human* descendant type may then be used wherever a *human* type is called for, and no change in any risk-assessment programming is necessary to incorporate the change — the object-oriented approach ensures that wherever an instantiation of a *human* type was required, a *P_human* type is also compatible. The selection of which descendant type to use in any particular circumstance has been made flexible also — it does not have to be made in advance by the programmer, but may be made at run time by the user of the program.

The *pathlinks* themselves represent physical objects (such as cows, humans, dogs, backyard gardens), but not the interactions between physical objects. Such interactions are handled in the code by *transporter* types that allow pair-wise interactions between *pathlinks*. In the code, instantiations of *transporter* objects handle such interactions by their own methods, with different *transporter* types for each type of interaction.

A complete exposure characterization using this methodology requires specification of the source and receptor *pathlinks*, intermediate *pathlinks*, and all the *transporters* that mediate interactions between the *pathlinks*. We have devised an approach that allows easy specification of all these, using keyword-parameter value lists in data files, where each parameter may itself be (recursively) a keyword-parameter pair. The approach is set up to allow arbitrary numbers and types of sources, other *pathlinks*, *transporters*, and receptors, so that multi-pathway assessments can be handled consistently and easily. As for distributions, new types of *pathlinks* and *transporters* may easily be added, and again a registration procedure is in place to add the associated new keywords.

The keyword-parameter lists used to define *pathlinks* and *transporters* allow specification of the parameters as distributions, so that the parameter is another keyword-parameter pair that describes the distribution. With this approach, each parameter may be associated with a value from a probability distribution. All the objects in the code are automatically set up so that they contain a method that updates parameter values with new values from the associated distributions. The complete exposure characterization described above is set up so that a single method call results in all the parameters in all the *pathlinks* and *transporters* being updated automatically, allowing for easy implementation of probabilistic (Monte Carlo) calculations. It is a straightforward extension to associate each distribution with one of any finite number of distinct flavors (*e.g.* uncertainty and variability; or a more extensive list that separates out uncertainties and variabilities in different parts of the modeling), and to update just one flavor (or any combined set of flavors) at a time.

The approach adopted is actually even more flexible, in that the keyword-parameter definition mechanism allows associating particular parameters with function calls rather than distributions. This allows (for example) the use of default estimation procedures if no value is directly available (*e.g.* to estimate soil-water partition coefficients from octanol-water partition coefficients and soil carbon mass fractions). The object-oriented approach ensures that this flexibility is available for all input values.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

A further advantage of the object-oriented approach is in the treatment of time- and age-dependence. We have devised a mechanism to represent time-dependent and age-dependent parameters in a uniform way, through object types that can handle age- and time-dependence. Thus all parameters may actually be specified in a time- and age-dependent way, in addition to being distributions.

The risk-assessment approach that we propose to build upon is thus highly flexible, extensible, and can be made very easy to use. In its current form, all input is currently in the form of data files (ASCII text files) that are almost self-documenting, because of the freedom to use English text keywords and arbitrary comments in these files. The input methods have been designed so that they may be readily modified so that all input is to a graphics-based screen window, with all parameters entered on-screen or selected from a pick-list of default values (*e.g.* a list of available distributions, available correlations).

3.2Implementation details — high level structures

This and the following sections become more technical, and require that the reader have some familiarity with the concepts of modern object-oriented programming techniques and languages. The object of the approach is to allow relatively easy construction of multi-pathway exposure assessments similar to that sketched in Figure 3.1, with the flexibility to easily change the implementation details and even the high level structure of the pathways. The following description of an implementation technique allowing this flexibility is initially structured in a top-down fashion — the high-level structures are described first. Subsequent descriptions will build upward to illustrate how to construct such structures.

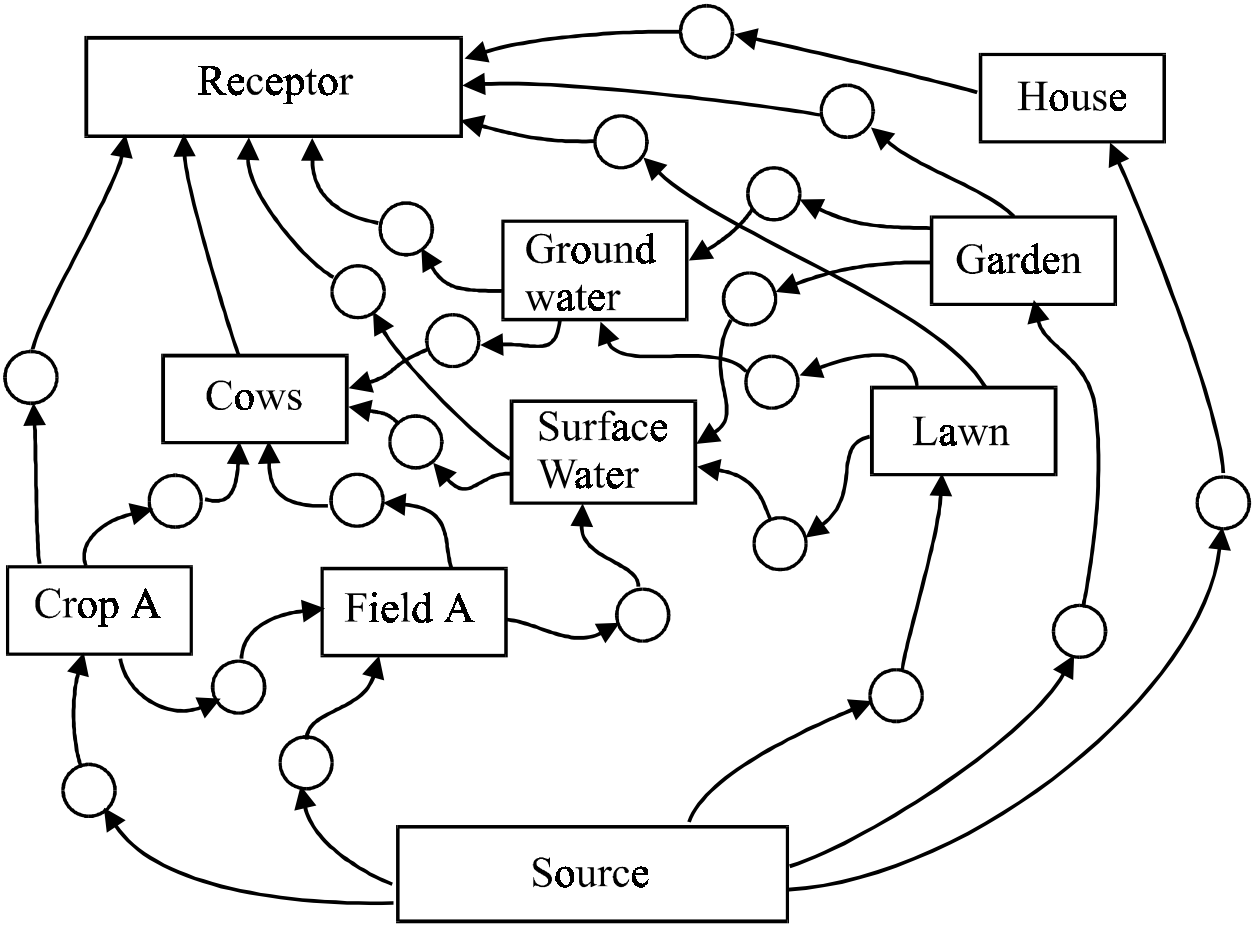


Figure 3.1 A diagrammatic version of a set of pathways

In Figure 3.1 the boxes represent physical entities and the circles represent interactions between those physical entities. For example, the “Crop A” box represents one particular crop to which the pesticide might be applied, and the circle between it and the “Source” box represents the methods of application of the pesticide to the crop.

The fundamental approach is to make the computer representation as similar as possible to such a representation of the physical world, and to be able to easily construct complex pathways, while still incorporating Monte Carlo analysis making use of all available data. The computer representation of CropA will contain information on the type of crop, the extent of plantings, its geographic area, and any other necessary details about this particular crop as required by any mathematical models models describing and associated with this particular crop. The intent of a diagram like this is to separate the physical world into discrete and reasonably independent parts, so that each may be modeled independently.

Figure 3.1 may be constructed by a sequence of instructions that first define the boxes involved, here called *pathlinks*, followed by a sequence of instructions that define the interactions, here

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

called *transporters*, between the *pathlinks*. A computer representation of Figure 3.1 may be constructed by the following sequence of instructions in an ASCII file:

```
PATHLINKS          ; Keyword

; Name       Type         Definition              Comment

Receptor     Person       PersonA.rec             ; The receptor
Source       Manuf        Manufacturer.src        ; The manufacturers
CropA        Crop         CropA.lnk               ; Crop A
FieldA       Fields       FieldA.lnk              ; Fields assoc. w. crop A
Cows         Cow          Cows.lnk                ; Milk cows on fields
Gwater       Groundwat    Groundwater.lnk         ; Groundwater supplies
Swater       Surfacwat    Surfacewater.lnk        ; Surfacewater supplies
Backgrass    Grass        Backgrass.lnk           ; Backyard grass
Backgarden   Soil         Backgard.lnk            ; Backyard garden
House        House        Housing.lnk             ; Household


TRANSPORTERS       ; Keyword

;From        To           Type              Definition

Source       CropA        supply            cropA.trn
CropA        Receptor     vegsupply         vegsupply.trn
Source       FieldA       drift             cropAdrift.trn
CropA        FieldA       washoff           cropA.trn
FieldA       GWater       Soakin            Fieldsoak.trn
GWater       Receptor     Watersupply       Watersupply.trn
FieldA       SWater       Runoff            runoff.trn
SWater       Receptor     Watersupply       Watersupply.trn
SWater       Gwater       Soakin            Riversoak.trn
FieldA       Cows         Coweat            coweat.trn
Cows         Receptor     milksupply        milksupply.trn
Source       Backgrass    supply            bgrass.trn
Backgrass    Gwater       soakin            bsoak.trn
Backgrass    Swater       runoff            runoff.trn
Bgrass       Receptor     contact           grasscontact.trn
Source       Backgarden   supply            bgarden.trn
Backgarden   Gwater       soakin            bsoak.trn
Backgarden   Swater       runoff            runoff.trn
Backgarden   Receptor     Gardening         gardening.trn
Source       House        supply            house.trn
House        Receptor     indoor            indoor.trn
```

This is an example of a control file that might be used to construct the pathways in Figure 3.1 (although it is likely that there will be more elements in the control file, in addition to those given here). A control program can read this file and construct a computer representation of Figure 3.1 – and changes to the pathways illustrated in Figure 3.1 (to incorporate new pathways, or to extend one of those shown, for example) may be readily incorporated simply by changing the control file.

