5. DEVELOPMENT OF THE HYDROSALINITY MODULE

As mentioned in the preceding chapter one of the main reasons for restructuring the *ACRU* model was to make it easily extendable and modular. The hydrosalinity module of *ACRU* is developed in the restructured version of *ACRU, viz. ACRU2000*. The new structure of *ACRU* has facilitated development of the hydrosalinity module with little interference with the existing modules. The term “module” in *ACRU2000* refers to groups of objects with a common overall purpose (Kiker and Clark, 2001). Since this module is developed in the *ACRU2000* environment, it inherits the basic structure and objects of the model. Hydrosalinity models, in general, involve the interaction of hydrological and salinity related processes. Thus, the hydrosalinity module of *ACRU* also involves the interaction of hydrological processes, as determined by the hydrological modules of *ACRU*, and salinity related processes. Hence, the hydrosalinity module of *ACRU* is designated as *ACRUSalinity*. This chapter describes the development of the module with special emphasis on the processes and the interaction between various objects involved in these processes.

5.1 Modelling Approach and Basic Objects in *ACRUSalinity*

A series of steps is followed in the development of *ACRUSalinity.* First, an extensive review of the *ACRU* and hydrosalinity models was undertaken so as to get the basic idea on the way processes are represented in these models (Chapters 2 and 3). The next step was conceptualisation, where hydrological processes from *ACRU* and relevant salinity processes from the hydrosalinity models are conceptually linked to accomplish the required tasks. This step was then followed by a review of relevant UML designs and corresponding Java Classes in *ACRU2000.* Some of the major classes are described in Chapter 4.

After the conceptualisation and review of relevant UML diagrams and Java Classes in *ACRU2000*, the design of *ACRUSalinity* objects is implemented in the Rational Rose Software (Rational Software Corporation, 1995) with subsequent primary code generation using the same software. The generated primary code was further edited in JBuilder (Borland Software Corporation, 2001) to accomplish the required tasks for which the respective class was intended. The abovementioned stages are followed during the building process of all classes in *ACRUSalinity*. However, these steps were not followed step-wise since the significance of adding some new classes was emerging during the development process. Rather, the development process was iterative.

The development of *ACRUSalinity* is based mainly on the interaction between three objects, *viz.* Components, Data and Processes. Thus, the following topics will briefly introduce for these basic objects. This will be followed by details of each process representing the various real hydrosalinity processes, starting from salt input from wet atmospheric deposition and irrigation water to salt balance and transport in surface and subsurface components of the hydrological system, as well as the associated component and data objects.

5.1.1 Component objects

No new component (physical feature) was added to the hydrological system in *ACRUSalinity*. However, attributes that belong to a certain physical feature and hydrosalinity processes taking place in a particular physical feature are described from a reference of the component object to which these attributes belong and in which the processes take place. According to Clark *et al.* (2001), all component objects are part of the abstract *CComponent* Class and most of them represent either surface features such as *CIrrigatedArea*, *CImperviousArea* and *CDam* or, alternatively vertical layers such as the *CHorizon* subcomponents of the *CSoil* layer and *CGroundwater*.

5.1.2 Data objects

The role of data objects in *ACRU2000* is described in Chapter 4. Data objects in *ACRUSalinity* also serve a similar purpose. For example, simple data objects, such as *DRainfallSaltLoad*, *DTopsoilSalinity* and *DBaseflowSalinity* are used to store data pertaining, respectively, to the *CClimate*, *CSoil* and *CGroundwater* component objects. Some data objects still hold information about certain processes. For example, the *DSalinityOption* stores information on whether the hydrosalinity module is to be executed or not in a particular simulation, whereas the *DSaltFluxRecord* Object serves not only to store the salt load of a particular component, but also to conduct internal salt balance computations with the help of its parent classes. Such computations are automatically executed whenever salt transport occurs from one component to another. Like all the data objects in *ACRU2000,* data objects in

ACRUSalinity also extend to the *DData* Class. This module is comprised of a number of new data objects that describe the various hydrosalinity attributes (Appendix A).

5.1.3 Process objects

A number of process classes have been built to describe salt input, salt balance and movement taking place on the surface and subsurface components, including reservoirs and channels. All the process objects in *ACRUSalinity* extend ("are type of") the *PProcess* Class, which is the parent class of all process objects in *ACRU2000*. The processes taking place within a particular component are executed based on a predetermined order. However, the order of execution for hydrosalinity processes in different component objects follows the direction of water flow as determined by the hydrological modules of *ACRU2000*. According to Clark *et al.* (2001), on each day of simulation, the processes for the land segment on the edge (head water) of the simulated catchment are executed first, followed by land segments in progression towards the catchment exit. Thereafter, processes for each *CReach* type are executed, starting with reaches on the edge of the flow network and moving progressively downstream. Processes responsible for accomplishing the various hydrosalinity computations are grouped into six objects. These objects are briefly described below.