The `PATHLINKS` keyword is an instruction that there follows a list of *pathlinks*. Each *pathlink* is given a name (unique to this control file) as the first element on each line. The following

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

token on the line is a *pathlink* type — internally, each distinct name in this column has been registered, and corresponds to a particular class type descended from a common abstract *pathlink* ancestor. The internal registration of names allows run-time polymorphic class instantiation – the control file is read, and an instantiation of a *pathlink* object of class corresponding to the name is instantiated (constructed) for each entry. The last entry on each line in the PATHLINKS section is a file name that contains the data required for this particular instantiation of the particular *pathlink* class (and every *pathlink* knows how to construct itself from such a file). Thus there may be multiple instantiations of the same *pathlink* type, using different data in each case — or using the same data if that is appropriate (there is no such example here).

Each *pathlink* (computer) object represents a physical object (or set of objects — for example, "FieldA" here might represent all the agricultural fields to which the pesticide is or has been applied, together with all adjacent fields affected by drift). It contains fields that parametrize the physical object, and methods that represent how the physical object behaves. In particular, it contains a GetConcentration method overriding the ancestor abstract *pathlink's* similarly named method, which calculates the concentrations in the physical object(s) that will come from this particular *pathlink* (*e.g*. residues for CropA). For the *soil* type, for example, the fields are such information as soil density, porosity, organic carbon content. The methods represent models that allow (for example) the calculation of the concentration of a chemical as a function of time and depth, given information on the inputs of that chemical to the surface of the soil as a function of time, and the GetConcentration method returns this information. For CropA (a *crop* type), the method may be as simple as reading from a table of measured residue values, or may be a complex model that relates residue levels to period of growth, period since application, application rate, and so forth. At this level of abstraction, the details of such methods are not required to be known. Addition of new types of *pathlinks* to the modeling is straightforward, by defining new *pathlink* class objects descended from the ancestor *pathlink* class, or from one of the available *pathlink* classes. Once a new *pathlink* has been defined and registered, it may be used in exactly the same way as previously defined *pathlinks*. Since all general uses of *pathlinks* (like adding them into the model structure defined by a diagram like Figure 3.1) involve operations that use only the methods defined in the abstract ancestor *pathlink*, and hence available to all descendants, new *pathlinks* may be defined and added into the modeling with no changes to the rest of the code (this is one of the advantages of object-oriented programming).

For convenience of access, all *pathlinks* should be strung together in a linked list as they are defined, using a link field that is defined in the abstract ancestor *pathlink*. It is then straightforward to search for source and receptor *pathlinks*, and return those as results of the construction code. One needs a handle on the receptor *pathlink*(s), since the whole risk assessment process is performed simply by invoking the CalculateRisks method of that receptor.

The TRANSPORTERS keyword indicates the end of the PATHLINKS and the beginning of a listing of all the *transporters* in the pathway. Each *transporter* is a logically-one-way link between two *pathlinks*. A new instantiation of a *transporter* is created by each line of the control file, with the "from" and "to" *pathlinks* listed first on the line (using the names defined above under the PATHLINKS keyword — and of course, one performs run-time checks for the names

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

having been defined there). The type of *transporter* is defined by the name listed third in line. Once again, an internal registry of *transporter* classes links this external name with a particular *transporter* class type, and all transporter class types are descendants of an abstract ancestor *transporter* class type. The registration mechanism allows polymorphic run-time instantiation of *transporters*. As for *pathlinks*, new *transporters* may be defined whenever necessary, and all previously operating code will operate on the new *transporters*.

Each *transporter* class contains fields that represent parameters and methods that represent models. The models are representations of the physical processes or market mechanisms that move chemical pollutants from one physical object to another or move produce through markets. There is in principle no limit to the complexity of the models that may be employed here. Each *transporter* requests the concentrations of the pesticide in its "from" *pathlink*, and computes the flow of pesticide from the "from" *pathlink* to the "to" *pathlink*. The result of this computation is available as the result of its *GetInput* method (which is originally defined in the abstract ancestor *transporter* type). Although the *transporter* is logically-one-way, the flow of pesticide could theoretically be in either direction.

Each *transporter* is linked to its "from" and "to" *pathlinks* during model construction (as the control file is read). The linking mechanism is straightforward — each *transporter* is incorporated into a linked list originating in the "to" *pathlink* (the anchor of the linked list is a field in the *pathlink*), and one of its fields is set equal to the *self* parameter of the "from" *pathlink* (*i.e*. one of the fields of the *transporter* is a *pointer* to the "from" pathlink). As each *transporter* is linked into the model, it checks that it is being linked between the correct two *pathlinks* — that it knows how to handle pesticide flow between these particular *pathlinks.* All aspects of the two pathlinks that it is connecting are available to the *transporter*, and it can easily obtain the concentration in the "from" *pathlink* by invoking that *pathlink*'s GetConcentration method. The GetConcentration method of the "to" *pathlink* , when it is invoked, runs through the linked list of incoming *transporters* and invokes each of their GetInput methods in order to be able to calculate its own concentration (the "to" *pathlink* makes all aspects of itself available to the *transporter* by passing a pointer to itself in this call).

The Person *pathlink*, representing a human being, has an additional method CalculateRisks that performs similarly to the GetConcentration method. Instead of calculating a concentration, however, it runs over the linked list of incoming *transporters* and uses the resulting estimates of pesticide inputs to estimate risks. Simply invoking a Person CalculateRisks method thus automatically performs a complete risk estimate, since the whole pathway is calculated by that single call (the calls to the GetInput methods of *transporters* to the Person in turn invoke the GetConcentration methods of the *pathlinks* directly connected to the Person; in turn, they invoke the GetInput methods of the next removed *transporters*, and so on back to the source(s)).

There is no limit on the numbers of pathways, sources, etc. connected ultimately to the same receptor (linked lists can be as long as one wants). One could strictly perform simultaneous assessments in parallel by incorporating multiple receptors.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

There is no fundamental limitation to the level of detail that may be modeled using this approach. Figure 3.1 represents one breakdown that probably corresponds fairly closely to the level of detail that is possible at present. However, if further detail were to become available, it would be straightforward to change the structure of the pathways — for example, to replace a single *pathlink* (box) in Figure 3.1 with a combination of *pathlinks* and *transporters* that model pesticide flows in greater details. Alternatively, if more complex models become available using the same generic structure, they may be incorporated by defining new *pathlink* and *transporter* types incorporating those models in their methods. All that is necessary is to register these new *pathlinks* and *transporters*, and substitute their registered names in the control file, in order to update the complete assessment to use them.

The basic idea is that if the risk assessment can be represented by a diagram like Figure 3.1, then the implementation can handle it without any modification. New types of boxes and circles will themselves have to be programmed, but the overall implementation is unchanged — the new boxes and circles are simply added into a list of available boxes and circles for any assessment.

There are two limitations to this implementation method, however, both of which might be overcome by more complex programming:

(a) There must be no loops. Logically, there should be no need for loops, since each *transporter* should be able to account for both directions of pesticide transport. More complex programming would be able to detect loops, and it might be possible to allow for them.

(b) The *transporters* must be able to estimate the transport rates of the pesticide using just the concentrations in the "from" *pathlinks* — so that the models cannot handle interactions between different pathways. This can be awkward if equilibrium models are being used for *transporters*, for example, since the transport rates will depend on the total input to the "to" *pathlink*, not just the input through a particular transporter. This problem seems to be rare, and can often be fixed up by a slight re-definition of what is included in the *pathlink* and what in the *transporter*. In principal, and with some more complex programming, one could define a parallel matrix representation of the complete set of pathways, with the *pathlinks* and *transporters* being used to evaluate the matrix elements from locally linear approximations to all the models, and then invert the matrix. This would have to be iterated if any of the models were non-linear in concentrations. This problem appears to affect all current attempts at multi-pathway analyses.

One can add various implementation details to the above structure. For example, examination shows that the approach taken may be inefficient, since the GetConcentration methods of a particular *pathlink*, or the GetInput method of a particular *transporter*, may be called multiple times with identical arguments. Since these may represent complex and time-consuming model calculations, it may be desirable to cache the results of such computations.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

3.3Implementation details — low-level structures

Section 3.2 defines the general high level structure of the approach. A large part of the interest comes about in the methods of implementing *pathlinks* and *transporters* so that it is easy to construct them, so that Monte Carlo analysis is trivial, and to take account of the vagaries of physical modeling. To do that, it now seems more natural to switch to a bottom-up descriptive approach. In order to avoid momentarily extraneous detail, the following descriptions are in many cases incomplete, but extra detail may be subsequently added.

3.3.1 Distributions

At some point, everything is a distribution. All the parameters going into the *pathlinks* and *transporters* are described by distributions, and indeed, every parameter required in any risk assessment may be described by a distribution. This is taken as fundamental — everything is designed so that every value entering the risk assessment is described by a distribution.

The ancestor `Distribution` class has an essential method, the `Random` method, that returns a random number from the distribution. It also has a more useful method, `Ran`, that returns a random value, a fixed value, or some well-defined other value, depending on a state value `Returnvalue`, associated with the distribution. The "fixed" value is designed to accommodate EPA "preferred" values for parameters. The other well-defined values are incorporated to allow the use of EPA's procedure of assigning central tendency and extreme values, but is more general.

From this ancestor, various useful descendants may be defined. Typically, one defines Point, Uniform, Normal, Lognormal, Exponential, Beta, Gamma, and other classical distributions, together with distributions defined by Spline fits (interpolations) of data, and simply tables of data. Each descendant class is registered in a Registry class with a registered name. The Registry class has a method that creates an instantiation of a particular *Distribution* class descendant when presented with the registered name for that descendant. Thus it is straightforward to allow polymorphic construction of distributions from external data files. The external file simply has to contain information like

```
lognormal    3.5    1.5    57.35
Table        thistable.dat
```

and presenting this to the Registry results in the instantiation of a lognormal distribution with geometric mean exp(3.5), standard deviation of logarithm 1.5, and "fixed" value 57.35; and a distribution defined by the tabular data in the file "thistable.dat".

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

With this approach, there is no limitation on what types of distribution may be used. All applications can use the generic abstract `Distribution` class, and then automatically handle all descendants. Moreover, the user generally is not aware (nor does the user have to be aware) of the particular distributions that may be subsequently used. Changing distribution types simply requires a change in the data initialization files, not in the programming.

Handling correlated multiple random variates require some extensions that are handled by the `MultiDistribution` class. The `MultiDistribution` class is descended from the `Distribution` class, but its `Random` and `Ran` methods need to return multiple values. This is handled by providing to the `MultiDistribution` (either at initialization, or at some other convenient time), a vector of the addresses where those variates have to be returned. As will be seen in Section 3.3.2, such an approach is precisely what is required in applications. Descendants of the `MultiDistribution` class may be initialized by providing an input line of the form

```
Multiple  filename     elementname
```

where `Multiple` is a keywording indicating a descendant of `MultiDistribution` class, `filename` is the name of an external file containing details required to initialize this distribution, and `elementname` is a name of one of the variates described in the file `filename`. The external file `filename` then takes the form

```
Multipledistributionname     numberofvariates
variate1 variate2 variate3 .......
...................(data)...................
```

where `Mulitipledistributionname` defines the particular type of distribution; `numberofvariates` gives the number of variates produced by this particular distribution; `variate1, variate2`, etc. provide convenient names for referencing each of the variates produced; and further lines in the file provide the data required to initialize this particular distribution (and each `MultiDistribution` class knows how to read its own initialization file). An example might be:

```
multinormal    4
v1   v2   v3   v4
7.0    2.5   8.3   10.4
1.5
0.5    5.0
-0.5   0.7   3.5
0.3    1.0   0.5    7.0
```

providing the initialization for a multinormal distribution with four elements, with names v1, v2, v3, v4, with means (7.0, 2.5, 8.3, 10.4) and variance-covariance matrix given by the following four lines (only the lower diagonal elements are needed, since the matrix is symmetric).

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

The registration method that associates external names with internal distribution types caches the file name, so that further references to the same filename will automatically link to the same instantiation of the distribution. The instantiation procedure for the distribution itself caches these particular instances of external names for the variates, to allow subsequent references to these names in external control files (and to allow name cross-checking of the input date). Thus the same high-level initialization files (see Section 3.3.2, below) may refer to multiple variates from the same file, thus:

```
Multiple  mfile1.dat     v3
Multiple  mfile1.dat     v2
Multiple  mfile1.dat     v1
Multiple  mfile1.dat     v4
```

with the assurance that just one instantiation will be produced, giving the correct correlation structure between the variates `v1`, `v2`, `v3`, `v4`.

3.3.2 Application of distributions — the `Distfunc` *class*

Monte Carlo risk assessment requires selection of random values from distributions, followed by computations using those random numbers. The particular values are often used multiple times within each calculation, so some storage method is required — one cannot each time call for a new random value. An approach is required that allows easy connection of a computer variable with a distribution, so that the variable can be updated to a new random value automatically and with minimum fuss. Descendants of the abstract `Distfunc` class satisfy this requirement.

The abstract `Distfunc` class contains two essential methods. The `CreatefromFile` constructor method allows association of any of the value fields of a `Distfunc` descendant with any distribution, and the `Nextran` method updates all such fields with the next random value from its associated distribution (but see below, Section 3.3.4). In fact, the `CreatefromFile` constructor is designed to associate field values with names in an external file, and simultaneously associate the corresponding fields with the distributions described by those names. Thus, a particular `Distfunc` descendant class might contain the fields named "`c_dens`", "`c_poros`", "`c_depth`", representing density, porosity and depth in some model. The `CreatefromFile` method would then allow associating these three fields with the names "density", "porosity", and "depth" (or any other chosen names) in an external file. Then the external file might be of the form:

```
density     lognormal    3.5   1.5   35.47
depth       uniform      1.0   3.0   2.5
porosity    point        0.4
```


58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

When this file is used to create the `Distfunc` descendant, the `c_dens`, `c_poros`, `c_depth` field values become associated with instantiations of lognormal, uniform, and point distributions respectively with the parameters shown, in such a way that the `Nextran` method updates their values from those distributions.

The ease-of-use (for the programmer) of this technique is probably best illustrated by a code example. The following code fragments illustrate the above example, and is essentially all that is required to use `Distfunc` descendants. The code is pseudo-Pascal, but would be very similar in any other language.

Type definition (this is the simplest possible — in practice, one would also be defining methods that could make use of the fields to calculate soil concentrations and so on) :

```
Soil = Class (Distfunc)
         c_dens,
         c_poros,
         c_depth  :double;

         constructor MakefromFile(f:string);
       end;
```

The constructor MakefromFile method has been added to allow a very simple initialization, just giving the name of a file. The implementation of this type definition, has the form:

```
constructor soil.Makefromfile(f:string);
begin
   inherited CreatefromFile(f,Newdistr(c_dens,”density’,
                              Newdistr(c_poros,”porosity”,
                              Newdistr(c_depth,”depth”,nil))));
end;
```

In this, `f` is a file name, and `Newdstr` is a function signaling that the cited parameters are to be connected with distributions (see below for other options), and designed to allow the repetitive entry of an arbitrary number of field values (note the terminating `nil`).

Then creating examples of `Soil` type is trivial:

```
type
   soil1, soil2, soil3 :Distfunc;

begin
   soil1:=soil.MakefromFile(“file1.dat”);
   soil2:=soil.MakefromFile(“file2.dat”);
   soil3:=soil.MakefromFile(“file3.dat”);
end;
```


58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

Here “file1.dat”, “file2.dat” and “file3.dat” are simply files that contain lines like those illustrated earlier. Thus “file1.dat” may be the file:

```
density     lognormal    3.5   1.5   35.47
depth       uniform      1.0   3.0   2.5
porosity    point        0.4
```

while “file2.dat” may be:

```
density     uniform    15.7 157.25 23.0
depth       table       tabulardata.dat
porosity    point       17
```

and so on.

Of course, the `CreatefromFile` method in the abstract `Distfunc` class also performs checking to make sure that its arguments exist, that all the fields are present in the file, and so forth.

With this type of structure, changing the distributions is straightforward — they simply have to be changed in the input data files. The programmer does not know what distributions might be used. Indeed, none of the distributions that are actually used even have to be known by the time this code is compiled — other descendants of the `Distribution` type may be created after compilation of all the code that handles the `soil` types, but this code will handle them fine (all that is required is that they be registered).

The `Distfunc` type is now almost ideal for modeling. For writing models the fields may be treated as simple scalar variables (as they are). However, the `Nextran` function allows their easy updating to the next Monte Carlo iteration. None of the modeling code has to know anything about distributions, but by melding such code with `Distfunc` descendants (by calling legacy code from within `Distfunc` methods, for example) it can become completely compatible with Monte Carlo analysis.

The `CreatefromFile` method creates instantiations of the distributions called for by the input file, and puts them in a linked list of holder objects created for just this purpose. The anchor of the linked list is defined as a private field in the ancestor `Distfunc` class. The holder objects also contain a pointer to the associated field value. The `Nextran` function traverses this linked list of holder objects, and calls the `Nextran` methods of those holder objects. The Nextran method of the holder objects conditionally calls the `Ran` function of each distribution (see below — Section 3.3.4), and places the result into the pointed-at field.

It turns out in applications that the initialization described above for the fields of `Distfunc` descendants so far described is not quite general enough. One often finds that some parameters needed by models are estimated from other parameters (*e.g.* using correlations). In addition,

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

sometimes it is desirable to initialize vectors of values rather than single values, sometimes it is desirable to initialize strings, and sometimes it is desirable to initialize true constants (*e.g.* molecular weights). These additions lead to examples like the following (this is Pascal-like pseudo-code, close to what is used in Delphi), in which fields can be specified as being set equal to strings, to constants, to distributions, or to the results of a function call that occurs after all fields associated with distributions have been assigned values by the `Nextran` method.

```
type
   Texample=class (Distfunc)
                    name:string;         { name }
                    CASnum:string;       { CASnum }
                    molwt,               { Molecular weight }
                    Kow,                 { Octanol-water coefficient }
                    H,                   { Henry's law constant }
                    LeafyBv,             { Air to leaf biouptake }
                    rootBv: double;      { Root uptake }

                    function airleafBV1:double; { Three different formulae }
                    function airleafBV2:double; {  for different cases }
                    function airleafBV3:double;
                    function henry1:double;     { Estimation formula }

                end;


{ Define an instantiation }
var
   m:Texample;

begin
{ Instantiate and initialize }
      m:=Texample.CreatefromFile('chemical.dat',
                NewStrng(name,'chemicalname',
                NewStrng(CASnum,'CASnumber',
                NewConst(Molwt,'Molwt',nil,         { no default functions }
                NewDistr(Kow,'Octanolwater',nil,    { no default functions }
                NewDistr(H,'Henryslaw',NewFunc('default1',@Texample.henry1,
                                                                  nil),
                NewDistr(LeafyBv,'AirLeafBV',
                              NewFunc('equationA',@Texample.airleafBV1,
                              NewFunc('equationB',@Texample.airleafBV2,
                              NewFunc('equationC',@Texample.airleafBV3,
                                                        nil))),
                NewConst(RootBv,'Rootuptake',
                                          nil)))))))));


{ Note the syntax: }
      NewStrng(stringvariable,'externalstringname',keylist)
      NewConst(variablename,'externalname',functionlist,keylist)
      NewDistr(variablename,'externalname',functionlist,keylist)
      NewFunc('externalfunctionname',@functiondefinition,functionlist)

{ Where  nil will terminate the keylist or functionlist. }
{ You have to get the parentheses correct, but the compiler will force }
{ that! }
```

This code segment illustrates the initialization of a `Distfunc` descendant object in such a way as to associate strings, constants, and distributions with the object fields, with the option of specifying default function(s) instead of constants or distributions. The default functions are specified from an arbitrarily long set of named functions. Such default functions are defined as methods of the object, and they may use any of the other fields in the object. The initialization adds such function associations to a linked list of function holder objects, with its anchor included as a private field of the ancestor `Distfunc` class. As previously described, a call to the `Nextran` method of a `Distfunc` results in the (conditional) updating of all field values associated with distributions. After this, the `Nextran` method scans along the linked list of function associations, makes the corresponding function calls, and updates the associated fields.