I. Initializing Salt Load

For the ease of module use by users, most inputs to the module are prepared in units that are readily available from physical measurements rather than in a way that can readily be used for internal computations. For example, in most cases, data for the initial TDS level of the soil solution is usually available as a concentration (mg/l) rather than a mass (mg). Therefore, the module is structured in such a way that it can accept inputs in readily available unit (mg/l). However, internal computations of the hydrosalinity processes involve salt load (mg). Hence, the main aim of this object is to set the initial salt load through computations based on the initial salt concentration and volumetric water content of the soil layers in irrigated and non-irrigated lands. This object also sets the initial salt load of reservoirs based on the initial reservoir water storage and its associated TDS concentration.

II. Salt Input

This object contains classes that are responsible for salt load input from rainfall and irrigation water to the topsoil horizon of irrigated and non-irrigated lands as well as to reservoirs. External salt input to non-irrigated lands and reservoirs has rainfall as its origin. However, irrigated lands receive additional salt input from irrigation water. Processes which undertake the salt input mechanism to irrigated land, non-irrigated land and a reservoir, respectively, are *PIrrigSaltInput*, *PLandSegSaltInput* and *PReservoirSaltInput*.

III. Surface Salt Movement

This object generally contains such process classes that describe the stormflow and runoff salinity as well as distribution of salt load from irrigated, non-irrigated and impervious areas and also reservoirs to an appropriate destination component. Some of the process classes contained in this object include *PRunoffSalinity*, *PIrrigAreaSaltMovement* and *PLandSegSaltMovement*.

IV. Subsurface Salt Movement

This object includes process classes that handle the salt balance and salt generation computations in subsurface components. Thus, processes in this object describe the movement of salts from the topsoil through subsoil to the groundwater store with subsequent salt generation taking place in each soil horizon and the groundwater store. This object also supports upward movement of salt load from the bottom horizon to stormflow through the overlying horizons, in the case of saturated upward flow. *PIrrigUpwardSaltTransport*, *PSubsurfaceSaltMovement* and *PSaltUptake* are some example classes from this object.

V. Reservoir Salt Budget

Processes included with in this object describe the reservoir salt budget with subsequent determination of the current reservoir storage salinity and salt concentration of the various outflow components, such as overflow and seepage. In the case of distributed hydrosalinity modelling, if the reservoir under consideration is situated at a particular sub-catchment's outlet, this object also carries out the transport of salt load associated with the various outflows from the reservoir to an appropriate sub-catchment. The main classes in this object are *PReservoirComponSalinity* and *PSaltStacking*.

VI. Channel Salt Movement

This object contains classes that describe the salt balance at the channel outlet of a particular sub-catchment. This object also performs the transfer of salt load from one sub-catchment to the relevant downstream sub-catchment, in the case of distributed hydrosalinity modelling. The main process class contained in this object is the *PCatchmentSalinity*.

5.2 Subsurface TDS Balance and Baseflow Salinity

Subsurface TDS balance and movement through the soil profile is based on the concepts used in the DISA model. However, the Lagranian salt lagging approach used in the DISA model to account for the varying sources of percolated water and its influence on TDS balance of the various layers is not employed in *ACRUSalinity*. This is not expected to have a significant impact on the subsurface TDS balance if the subsurface system is divided into only two layers, i.e. topsoil and subsoil, plus the groundwater store as vertical components of the subsurface system. Therefore, it is based on the assumption that each layer is deep enough to store more volume of water in comparison to the percolated water out of the layer for the day, and hence at a particular time step (day) the source of percolated water into a given layer is only from its immediate overlying layer.

5.2.1 Total evaporation and the soil water balance as conceptualised in the *ACRU* model

An increase in soil- and groundwater salinity is attributed mainly to the combined effects of hydrological and geochemical processes. Two of the main hydrological processes that influence the subsurface salt balance include precipitation and total evaporation. In most cases, the recharge of the soil horizons and groundwater store by precipitation has a dilution effect on subsurface water TDS concentration. On the other hand, the removal of water through evaporation and transpiration has a concentrating effect. Therefore, these processes coupled with the physiographic characteristics of the catchment, such as drainage of the soil, have a substantial effect on the subsurface water balance, and thereby on its salt balance.

5.2.1.1 Total evaporation

In the dryland routines of *ACRU,* total evaporation consists of evaporation from the plant tissue (transpiration) and soil water evaporation (Schulze, 1995c). Both transpiration and soil water evaporation occur at maximum rates when the plant is not under environmental stress. Maximum transpiration can be calculated either from LAI (leaf area index) values or water use coefficients (formerly termed crop coefficients). Similarly, maximum soil water evaporation is estimated either as a residual of the available energy remaining after estimating maximum transpiration, or from considerations of shading of the soil surface by aboveground biomass.