This function association allows easy use of estimation procedures for parameter values that are required in models, based on other parameters which have associated uncertainties. For each Monte Carlo iteration, the known parameters are updated to the next random choice from their distributions, and then the correlation functions may be used to estimate other parameters using the updated parameters.

Actually, it is fairly straightforward and useful to generalize further by adding the following three types of initialization function, akin to the `NewStrng`, `NewConst`, `NewDistr`, and `NewFunc` initializers above.

```
NewVectConst(vectorfield; 'external name'; functionlist; keylist)
NewVectDistr(vectorfield; 'external name'; functionlist; keylist)
NewVectStrng(vectorstringfield; 'external name'; functionlist; keylist)
```

These functions that allow a single key name in the external file to be associated with an arbitrary number of constants, distributions, function calls, or string values. The external key name must be followed by the keyword VECTOR, followed by an integer that specifies the number of values to follow. Subsequent lines of the file should than have those values in the same format as would be expected for the single value functions described above, except without any initial keyword. The vector field given should initially be nil, and will be assigned the address of the created vector of values. The initialization vectors will contain distributions, function calls, or mixtures of both (since constants or distributions may be substituted by function calls). Each distribution or function call is inserted into the linked list of distributions and function calls respectively. A record is also kept of details of the vectors, to allow correct destruction of the objects when necessary. The size of the created vectors are available through `vectorsize` and `stringvectsize` methods. The reason for introducing such initialization possibilities is to allow the initialization of time- or age-variable values (see below).

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

3.3.3 Application of Distfuncs to the modeling

The *transporter* and *pathlink* ancestor classes are descendants of the `Distfunc` class. Thus all *transporters* and *pathlinks* may be initialized from files as described for Distfuncs, and all the fields of *transporters* and *pathlinks* are automatically and easily associated with distributions. The fields of *transporters* and *pathlinks* are the parameters for their associated methods, which methods describe transport of chemicals through the environment, so that all the parameters of all models (and hence the whole risk assessment) are automatically and easily associated with distributions.

The methods of *transporters* and *pathlinks* correspond to the models that represent transport of pesticides through the environment. There is no particular requirement that the models be integerated. At one extreme, the methods of the *transporters* and *pathlinks* might be simple or highly complex numerical models that run for every Monte-Carlo iteration; or they could be designed to write control programs for external models and then read the results. At another extreme, they might simply read files of results from external models and interpolate between them to obtain estimates corresponding to the relevant parameter values.

3.3.4 Control structures — tracking uncertainties

As described so far, we have laid out a scheme for performing risk assessments, with every parameter associated with a distribution, and have the Nextran method available in each object (since all descend from `Distfunc`) to update to another random sample. However, no method of control has been indicated yet.

The `Nextran` method of `Distfunc` traverses the linked list of distribution holder objects associating distributions with fields, calling the `Nextran` method of those holder objects, then the linked list of function call holders associating function calls with fields to update the fields using the function calls. In *pathlinks*, the `Nextran` method is augmented to also traverse the linked list of *transporters* for which this is the "to" *pathlink*, and to execute the nextran procedure of each such *transporter*. Updating to the next random iteration of a Monte Carlo routine thus simply requires traversing the linked list of all *pathlinks* (recall that this linked list was set up as the pathlinks were read and created by the control file), and executing their `Nextran` methods. Since all transporters have exactly one "to" *pathlink*, this procedure results in a single call to every `Nextran` procedure of every *pathlink* and *transporter*.

However, such control is too crude. It is certainly necessary to keep track separately of uncertainties and variabilities; and it may be advantageous to allow separate tracking of other distinguishable types of uncertainty. For example, current EPA approaches to toxicity assessment require that toxicity values be treated as constants (using published RfDs and carcinogenic potency slopes, for example). However, there are actually large uncertainties

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

associated with such values, and there are available methodologies to estimate those uncertainties. It is therefore desirable to introduce a mechanism to allow an arbitrary separation of the input distributions to different sorts of uncertainty, which may be treated differently by the control structures. For clarity, since "class" and "type" have meanings for object-oriented programmers, I will distinguish distinguishable types of uncertainty by saying that they are of different "color".

To introduce different "colors" of uncertainty requires adding an extra "color" keyword to the description of all input distributional parameters. This "color" keyword (actually, an equivalent value) is stored in the distribution holders in the linked lists anchored in each *pathlink* and *transporter*. The Monte Carlo update procedure may then be then modified by globally publishing the set of "colors" of distributions that must be updated during the `Nextran` sweep, and the `Nextran` method of the distribution holder objects only makes the distribution `Ran` call and updates the associated field if the distribution "color" is in the list published by the update procedure.

While this level of control is adequate for Monte Carlo analysis, one would also like to access all the individual distributions in order to allow use of their methods to switch to a "preferred" value or an alternate defined value (for example, to force every distribution with uncertainty corresponding to "color" to be set to a "preferred" value) . This is done by adding an extra link in each distribution holder, and linking all into one continuous linked list through all distributions. An alternate method of updating all distributions is then available by traversing this linked list; but its main purpose is for a control mechanism to allow selection of a probabilistic assessment versus a fixed assessment using "preferred" values, or to do this selectively based on the "color" of the uncertainty.

3.3.5 More Control Issues — EPA's "central/high end" approach

The data structures outlines so far cannot handle the EPA's "central" and "high end" approach used in Superfund-type assessments and possibly of use in some aspects of pesticide analyses. In this approach, some undefined but small number of parameters in the modeling are set at plausibly "high end" values, while the remaining ones are set at "central" values. The precise selection of which parameters are set at "high end" has not been explicitly specified in Superfund. In another context (the Hazardous Waste Identification Rule) a well-defined rule was specified that entailed an enumeration of all possible results obtained by setting sets of different "colors" of uncertainty to "high end".

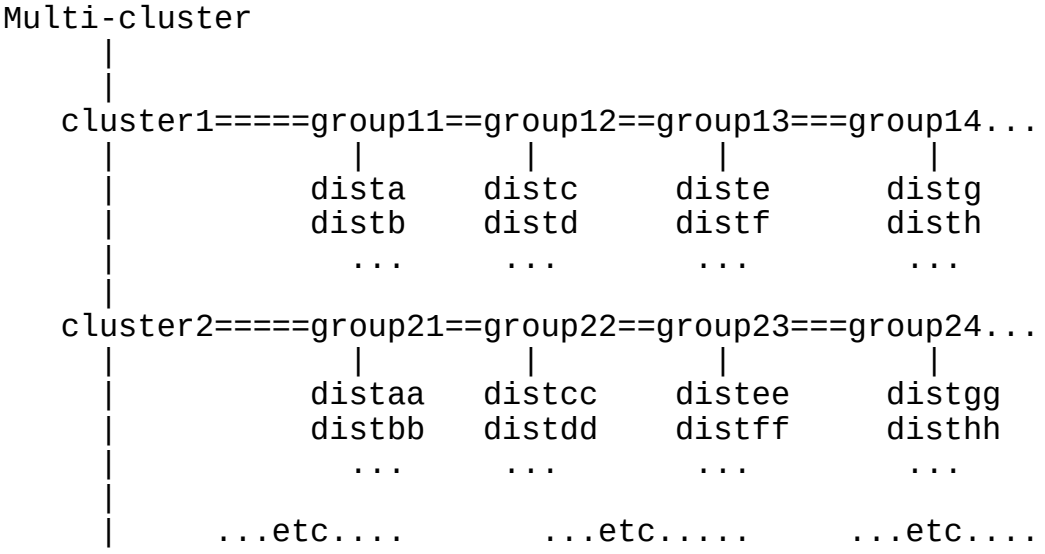
There is a simple extension to the control methods that allows incorporating this approach. Define an `EPADistribution` class that is a descendant of the `Distribution` class. The `EPADistribution` class members behave like distributions, and are initialized with five

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

parameters — central value, high end value, preferred value, a probability for the high value (optional – may be set arbitrarily and ignored), and a control group name (which might be blank).

All `EPADistribution` objects respond to a single set of controlling functions that allow switching their behavior from probabilistic (returning the "high value" with the given probability, otherwise their central value) to deterministic (returning either the "high end" value or the "central value", with selection as described below). This is in addition to the usual (individual distribution, overriding) control methods that allow switching to returning particular fixed values.

At instantiation, all `EPADistributions` are linked into another control structure by using their control group name. For convenience, this control group name can be the (external string representation of the) "color" of the uncertainty distribution (but note the distinction of `EPADistributions` — they always require a control group name, whereas the "colors" of uncertainty have been introduced as external to other distribution types). The `EPADistribution` is linked into a control structure by pointing at its group name. The group names themselves are linked into clusters, and all clusters are controlled by a multi-cluster object. Thus `EPADistributions` are linked into a structure that looks like:



Each group contains a central/high-end flag field that is consulted by each `EPADistribution` that points at that group. A control method of the multi-cluster may then be used to implement any required strategy for selection of central/high-end values - for example, enumerating results over all possible combinations of central/high-end combinations with no more than two high-end values selected in each cluster (*i.e.* no more than two groups of distributions within each cluster).

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

3.3.6 Accounting for time and age variation

Various factors in any risk assessment vary with time or with age. For the pesticide risk assessments, individuals have to be tracked through time as they age, so some method of handling these variations is essential. Indeed, the results of all the models of pesticide exposure have to result in estimates of such exposure as a function of time (and hence age for the individual).

The implementation handles variations with time by requiring that all calculations provide as their output as a function of time. Thus, all `GetConcentration` methods of *pathlinks* and `GetInput` methods of *transporters* require as an input the time at which the concentration or pesticide flow rate is required. The `CalculateRisks` methods of Person *pathlinks* may repeatedly call the *transporters* linked to the Persons to return results for different ages of the individual. The GetInput methods of the *transporters* to Persons translate the age of the individual represented into an absolute calendar time, so that all other parts of the model have to handle only variations with time — they have no knowledge of the age of the receptor.

Many aspects of individual vary with age, so that the Person *pathlinks* often have to handle age-dependent functions, and combinations of such age-dependent functions may be used to define other age-dependent functions. The implementation handles age-dependent functions through a common `Agetype` class, the most important method of which is `AtAge`, which returns the value of the function at a specified age. Other methods defined in this ancestor class include `MaximumIn`, and `AverageOver` that return maximum values within a specified age range, and averages over a specified age range, respectively. By ensuring that these functions only access the `AtAge` method, all descendants of `Agetype` functions have equal access to such functions. Moreover, it is then straightforward to combine age-dependent functions in expressions to define further age-dependent functions, and the same generic functions are automatically available for the compound functions.

4 Databases and Defaults

4.1 Introduction

HESI has made available a set of databases that represent practically the best that would be currently available for performing the assessments required under FQPA. In this chapter we examine these databases and indicate (a) how they fit into the analysis methods discussed in the other chapters of this report, (b) what further information would ideally be required, and (c) what default information could be used in the absence of such information.

To perform assessments one needs information on many factors, including but not limited;

1. chemical concentrations in the environment, including food;
2. human activity patterns;
3. human diet, as a function of age, including ingestion of drinking water and other fluids and including information about long-term trends in national consumption patterns
4. human physiology, as a function of age and sex;
5. age structure and mobility of the population; and
6. agricultural locations and practices.

In what follows, there are some general principles that describe our preferences for finding and using such information in any modeling or simulations:

Principle 1.

For each category of information, the highest preference goes to data measured directly on the most appropriate population. When such data are not available, and it becomes necessary to extrapolate using models from surrogate data, the preference is for the least amount of extrapolation. For example, measured concentrations of pesticides in foods or drinking water supplies are the gold standard. For new pesticides, there are no such measurements, and various extrapolations could be contemplated. Probably the simplest and most reliable is to use analogy to similar pesticides applied in similar circumstances. While complete physical/chemical modeling of behavior of pesticides in the food chain might ultimately be possible, it is currently unlikely to out-perform analogy.

Principle 2.

When new data become available, the architecture and implementation must allow the user to substitute measured values for modeled values. Our implementation is specifically designed for this — either by changing the definition of the distributions (*e.g*. from a theoretical estimate to empirical data) or by substituting a different *pathlink* or *transporter* with methods based on models that specifically include the new data.

Principle 3.

The implementation should be indifferent to the implementation details for distributions (and such details should be easy to change). For example, using measured data, we may sometimes prefer to simulate distributions by directly sampling from the recorded measurements, but at other times we prefer to sample from parametric distributions published (or publishable) in refereed journals. For concentrations of pesticides in foods or drinking water supplies, we prefer to sample from the appropriate database(s) that contain the direct measurements. For the body weights of children, teens, or adults, we might prefer to sample from parametric distributions (*e.g.* Burmaster & Crouch, 1997). The implementation described has the required indifference. It is straightforward to substitute different implementations for distributions — all that is required is to change the initialization method for the distribution.

Principle 4

(Graceful degradation) All values have default values that may be automatically selected. This is built into our implementation. All parameters must be supplied somehow by the user, but the option is available of specifying a default (that may be either a default value, or a default modeling method to estimate the default value).

In the following sections, we examine the databases supplied by HESI, and evaluate

(a) the extent to which they provide the ideal data required by the architecture to meet FQPA requirements,
(b) what defaults are available for them, in cases where fewer data are available, and
(c) what data may be available to augment those provided, and which might assist in better fulfilling FQPA requirements.

The FQPA is likely to require evaluations that cover various time frames for evaluating toxicity, including daily, weekly to monthly, monthly to quarterly, quarterly to yearly and longer, and a full lifetime. Many data are not available on all such time-scales, so much of the following discussion is has been centered around evaluation of the time-frames of interest. The following abbreviations are used:

D	-	one day	Q	-	one quarter
W	-	one week	Y	-	one year

M - one month

L - one lifetime, nominally 70 years, but possibly longer

4.2Ingestion of pesticide residues in foods

4.2.1 Concentrations in foods

The database distributed by HESI lists the allowable tolerances in foods, along with measured concentrations in foods, all concentrations by convention being expressed in ppm. The available data in this case, in order from the crop to the consumer, are:

- tolerance levels,
- field trial data and the fraction of the crop treated,
- market basket data, and
- reduction factors during consumer operations on foodstuffs.

We can visualize the "delivery system" for a pesticide applied to a given crop as consisting of a sequence of components, with the characteristics listed below. Ideally, all the listed characteristics would be measured, but we suggest the various defaults shown for use at varying levels of sophistication of modeling. In the example given in the discussion of the implementation, all these components have been incorporated into a single *transporter* (see Figure 3.1). However, with sufficient data and sufficiently detailed models, it might be more appropriate to separate them into a sequence of *pathlinks* and *transporters* (Figure 4.1).

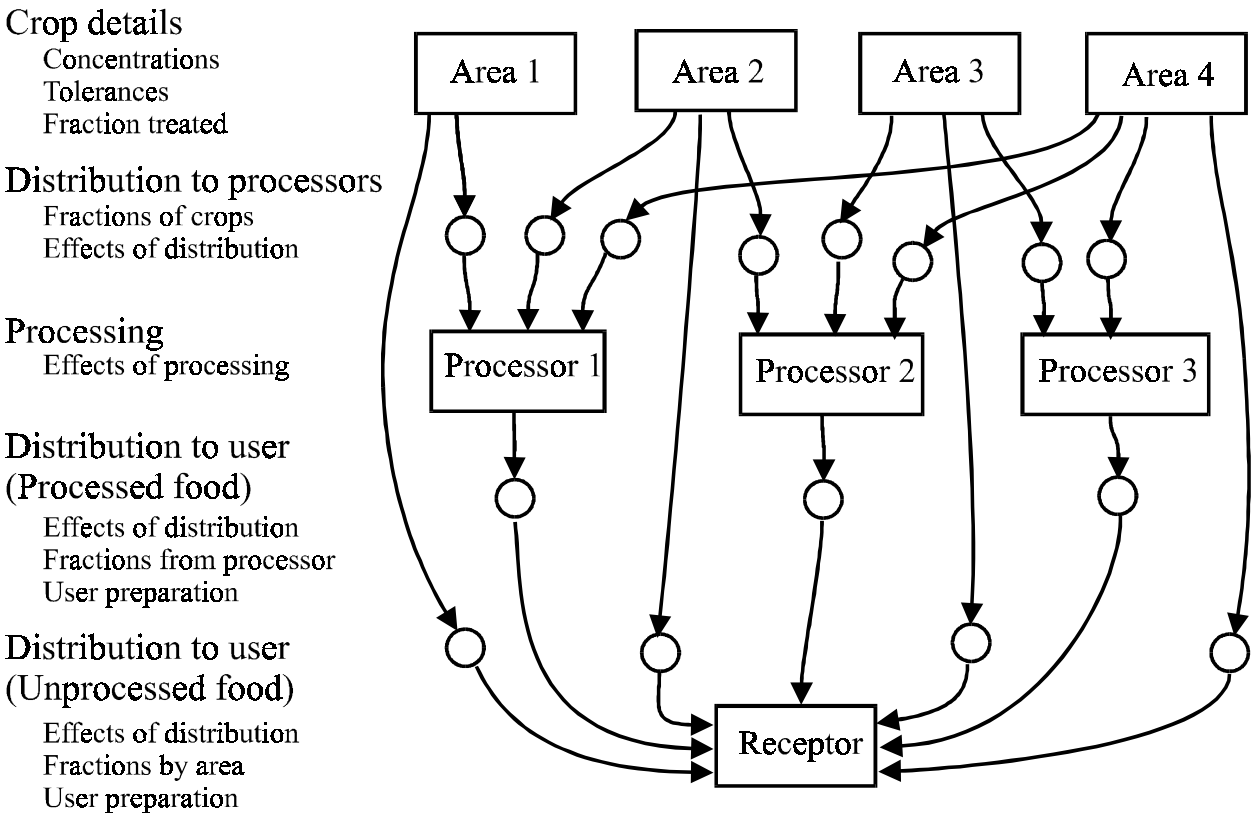


Figure 4.1 Example of an expanded food pathway, showing a possible breakdown into *pathlinks* and *transporters*, and the models or measurements handled by each component.

Component **Characteristic**	**Default**	**Comment**
Geographical crop areas:		
Fraction treated	Uniform with area	Different fractions of the crop may be treated in different geographic regions
Seasonality	Uniform throughout season	Includes seasonality of crop and of pesticide application combined

Tolerance	Uniform with area	Tolerances could feasibly be different in different areas
Crop at the input to the distribution system		(At the farm gate)
Pesticide concentration	Measured	If measurements include seasonality, geography.
	Measured + model	Modify measurements to account for seasonality, geography from crop areas.
	Measured uniform	Omits seasonality &/or geography of crop areas.
	Modeled from tolerance	Taking account of any known processes, or limitations (e.g. lower application rates)
	Tolerance(s) + fraction treated	
Distribution system		
Geographic matrix	Uniform	Every market supplied in proportion to crop production
Crop to Food	Measured	Effect of processing crops to produce marketable foods.
	No effect	
Food at the retail level		(Market basket)
Pesticide concentration	Measured	If measurements include seasonal, geographic variations
	Measured + model	Modify measurements to account for all prior components of delivery system.
	Measured uniform	Omit information available from prior components.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

	Modeled from above	Using measurements/defaults closer to the original crop, adding in any effects from processor to retailer
	Equal to crop at farm gate	The simplest model

Prepared food

Pesticide concentration	Measured	If measurements includes seasonal, geographic variations
	Measured + model	Modify measurements to account for prior component of delivery system
	Measured uniform	Omit information available from prior components
	Modeled from above	Using measurements/defaults closer to the original crop, adding in the effects of food storage and preparation.
	Equal to retail level concs.	The simplest model.