In *ACRU,* maximum transpiration is expressed as a function of reference potential evaporation and the fraction of total available transpiration. Maximum transpiration is allocated among the different soil horizons in proportion to the fraction of root mass density and degrees of colonisation of that specific horizon. However, when one of the horizons experiences a greater soil water deficiency than the other, the unstressed horizon contributes more to transpiration than computed by its proportion of root mass available for transpiration.

Actual transpiration may take place at its maximum rate or below. Plants may transpire below maximum rate under saturated or deficit soil water conditions. In *ACRU*, actual evaporation from the soil surface is calculated in two stages. In the first stage, when the soil is wet, evaporation from the soil surface is limited only by the energy which is available at the surface, and is thus equal to maximum soil water evaporation. Once the accumulated soil water evaporation exceeds the stage 1 upper limit, the stage 2 evaporative process starts, after which evaporation from the soil declines rapidly. Actual evaporation may be suppressed by surface cover such as mulch, litter or surface rocks. *ACRU* accounts for this effect by a linear relationship between the surface cover and soil water evaporation (AT7-10) (Schulze, 1995c).

Mathematical expressions that describe the preceding principles of total evaporation are given in the soil water budgeting and total evaporation as well as irrigation crop water demand chapters (Chapters 7 and 17) of the *ACRU* model documentation (Schulze, 1995c).

5.2.1.2 Soil water balance

The standard *ACRU* water budgeting routines for general use operate within a surface layer and two "active" soil horizons in which rooting development and hence soil water extraction, as well as soil water uptake and drainage can take place (*ACRU* Theory, pp AT5-4) (Schulze *et al.*, 1995b).

The soil water budgeting process in *ACRU* takes place in a sequence of steps. First, the soil water content at total porosity, drained upper limit and permanent wilting point need to be stipulated for each of the active soil horizons. After the soil water content of the topsoil horizon is adjusted by the addition of net rainfall, the soil water content of the topsoil horizon is re-assessed. If it exceeds the topsoil's drained upper limit, a proportion of the excess water drains into the subsoil horizon. Similarly, if the subsoil water content is above its drained upper limit (DUL), a fraction of the excess water drains below the root zone and to the groundwater store. Thereafter baseflow releases are calculated as the product of the previous day's groundwater store and a user specified baseflow recession coefficient, which depends on factors such as geology, catchment area and slope. On the other hand, if the drainage rate of the lower soil layer is very low (for example, because of a subsurface impervious layer), the soil water content may accumulate to a level exceeding its porosity. In such cases the water accumulates from the lower soil layer in an upward direction, filling first the subsoil horizon to porosity and thereafter contributing to the topsoil horizon from below. Should the topsoil's water content exceed porosity, excess water contributes directly to stormflow as saturated overland flow.

Unsaturated soil water redistribution can take place in the model as a result of differences between soil water conditions of the respective horizons. This slow movement of water will occur from the top soil horizon, when the soil water content is below its drained upper limit (DUL), to the subsoil horizon if the topsoil horizon is relatively wetter than the subsoil horizon. Unsaturated redistribution depends on the soil water gradient, the head of water and soil texture. Upward soil water redistribution in *ACRU* mimics capillary movement, and takes place when the subsoil horizon contains a higher relative soil water fraction compared to that of the topsoil (*ACRU* Theory, pp AT7-19) (Schulze, 1995c).

Various options are provided by *ACRU* decision support system to estimate soil water content at permanent wilting point and drained upper limit, depending on the level of available soil information (*ACRU* Theory, Chapter 5) (Schulze *et al*., 1995b).

5.2.2 Rainfall and irrigation water salt input

The source of dissolved solutes in the soil solution, other than the primary source, i.e. due to *in situ* weathering processes, is assumed to be from solutes added along with rainfall (wet atmospheric deposition) and irrigation water. In irrigated areas, solute input includes both that from applied irrigation water and that from rainfall. The average TDS concentration of rainfall and irrigation waters are input to the model and are stored in the *DRainfallSalinity* and *DIrrigationWaterSalinity* data objects for subsequent computations.

In most cases, it is difficult to obtain a time series of rainfall salinity. Therefore, salt concentration of rain water is assumed to have a constant value at a specific location and can be taken as the average observed value at the site. However, time series irrigation water salinity is available for most irrigated areas and is characterised by seasonal variations in some areas. Therefore, irrigation water salinity is input to the model on monthly basis.

These processes of salt input to the soil are carried out in two similar process classes functioning on irrigated and non-irrigated lands. The *PLandSegSaltInput* Process carries out the daily input of salt load associated with rain falling on non-irrigated lands. On the other hand, the *PIrrigSaltInput* Process determines the daily salt load from rain falling on irrigated areas and applied irrigation water, with subsequent addition of this salt load to the topsoil horizon. The quantity of salt load added from a rainfall source is described as the product of effective rainfall volume and rainfall salinity (Equation 5.1). Similarly, the salt load associated with irrigation water is determined as the product of the volume of applied irrigation water and its average TDS concentration (Equation 5.2). The flow diagram in Figure 5.1 represents the main steps included in the *PIrrigSaltInput* Process.