The ideal information in this case would be measured concentrations of the pesticide in as-eaten food, stratified by geographic area, season, and perhaps by time since the introduction of the pesticide, together with such factors as the fraction of the U.S.-eaten crop treated. Since some pesticide concentrations may build up slowly with time (e.g. in milk, due to gradual build-up of pesticide levels in soils, with subsequent transfer to the milk supply), the ideal would be to have repeated measurements over many years. Such repeated measurements might allow correlation of any variations with such factors as fraction of crop treated (ideally taking account of the fraction treated in each growing area, and the location of the final eater) and weather. In the implementation given in Chapter 3, ideal measurements could be incorporated into the *transporter* to the receptor, with the previous *pathlinks* simply supplying such information on the tolerance levels and fractions of the crop treated as would be required to assess the expected variation in food concentrations with such factors.

Less detailed measurements could be used directly, with an assumption of uniformity across geographic area, season, and/or calendar time; or (preferably) such less detailed measurements could be augmented by modeled modifiers to take account of such factors. For example, models

of pesticide application, distribution, processing, preparation, and so forth might predict a certain seasonality and geographic pattern to the expected concentrations of the pesticide. If the only measurements available were aggregate averages across regions and throughout the year in prepared foods, the modification factors implied by the modeling might be applied in such a way that the regional, annual average matched the measurement, while incorporating the regional and seasonal variations.

In the absence of direct measurements in prepared foods (the usual situation), it is necessary to start with measurements (or default assumptions) closer to the crop, and apply models to estimate the concentrations in prepared foods. The models have to be based on measurements that correspond to the processes occurring, but that are not direct measurements of pesticide concentrations in the food delivery system (*e.g*. laboratory or field measurements of reduction factors due to food preparation).

Thus next best after direct measurements in as-eaten food would be market basket information stratified by geographic area, season, and calendar time. Once again, less detailed measurements might be augmented by modifiers obtained from models of the rest of the food delivery system, and concentrations in as-eaten food may be estimated by models (that may be as simple as application of measured preparation factors).

The same approach may be taken to use the best data available from anywhere in the food delivery system. Measurements further from the ultimate consumer may be used, but must be augmented by more and more modeling (although the modeling may be extremely simple and conservative — in such a direction as to overestimate concentrations in as-eaten foods). The construction of the models requires further data that may be available from other sources. For example, for all major crops in the U.S., U.S DA (1997) routinely publishes large databases on the Web that detail the amount crop harvested in each county in a particular year.

4.2.2 Consumption rates of foods

There are two databases now in use for estimating dietary intake. First, the U.S.DA collected data on dietary intake from 1977–1979. Many years ago, the U.S.EPA analyzed these consumption data for 23 sub-populations. The Agency continues to distribute these results (and to use these results) in an old format, but almost all other parties no longer use these data. Second, more recently, the U.S.DA collected additional data in the CSFII Survey of 1986–1989. Soon thereafter, TAS, a consulting company in Washington, DC, analyzed these data using more detailed methods. This newer analysis of the database is not available to the public or for public review. Now that TAS has split into two separate consulting companies (one is called TAS/Environ and the other is called NoviGen Sciences), the only way to see or use the data is to buy a license good for a year.

The TAS analysis of the CSFII Survey is the one more commonly used. It can provide information on a geographic and age-specific basis, and in such respects is almost ideal for FQPA requirements. However, there are at least two defects.

First, all but one of the dietary studies of which we are aware (including the CSFII Survey) have a common design problem — the survey covers only a few days, and so cannot account for longer-term variations in individual food intake. These surveys generally measured also the bodyweight of the persons surveyed, so that correlations with bodyweight are possible. Such surveys may give excellent information on population average amounts of food ingested, but they are not ideal for tracking individual food consumption over time (even over a single season). Various default strategies may be adopted for extrapolating to individual food intakes over time:

- After selecting an individual, select an arbitrary age for that individual, then randomly choose a food consumption pattern (from the food consumption database) corresponding to that age and bodyweight. Assume that the food consumption pattern remains constant throughout the persons lifetime.

- After selecting and individual, choose a set of ages for the individual (with corresponding bodyweights), and randomly obtain food consumption patterns (from the food consumption database) corresponding to those ages and bodyweights. Interpolate the food consumption patterns between ages in some manner (*e.g*. linearly in fraction of total diet).

- Either of the above, but impose seasonal variations corresponding to the seasonal variations seen in the average food consumption data. Or impose seasonal variations corresponding to the seasonal variations observed in food markets. (The former uses data from within the food survey database; the latter from external sources, *e.g*. U.S.DA consumption estimates).

- For each individual at each age considered (*e.g*. every day in their life), choose a food consumption pattern randomly (from the food consumption database).

To augment such default approaches, it may be possible to use information in the one long-term study known to us to assist in extrapolating from a few days of consumption to a year of consumption. Although researchers at the U.S.DA measured and published these data almost 10 years ago (Basiotis *et al*, 1989), exposure and risk assessors are largely unaware of them.

The second defect of the food consumption databases that are currently available is their age. The most recent is almost 10 years old, and food consumption patterns for certain foods can change substantially in such a time period. It may, however, be possible to augment the databases by adding corrections based on continuing surveys of food consumption (*e.g*. U.S.DA 1970–1995).

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

No matter how good the market basket food consumption surveys are, they will always have to be extrapolated or interpolated in some manner to fully support the FQPA, since analyses of potential lifetime exposures for individuals will require extrapolation to the future. The method of extrapolation is open to selection. The defaults suggested above implicitly extrapolate as though the most recent available data correspond to all future times; while it may be possible to incorporate some trends from the continuing surveys (although this would require careful analysis to prevent implausible results).

Finally, it may be possible and expedient to supplement or replace certain parts of the market basket surveys of food consumption with information from other sources, including fitted and published parametric distributions (*e.g.* for fish ingestion, Ruffle *et al.*, 1994).

4.3Ingestion of pesticide residues in drinking water

4.3.1 Concentrations in Raw/Finished household water

Database for Atrazine and Simazine

As a part of a "special review," a large manufacturer of atrazine, a pesticide widely used on corn, soybeans, and other crops throughout the Midwest and elsewhere, has collected data from state agencies and other sources on the prevalence and concentration of on atrazine and simazine in raw water (and some finished water) used for drinking water. The database now includes data for >5 years from 21 "high-use" states. The data were collected quarterly (and sometimes more frequently) from rivers and streams, lakes and impoundments, and shallow and deep ground water that supplies water to a "community water supply," a term defined in and under the federal Safe Drinking Water Act (SDWA) of 1974, as amended. Under the SDWA, large community water supplies must send samples to a laboratory for chemical (and microbial) analysis more frequently (perhaps monthly or weekly) than do small systems (usually annually or quarterly, depending on the analyte), and private water supplies need not send any samples to a laboratory for analysis. When the samples have concentrations reported above the detection limit, they typically have measurement errors of a factor of <2 in the parts-per-billion range. The database also includes the number of people using the particular water supply.

Database for Product R

For this coded, real product, the industry has provided three tables: (i) a table on market share (percent of acres grown), disaggregated by crop and state; (ii) a table of concentrations (ppb) in ground water from private wells in vulnerable locations; and (iii) a table, for 39 states for 1979 - 1997) listing (a) the number of samples, (b) the total number of wells, and (c) the number of wells with residues in the last sample.

4.3.2 Analyses and defaults for water concentrations

Using the database for atrazine and simazine, the manufacturer has sent summary reports to the US EPA regarding the continuing registration of the product. A key feature of the database is that it includes a large proportion of values reported as "nondetects" (with a detection limit at or near 1 µg/l). In other words, the concentrations in the water are either nonexistent or so low that current instruments cannot measure them. The database also includes numerical estimates for the nondetects, based on the (well founded) assumption that all the data follow a lognormal distribution. For simulations under the FQPA of 1996, one may sample directly and randomly from the database (using both the values above the detection limit and the numerical estimates for the nondetects) than to sample from parametric distributions fit to the data.

The estimates for the non-detects currently included in the database account only for the distributions of each chemical (atrazine and simazine) separately, and so is strictly suitable only for an aggregate analysis of either chemical separately. In an analysis of both chemicals simultaneously (slightly beyond the remit of the current discussion, but within the capabilities of the architecture and implementation method described), it would be necessary to extend the methodology for generating fill-in values to include the correlation structure of the measurements. The obvious extension would be to assume that the logarithm of the measurements of concentrations of the chemicals are jointly normally distributed with a variance-covariance matrix that can be estimated in a manner similar to that described for the individual distributions.

Direct sampling from the augmented database preserves many of the important correlations, including correlations with geographical area and season changes. The observations are measurements of short-term samples taken quarterly, and so must be interpolated and extrapolated. The interpolations and extrapolations in time that are required, together with suggested defaults, are:

D to W to M	Most of the concentration data for atrazine and simazine come from quarterly measurements of short-time samples. Each sample strictly represents the concentration during a short period (minutes) on the day of sampling. The database contains little information on variability during periods up to a quarter. Various default approaches are: • Randomly select a relevant (geographic area, season) sample, and assume this concentration for a complete quarter • Interpolate (linearly or otherwise) between successive quarterly measurements at the same sampling location, randomly selected for correct geography, season. • Randomly select a relevant (geographic area) sampling location, and fit a model of seasonal variation with a random component to available data at this sampling point. Randomly sample using this model. • Select all relevant sampling locations, and fit a model a seasonal variation with a random component and different mean values for each location. Randomly select one of the locations, and randomly select from the model as applied at this location.
Q	• Randomly select from relevant samples (geography, season) • Apply models with longer-term trends to all the data, then randomly select a relevant sampling location and obtain a value from the model applied at that location.
Y to L	As the length of time increases to a year or more, the variability and uncertainty in concentrations will change in unknown ways. Default approaches include: • Assume the average concentration of the last year of measurement at an appropriate sampling location. • Fit a model with longer term trends to one or all sampling locations, then randomly select a relevant sampling location and apply the model. • Predict the concentration using a multimedia transport model coupled with predictions on future use, (*e.g.* the GENEEC model — GENeric Expected Environmental Concentration model mentioned in EPA documents).

For Product R, if a multimedia transport model were available and calibrated for the types of crops, soils, and aquifers found in the areas where Product R is used, it might be possible to predict concentrations from the use data (*e.g.* the GENEEC model?). Such models are likely to require data from the market share database and from an agriculture database on crop locations.

Potential defaults for a pesticide such as Product R are:

- Assume no exposure. This might be appropriate in view of the FQPA's requirement for adequate information on exposures.
- Application of a multimedia transport model(s). Such a model or models might be as simple as an analogy with another similar pesticide, or might be exceedingly complex (requiring

detailed information on soil types, weather, terrain, drinking water sources, and so forth, in the area where the pesticide is applied to crops).

4.3.3 Ingestion rates for drinking water

As a part of its 1977–1978 Nationwide Food Consumption Survey (NFCS), the US Department of Agriculture collected data on the total water intake and tap water intake for some 26,000 people, aged birth to >65 years. Ershow and Cantor (1989) published summary statistics (*e.g.*, arithmetic means and certain percentiles) for the water intake of males and females in the whole US and in some regions of the US. Subsequently, Roseberry and Burmaster (1992) separately fit lognormal distributions to the data for the variability in the water ingestion rates for males and females in the age ranges 0 to 1 year; 1 to 11 years; 11 to 20 years; 20 to 65 years; and over 65 years. Neither Ershow and Cantor nor Roseberry and Burmaster quantified the uncertainty in the data or the fitted distributions. Since the popularity of bottled water and cola drinks has increased substantially since 1978, there is more uncertainty in the distributions for ingestion of tap water than in the distributions for the ingestion of total water. With a modest amount of work, it is not difficult to fit distributions that are continuous in age, with separate analyses for different regions of the U.S..

The principal default extrapolations required here are to longer periods:

D	Since the data arise from a 3-day survey, the data and the fitted distributions apply directly. The same types of default are required as for the food consumption surveys (Section 4.2.2).
W to M to Q	As the length of time increases from week to month and quarter, the variability and the uncertainty in ingestion rates change in unknown ways. We have no information on the autocorrelations in water ingestion rates for an individual or for a population.
Y to L	As the length of time increases further, the variability and the uncertainty in ingestion rates change in unknown ways. We have no information on the autocorrelations in water ingestion rates for an individual or for a population.

Ershow *et al.* (1991) published summary statistics for pregnant women and certain other sub-populations. The statistics included the variability in the ingestion rate of water by pregnant women (n = 188), lactating women (n = 77), and nonpregnant and nonlactating women (n = 6,201), all based on data collected in 1978 by the US Department of Agriculture in its Nationwide Food Consumption Survey (NCHS). More recently, Burmaster (1997, in press) has fit correlated bivariate lognormal distributions to the variability in water ingestion rates (total water intake and tap water intake) and body weight for these three groups of women. Ershow *et al.* did not quantify the uncertainty in the data, nor did Burmaster quantify the uncertainty in the fitted distributions.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

4.4Exposure to pesticide residues in the home

4.4.1 Concentrations in and around homes

Companies in the pesticide industry have supplied data measured for the household use of "Chemical X," a compound or group of compounds used in three ways (i) application to cracks and crevices, (ii) broadcast in or near living spaces, and (iii) application to turf (lawns). For Chemical X, the data contains concentration information useful — directly or indirectly — to estimating doses experienced by professional or nonprofessional applicators via (i) dermal contact with surfaces and materials, (ii) oral contact with surfaces or materials, and (iii) inhalation of vapors or mists at different times during or after an application. However, application of these measurements will require the use of models relating the measured concentrations to the doses received. We are unaware of any research on the reliability and availability of such models.

For exposures in homes, the following sequence of defaults appear most appropriate:

- Measurements of doses received by persons of various ages in practical application situations (through intrusive measurement methods, such as measurement of blood concentrations).
- Measurements of concentrations in air through personal monitors, and on skin and clothing surfaces, of people in practical application situations.
- Measurements similar to those in the currently available database — air concentrations, and skin and surface concentrations, but of surrogate to actual application situations.
- Generic estimates of doses and/or concentrations in air and on skin and other surfaces, based on the physical/chemical properties of the pesticide (and, presumably, on correlation with measurements on other materials, possibly other pesticides or a substitute).

4.4.2 Exposure rates in and around homes

The available database does not contain any information on the frequency of use for these products in homes, although for some types of product (particularly those applied by professionals), there is likely to be a recommended range of repeat application rates. The NHAPS database (Section 4.6.3) contains some generic information on the frequencies of application of pesticides (of unspecified type) in and around homes over the six month period prior to the interview time.

The rate of use in homes might be estimated using the following set of defaults:

- An average rate computed from marketing rate and expected end use population (stratified on geographic area). Assume uses occur with poisson statistics, possibly with different average

rates for weekdays and weekends, depending on the product, and with seasonal modifications appropriate to the pesticide type. Incorporate modifications to account for instructions with the pesticide, or the types of use (*e.g.* if the user instructions call for a fixed number of repeat applications at 1 week intervals, then incorporate such information).

- An average rate estimated from the NHAPS data on recalled pesticide applications within 6 months, modified by the expected type of use of the product. Assume uses occur with poisson statistics, possibly with different average rates for weekdays and weekends, and with seasonal modifications appropriate to the pesticide type, depending on the product. Incorporate modifications to account for instructions with the pesticide, or the types of use (*e.g.* if the user instructions call for a fixed number of repeat applications at 1 week intervals, then incorporate such information).

Exposures are likely to largely depend on models that incorporate many physiological variables like inhalation rates and skin surface areas (Section 4.5) and social variables like housing information, household sizes and makeup, and other activity patterns (some are discussed in Section 4.6).

4.5 Physiological variables

4.5.1 Body weight

The National Center for Health Statistics (NCHS) has reported data from the NHANES II Survey for the body weights of males and females, aged 6 months through 74 years, living in the United States between 1976 and 1980. These data, reported as percentiles of variability in the distribution of body weight disaggregated by gender and age, have the characteristics of a cross-sectional "snapshot" of the general population with no longitudinal information. These data are synoptic and contain negligible uncertainty from measurement error. Burmaster and Crouch (1997) have recently fit parametric distributions for the variability in body weight for males and females as a continuous function of age. The fitted distributions contain relatively little uncertainty from the statistical procedures used but they contain an unknown amount of uncertainty arising from the age of the original measurements by NHANES.

Various default assumptions have to be made to use these data in an FQPA analysis extending over various time spans:

D	The NHANES data and the fitted distributions apply directly (with no adjustments) for males and females as a continuous function of age.

W to M to Q	As the length of time increases to a week, month, or quarter, short-term or seasonal variability may occur in an individual's weight, but the NHANES data contain no longitudinal information across time. The data collected by USDA (Basiotis *et al.*, 1989) contain some information for ~36 adult men and women on weekly variability in weight for one year. Various default assumptions can be made pending further information or analysis: • Assume that a person has constant weight over these time scales. • Add extra uncertainty in bodyweight, based on the data in Basiotis *et al.*
Y to L	For the longer term, there appear to be no available data. Our suggested default: • Assume that a person's weight remains at a constant percentile of the weight distribution for the general population, with the understanding that the weight for a particular percentile changes as a continuous function of time.

For pregnant women and some other sub-populations, we may lack certain data. Thus while average weight gain in pregnancy may be known, we have no database providing distributional information as a function of time since conception. While this may add to the uncertainties involved, variations in weight are unlikely to substantially affect distributional estimates of risk.

4.5.2 Skin surface area

Skin surface areas are generally estimated using correlations with body height and body weight, based on approximately 400 old measurements of skin surface areas in various humans. Examples of such correlations are given by U.S.EPA (1995; 1996). While the Agency's formula works reasonably well, it suffers from problems of co-linearity not present in a simplified model (Burmaster, 1997) that predicts the variability in skin area as a function of body weight for males and females without sacrificing accuracy or precision. The accuracy of these correlations in specific sub-populations (*e.g.* pregnant women) has never been examined. However, the uncertainties are small enough that they are unlikely to have much effect on risk assessments.

4.5.3 Inhalation rate

As part of its research program, the California Environmental Protection Agency (Cal/EPA) in Sacramento has funded several research projects to develop distributions for the inhalation rates of children and adults, with correlations with based on the activity patterns and energy use of those individuals (Adams, 1993; Wiley *et al.*, 1991a and 1991b). From these data, Cal/EPA has developed and published empirical probability distributions for the daily average breathing rates for adults and children (Tables 3.19 and 3.20, respectively, in Cal/EPA, 1996).

Various default assumptions will have to be applied for analyses covering different periods:

D	Since the data arise from a 1-day activity (recall) survey, the data and the fitted distributions apply most directly.
W to M to Q	As the length of time increases from week to month and quarter, the variability and the uncertainty in inhalation rates increase in unknown ways. We have no information on the autocorrelations in breathing rates for an individual or for a population for a week, a month, or a quarter. Possible default (as for bodyweight and other physiological parameters) • Select a percentile, and assume that a particular individual always breathes at that percentile of the distribution of breathing rates for every activity pattern.
Y to L	As the length of time increases further, the variability and the uncertainty in ingestion rates increase in unknown ways. We have no information on the autocorrelations in breathing rates for an individual or for a population as a person ages. Suggested default: • Select a percentile, and assume that a particular individual always breathes at that percentile of the distribution of breathing rates for every activity pattern.

As for most physiological variables, there is no information on pregnant women or other interesting sub-populations (*e.g.* asthmatics). However, it seems unlikely that the uncertainties involved will have much effect on an overall risk assessment under FQPA.

4.6Social variables

4.6.1 Duration of residence at one location

From time to time, the US Bureau of Census publishes data from housing surveys that report current residence time disaggregated by year, by region of the country, and by other variables. These data come from answers to the question "How long have you lived in your current residence?" posed to a cross-section of the population at a point in time. Also, from time to time, the Canadian federal government has conducted surveys (Richardson, 1997), that tabulate answers to three questions: (i) "How old are you?", (ii) "How long have you lived in your current residence?" and (iii) "How long have you lived in this city (or town or rural area)?" By design, the Canadian survey collects more information so that it can distinguish moves within an area to moves outside an area, disaggregated by age and gender.