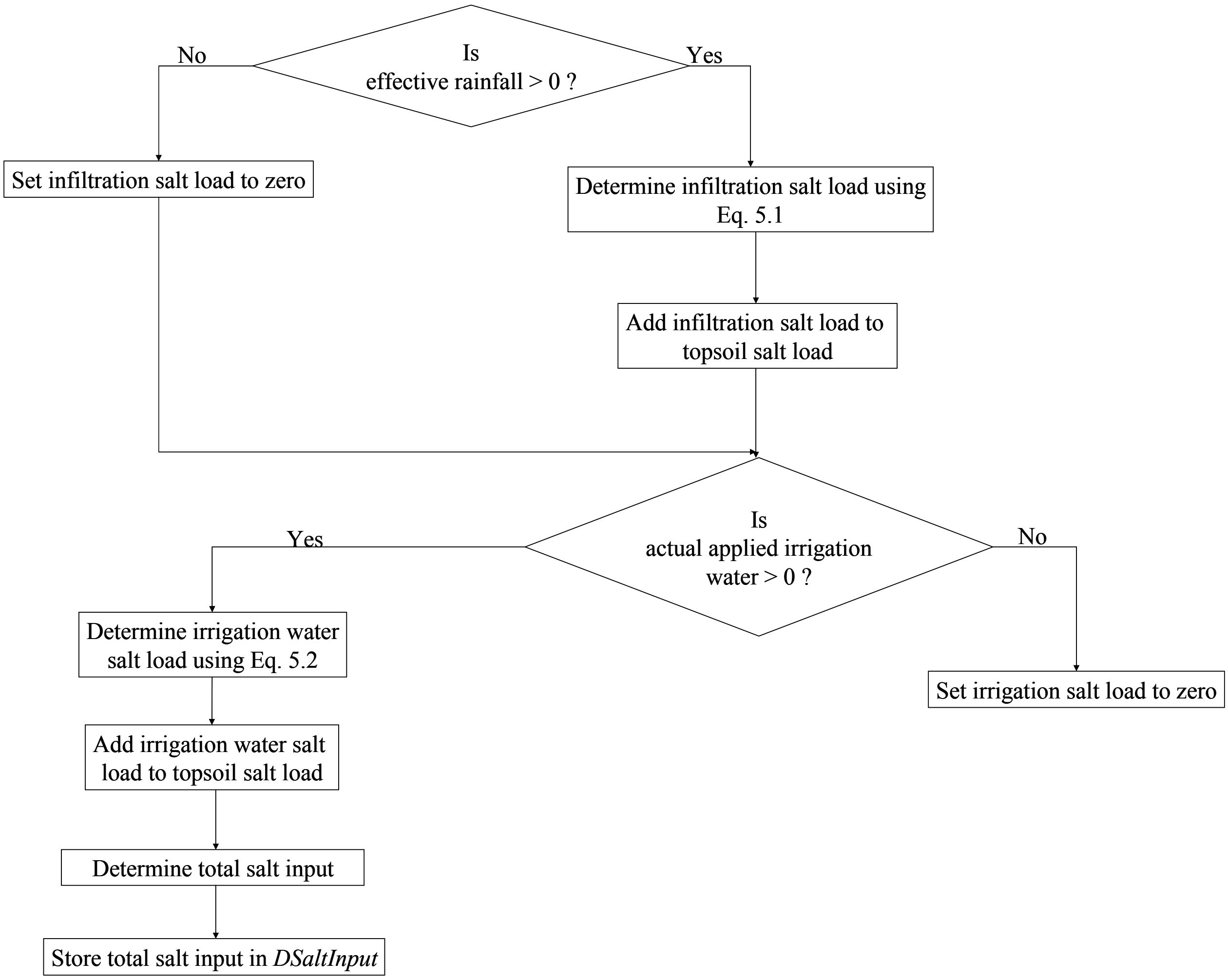

Figure 5.1 Flow diagram of salt input mechanism to irrigated lands as accounted in the *PIrrigSaltInput* Process Object

$$SL_{er} = ER * C_r \tag{5.1}$$

$$SL_{aiw} = IW * C_{iw} \tag{5.2}$$

where
SL_{er} = salt load input to topsoil associated with rainfall (mg)
ER = volume of effective rainfall (l)
C_r = rainfall salinity (mg/l)
SL_{aiw} = salt load input to topsoil associated with irrigation water (mg)
IW = volume of irrigation water (l) and
C_{iw} = irrigation water salinity (mg/l).