From the data collected by the US Census Bureau in 1985 and 1987, Israeli & Nelson (1992) inferred distributions for total residence time, the variable of interest in risk assessment. Israeli and Nelson fit distributions to data for all residences, for rented and owned residences, for urban and rural residences, and for residences in the "northeast, midwest, south and west." While this publication has many strengths, it cannot overcome the cross-sectional design used by the Census Bureau. Thus, the publication contains no information on temporal and social trends in people's behaviors over years or decades, except that it showed consistency of the moving rate over a time span of about 10 years.

To the best of our knowledge, no one has fit probability distributions to the data from Canada. Since the Canadian survey disaggregates the counts by province, by population of the city or rural area where the person lives, by age, and by moves outside or inside the same area, the database contains much more information -- making it ripe for statistical analysis to find the patterns and to fit distributions.

Plausible default assumptions for period of residence are thus:

- The distributions implied by Israeli & Nelson’s analysis (or a similar analysis). This has the disadvantage that we do not know where the individual moves — and such information is necessary to complete tracking the individual through their lifetime.
- Assume an individual remains in one location for their entire life.

4.6.2 Duration of job tenure

Every year, to update the Current Population Survey (CPS), the US Bureau of Labor Statistics (BLS) and the US Census Bureau collect data from tens of thousands of adult men and women (aged 16 years and more) who answer these questions: (i) "How long have you worked in this profession?" and (ii) "How long have you worked for your current employer?" In the most recent survey during one week in February 1996, the CPS included >50,000 households in >700 areas which represent ~2000 geographical areas in the US. The survey includes questions about occupational mobility, job training and length of employment with the current employer (often called "job tenure"). The BLS has reported detailed statistical summaries of these data in January 1987, January 1991, and, most recently, February 1996. In addition, researchers may obtain CD-ROM disks that contain more detailed information from the BLS.

Extending analyses published by Shaw and Burmaster (1996), Burmaster (1997) has recently fit parametric probability distributions to the most recent BLS data to infer the projected job tenures from the current job tenures for adult men and women in 22 occupations and 31 industries. In addition to the parametric distributions, the paper (in review) also reports the 10th, 20th, 30th, 40th, 50th, 60th, 70th, 80th, 90th, 95th 97.5th and 99th percentiles of the projected distributions. By comparing results from Shaw and Burmaster (1996) from the 1987 CPS and Burmaster (1997) for the 1996 CPS, we can observe secular trends in the economy.

For use in FQPA risk assessment, similar comments apply as for duration of residence (Section 4.6.1).

4.6.3 Activity patterns

The US EPA has published the raw data and statistical summaries of its 1992–94 National Human Activity Pattern Survey (NHAPS). The complete database, now available on CD-ROM, includes information from the 24-hour recall diaries of 8236 people, males and females of all ages (Klepeis, Tsang & Vehar, 1996; Tsang & Klepeis, 1996a; Tsang & Klepeis, 1996b). A similar study is under way in Canada (Leech *et al.*, 1997). In 1998, the Electric Power Research Institute (EPRI) will make the full databases of two other activity pattern surveys funded by them (Silvers *et al.*, 1994) available to the public via CD-ROM(s) or the Web.

From these data, the US EPA has published descriptive statistics and histograms for the time spent and frequency of occurrence for 22 activities (that were experienced by more than 50 respondents) in each of 39 locations. Some of this distributional information now also available in the latest draft EPA Exposure Factors Handbook (U.S. EPA, 1996). These histograms do not preserve the correlations (and autocorrelations) in activities for the same person or for different people, so it will be more reliable to sample directly from the database than to sample from the histograms published by the Agency.

58 Charles Street, Cambridge, Massachusetts 02141
617-225-0810 • FAX: 617-225-0813 • info@CambridgeEnvironmental.com

While direct sampling from the database mostly overcomes the problem of correlations in activities for a single day, it cannot overcome the lack of long-term information. The implementation has to follow individuals through long periods, so that assumptions have to be made about individual's daily habits based on a single day of information. For some pesticide exposures, this may be alleviated somewhat by the explicit questions in NHAPS about pesticide exposures in the last 6 months. However, it may be necessary to augment the database information with additional assumptions about rare events (such as multiple simultaneous and unusual exposures) and about seasonal use patterns. Some data about the former events may be available in the literature on hospitalization and poisoning.

The NHAPS Survey did include 12 subgroups: gender, age (including some children whose parents responded for them), race, Hispanic, education, employment, census region, day-of-week, season, asthma, angina, and bronchitis/emphysema — but it does not provide information on the activity patterns for pregnant women.

4.7Database references

Adams, W.C., 1993. Measurement of Breathing Rate and Volume in Routinely Performed Daily Activities, Final Report A033-205, California Air Resources Board, Sacramento, CA, June 1993.

Basiotis, P.P., R.G. Thomas, J.L. Kelsay, and W. Mertz, 1989, Sources of Variation in Energy Intake by Men and Women as Determined by One Year's Daily Dietary Records, American Journal of Clinical Nutrition, Volume 50, pp 448–453.

Burmaster, D.E. and E.A.C. Crouch, 1997, Lognormal Distributions for Body Weight as a Function of Age for Males and Females in the United States, 1976 - 1980, Risk Analysis, Volume 17, Number 4, pp 499–506.

Burmaster, D.E., 1997a. Distributions of Job Tenure for Men and Women in Selected Industries and Occupations in the United States, February 1996, Human and Ecological Risk Assessment, in review.

Burmaster, D.E., 1997b Lognormal Distributions for Total Water Intake and Tap Water Intake by Pregnant and Lactating Women in the US, Risk Analysis, in press

Cal/EPA, 1996. California Environmental Protection Agency, Exposure Assessment and Stochastic Analysis, Public Review Draft of a Technical Support Document, Air Toxics Hot

Spots Program Risk Assessment Guidelines, Part IV, Office of Environmental Health Hazard Assessment, Berkeley, CA, December 1996.

Ershow, A.G., L.M. Brown, and K.P. Cantor, 1993. Intake of Tap Water and Total Water by Pregnant and Lactating Women, American Journal of Public Health, Volume 81, Number 3:328–334

Ershow, A.G. and K.P. Cantor, 1989. Total Water and Tap Water Intake in the United States: Population-Based Estimates of Quantities and Sources, Federation of American Societies for Experimental Biology, Bethesda, MD 20814

Israeli, M. and C.B. Nelson, 1992. Distributions and Expected Time of Residence for U.S. Households, Risk Analysis, Volume 12, Number 1, pp 65–72

Klepeis, N.E., A.M. Tsang, and J. Vehar, 1996. Analysis of the NHAPS Respondents from the Standpoint of an Exposure Assessment, EPA/600/R-96/074, July 1996.

Leech, J.A., K. Wilby, E. McMullen, and K. Laporte, 1996. The Canadian Human Activity Pattern Survey: Report of methods and population surveyed. Health Canada — CDIC 17 (3/4)

MacIntosh, D.L, J.D. Spengler, H. Ozkaynak, L. Tsai, and P.B. Ryan, 1996. Dietary exposure to selected metals and pesticides. Environ. Health Perspect. 104:202–209.

Ott, W.R.,1984. Exposure estimates based on computer generated activity patterns. J. Toxicology: Clinical Toxicology 21:97–128.

Ott, W., J. Thomas, D. Mage, L. Wallace, 1988. Validation of the simulation of human activity and pollutant exposure (SHAPE) model using paired days from the Denver, CO, carbon monoxide field study. Atmospheric Environment 23:2101–2113.

Roseberry, A.M., and D.E. Burmaster, 1992, Lognormal Distributions for Water Intake by Children and Adults, Risk Analysis, Volume 12, Number 1, pp 99–104

Ruffle, R., D.E. Burmaster, P.D. Anderson, and H.D. Gordon, 1994, Lognormal Distributions for Fish Consumption by the General US Population, Risk Analysis, Volume 14, Number 4, pp 395–404

Shaw, C.D. and D.E. Burmaster, 1996, Distributions of Job Tenure for US Workers in Selected Industries and Occupations, Human and Ecological Risk Assessment, Volume 2, Number 4, pp 798–819

Silvers, A., B.T. Florence, D.L. Rourke, and R.J. Lorimor, 1994. How children spend their time: a sample survey for use in exposure and risk assessment. Risk Analysis 14:931–944.

Tsang, A.M. and N.E. Klepeis, 1996. Results Tables from a Detailed Analysis of the National Human Activity Pattern Survey (NHAPS) Response, Draft Report prepared for the US Environmental Protection Agency by Lockheed Martin, Contract Number 68-W6-001, Delivery Order 13.

Tsang, A.M. and N.E. Klepeis, 1996. Descriptive Statistics Tables from a Detailed Analysis the National Human Activity Pattern Survey (NHAPS) Data, EPA/600/R-96/074, July 1996.

U.S. EPA, 1995. Exposure Factors Handbook, External Review Draft, Exposure Assessment Group, EPA/600/P-95/002A, Washington, DC, June 1995

U.S.EPA, 1996. Exposure Factors Handbook: Volume I - General Factors - EPA/600/P-95/002Ba; Volume II - Food Ingestion Factors - EPA/600-P-95/002Bb; Volume III - Activity Factors - EPA/600/P-95-002Bc. August 1996 . Available at http://www.epa.gov/ORD/WebPubs/exposure/. (Update to Exposure Factors Handbook EPA/600/8-89/043 - May 1989)

U.S.DA, 1997. U.S. Department of Agriculture, 1997, Crops County Data, http://www.usda.gov/nass, Washington, DC

U.S.DA, 1970–1995, Food Consumption, Prices, and Expenditures, 1970–1995, Economic Research Service, SB-939, Washington, DC.

Whitmore, R.W., F.W. Immerman, D.E. Camann, A.E. Bond, R.G. Lewis, and J.L. Schaum, 1994. Non-occupational exposures to pesticides for residents of two U.S. Cities. Arch. Environ. Contam. Toxicol 26:47–59.

Wiley, J.A., J.P. Robinson, T. Piazza, K. Garrett, K. Cirksena, Y.T. Cheng, and G. Martin, 1991a. Activity Patterns of California Residents, Final Report from Contract Number A6-177-33, California Air Resources Board, Sacramento, CA, May 1991

Wiley, J.A., J.P. Robinson, Y.T. Cheng, T. Piazza, and L. Stork, 1991b. Study of Children's Activity Patterns, Final Report from Contract Number A733-149, California Air Resources Board, Sacramento, CA, September 1991