In *PIrrigSaltInput*, effective rainfall refers to the volume of rainfall infiltrated into the topsoil horizon on a particular day and is expressed in litres. Similarly, actual applied irrigation water refers to the volume of irrigation water infiltrated into the soil, i.e. the total applied irrigation water excluding the various irrigation losses. This process assumes that irrigation water is applied only to the topsoil horizon and hence its direct contribution is only to the topsoil TDS balance. Therefore, the daily calculated salt load both for irrigated and non-irrigated lands is added to salt load of the topsoil horizon. The *PIrrigSaltInput* and *PLandSegSaltInput* processes are similar in structure. Figure 5.2 shows the various data and component objects associated with the *PIrrigSaltInput* Process Object and the interaction between these objects. Notations representing the relationship types in the figure are described in Chapter 4. For a definition of the class names in the diagram see Appendices A, B and C.

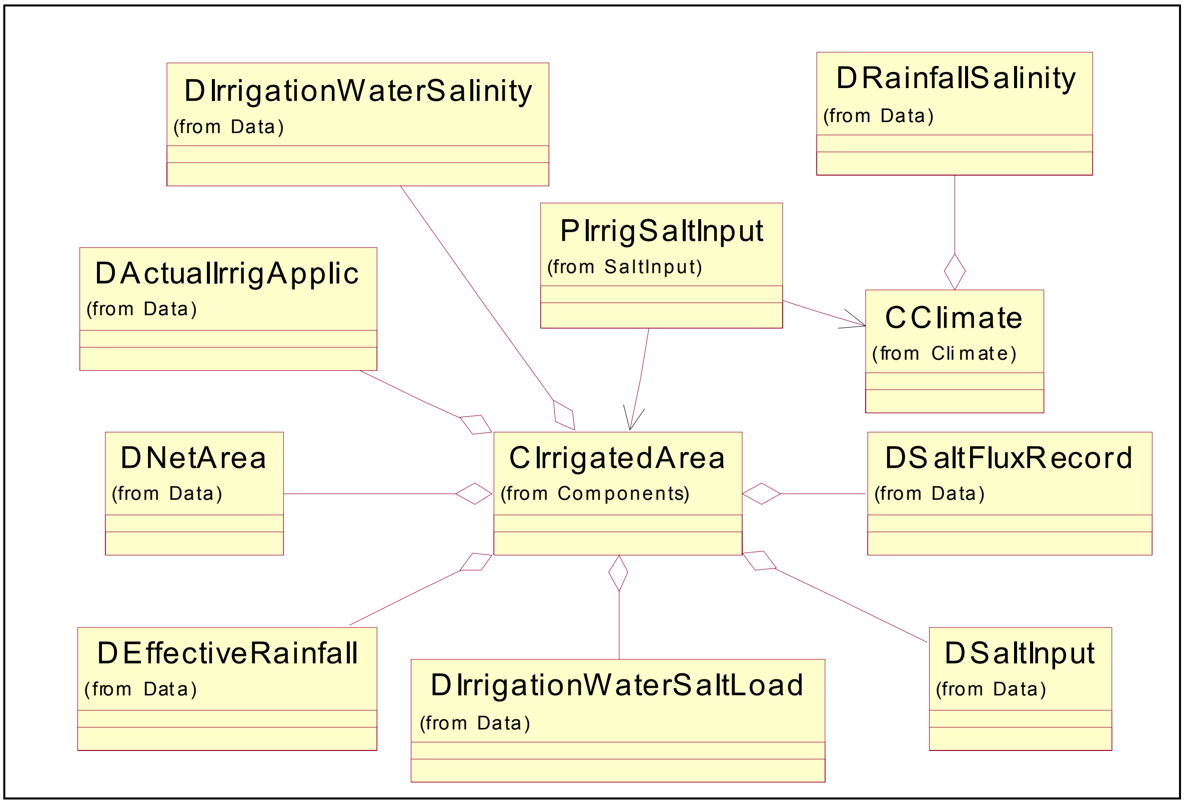

Figure 5.2 Class diagram of *PIrrigSaltInput* Process and associated data and component objects

5.2.3 Subsurface salt movement

Subsurface salt movement occurs in both irrigated and non-irrigated lands. It can be either downward or upward, depending on the direction of soil moisture movement. Downward salt movement occurs as a result of percolation of water from the topsoil to underlying horizons and the groundwater store. On the other hand, upward salt movement is associated with saturated upward flow of water from a bottom horizon to the overlying layer under conditions of poor drainage.

5.2.3.1 Downward subsurface salt movement

The downward salt movement from any layer in the soil profile occurs when the saturated downward flow of water from that layer is greater than zero. In *ACRU,* saturated downward flow takes place when the soil moisture store exceeds drained upper limit. Hence, downward salt movement can also take place only when the drained upper limit is exceeded. *ACRU* includes an option for unsaturated water movement in the soil profile, i.e. flow from topsoil to subsoil horizon or *vice versa* when the soil water content is below DUL. However, at present *ACRUSalinity* is not linked to this optional process. Therefore, this option needs to be switched off when conducting hydrosalinity simulations in order to avoid salt imbalances.

The subsurface system is composed of vertical layers. Thus, the algorithms that perform subsurface salt balance and movement computations are written assuming a multi-layered soil profile. In non-irrigated lands, for example, downward salt movement takes place from the topsoil through the subsoil to the groundwater store. Thus, in these conditions *ACRU* considers three subsurface components (two soil horizons and the groundwater store). However, the algorithm that carries out this salt balance process and in most other subsurface TDS balance processes are written in a way that they can be used for multi-layered soils with more than three stores.

The downward subsurface salt movement and salt balance computations are accomplished by the *PSubsurfaceSaltTra* and *PIrrigSubsurfaceSaltTransport* processes on non-irrigated and irrigated lands respectively. The component, process and data objects associated with the *PSubsurfaceSaltTra* Process are depicted in Figure 5.3. Definitions for the data objects are given in Appendices A and B. Similarly definitions for the component objects are presented in Appendix-C.

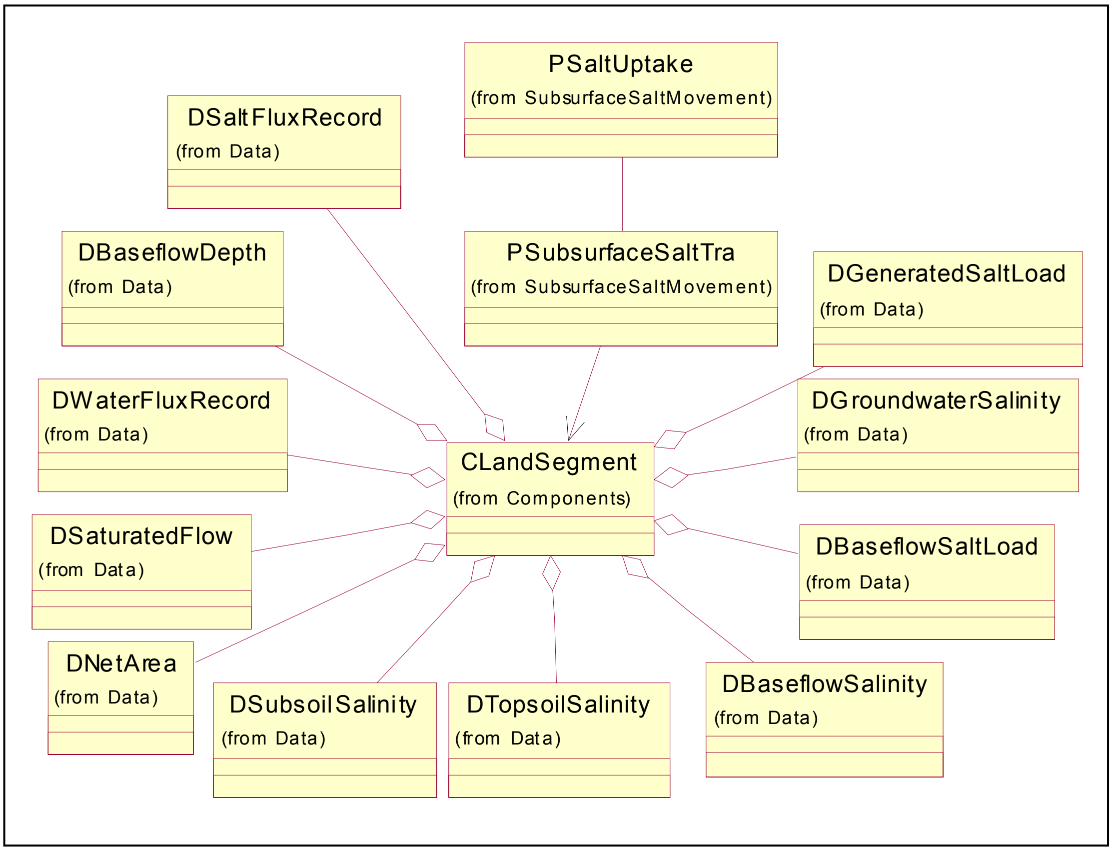

Figure 5.3 Class diagram of *PSubsurfaceSaltTra* Process and its associated component and data objects

The *PSubsurfaceSaltTra* Process determines the subsurface TDS balance in non-irrigated lands. In this process salt is transported from the topsoil to an underlying horizon and finally to the groundwater store depending on the volume of percolating water and its salinity. It also determines the TDS concentration of each horizon, salt load associated with percolation water and baseflow salt concentration.

The *ACRUSalinity* module and other water quality related modules of *ACRU2000*, such as sediment yield and nutrient simulations (Nitrogen and Phosphorous), are executed after the relevant hydrological processes are executed and the associated data objects are set for the day. Therefore, the salt concentration and salt load in a particular horizon are computed after percolation of water to an underlying layer has taken place for the day. Hence, taking into account the effect of this phenomenon the salinity of each horizon, before salt generation, is estimated using the following equation:

$$C_i = \frac{SL_i}{SW_i + PW_i} \tag{5.3}$$

where C_i = salt concentration of the i-th horizon before salt generation (mg/l)

SL_i = current salt load of the i-th horizon before salt generation (mg)

SW_i = volumetric soil water content of the i-th horizon after percolation has taken place out of the horizon (l) and

PW_i = volume of percolated water out of the i-th horizon (l)

The salt load of the subsurface components (layers) is replenished from internal and external sources. The salt load of the topsoil horizon is replenished from rainfall salt input, whereas, in the case of subsoil and groundwater store, it is replenished by the salt load added from an overlying layer along with the percolating water. The internal source of salt load to a particular layer, or the groundwater store, is through salt generation within the layer or the groundwater store. The increase in salt concentration of each horizon due to salt generation is determined in a separate process (*PSaltUptake*) and is described in Section 5.2.4. However, the *PSaltUptake* Process only updates the salt concentration of a given layer according to first order rate kinetics. Thus, once the *PSubSurfaceSaltTra* Process has received the updated salinity level as determined by the *PSaltUptake* Process, the salt load after the update of the horizon's salinity is calculated using Equation 5.4. The quantity of salt added to the particular layer due to salt generation is then determined as the difference of the salt load after and before the update of that horizon's salinity has taken place (Equation 5.5).

$$SL_{upd\_i} = C_{upd\_i} * (SW_i + PW_i) \tag{5.4}$$

$$SL_{gen\_i} = SL_{upd\_i} - SL_i \tag{5.5}$$

where SL_{upd_i} = salt load in the i-th horizon after salt generation (mg)

C_{upd_i} = updated horizon salinity (mg/l) and

SL_{gen_i} = salt load generated for the day in the i-th horizon (mg).

The water percolating on a daily basis from each horizon has the same salt concentration as the particular layer from which percolation took place. Thus, this process assumes that the volume of water entering each layer for the day originates only from its immediate overlying layer. This assumption seems to hold true in *ACRU*, since the soil profile is commonly divided only into topsoil and subsoil horizons. In this case, the layers are deep enough such that at a daily time step the amount of percolated water is likely to be less than the storage of the layer immediately above the current layer. The salt load associated with the percolation water out of the i-th horizon (SL_{p_i}) is described by Equation 5.6 and is transported to the underlying layer or groundwater store (if the current layer is the bottom horizon) before any salt balance computation commences for the underlying layer.

$$SL_{p\_i} = C_{upd\_i} * PW_i \tag{5.6}$$

The groundwater salt balance and baseflow salinity are determined after the salt balance of the soil horizons has been set for the day. Within the soil profile, salt load associated with percolation water is added to an underlying horizon. However, if the layer under consideration is the bottom horizon, the salt load of the percolated water is transported to the groundwater store and replenishes the groundwater salt load. If, on a particular day, the groundwater store is not empty, its daily salinity and salt load are determined in a similar way to that of soil horizons. In this case, however, the salt load leaving the groundwater store is a function of the baseflow volume and groundwater salinity. The daily groundwater salt concentration before salt generation is computed based on the current groundwater volume and salt load, as well as the volume of water released from the groundwater store as baseflow. This is expressed by the following equation:

$$C_{gw} = \frac{SL_{gw}}{GW + BF} \tag{5.7}$$

where C_{gw} = salt concentration of the groundwater store before salt generation (mg/l)

SL_{gw} = groundwater store salt load before salt generation (mg)

GW = volumetric groundwater content after baseflow release (l) and

BF = volume of baseflow release for the day (l).

Besides the salt load source associated with percolation water from the bottom soil horizon, the salt load of the groundwater store is also replenished by the salt generated from within the groundwater system as a result of the different weathering processes acting upon the soil and geological formations. This salt generation process in the groundwater store is also performed in conjunction with the *PSaltUptake* Class in a similar way as is done for soil horizons. The groundwater store TDS concentration is updated by the *PSaltUptake* Process (section 5.2.4). The groundwater salt load after update of the salt concentration is calculated in *PSubsurfaceSaltTra* using Equation 5.8 and the generated salt load is calculated as the difference of groundwater salt load before and after the salt concentration is updated (Equation 5.9).

$$SL_{upd\_gw} = C_{upd\_gw} * (GW + BF) \tag{5.8}$$

$$SL_{gen\_gw} = SL_{upd\_gw} - SL_{gw} \tag{5.9}$$

where SL_{upd_gw} = salt load of groundwater store after salt generation (mg)
C_{upd_gw} = updated groundwater salinity (after salt generation) (mg/l) and
SL_{gen_gw} = salt load generated for the day in groundwater store (mg).

Baseflow volume released from the groundwater store is assumed to have the same salinity level as the updated groundwater TDS concentration for the day. The associated salt load for baseflow release is then calculated as:

$$SL_{bf} = BF * C_{bf} \tag{5.10}$$

where SL_{bf} = salt load associated with baseflow release (mg) and
C_{bf} = baseflow salt concentration (mg/l).

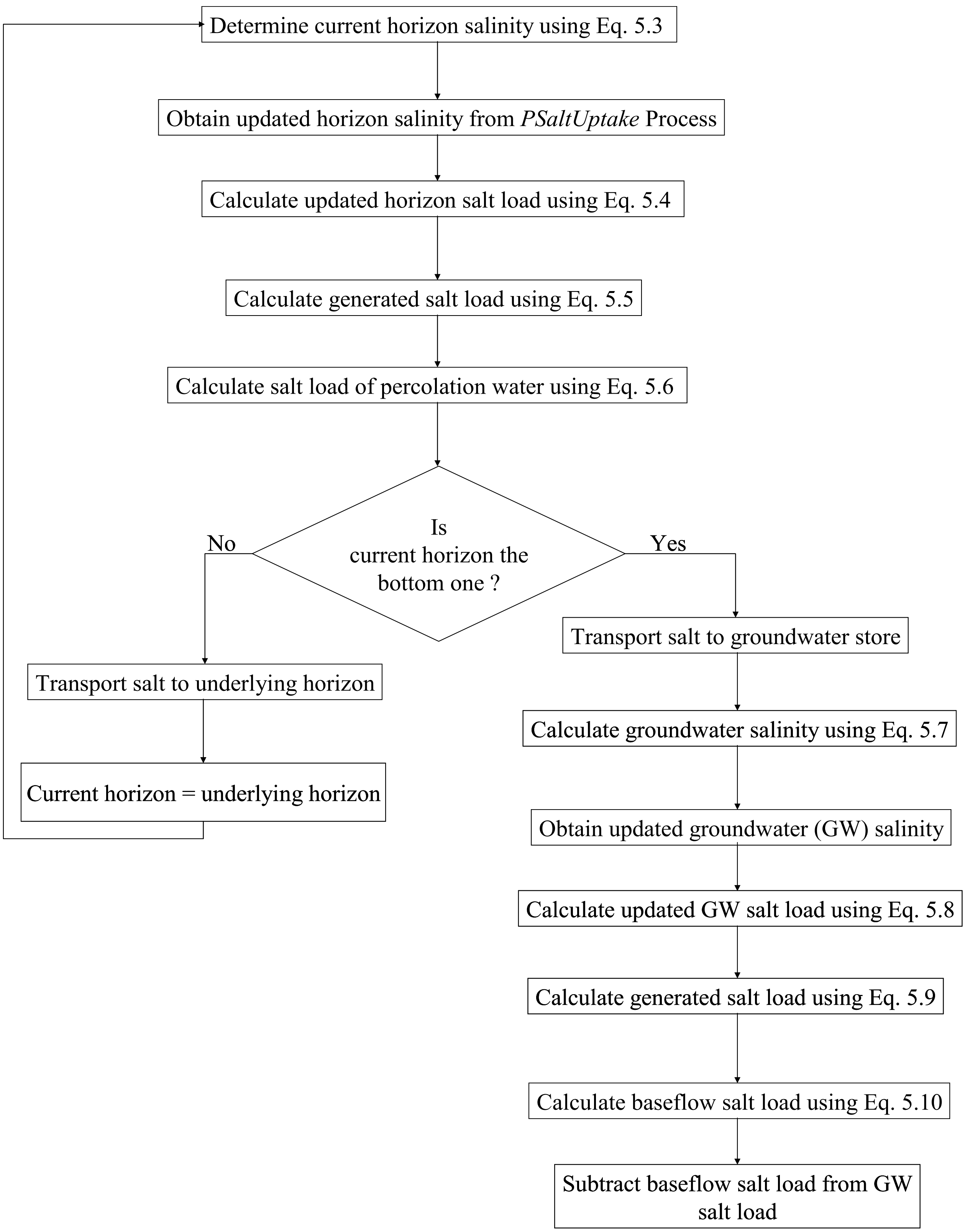

Figure 5.4 Flow diagram of subsurface salt movement in non-irrigated lands

The *PIrrigSubsurfSaltTransport* Process serves a similar purpose as that of *PSubsurfaceSaltTra*. This process, however, handles subsurface TDS balance and movement in irrigated lands. Generally it has similar algorithms to that of *PSubsurfaceSaltTra*, but in this case, only a single horizon and the groundwater store are considered as *ACRU* considers only these subsurface components in irrigated lands. In *ACRU* the months in which irrigation take place are specified by the user. Thus, this algorithm is invoked only for months in which irrigation is taking place. Component and data objects associated with this process are depicted in the following figure and definition of each class is given in Appendices A, B and C.

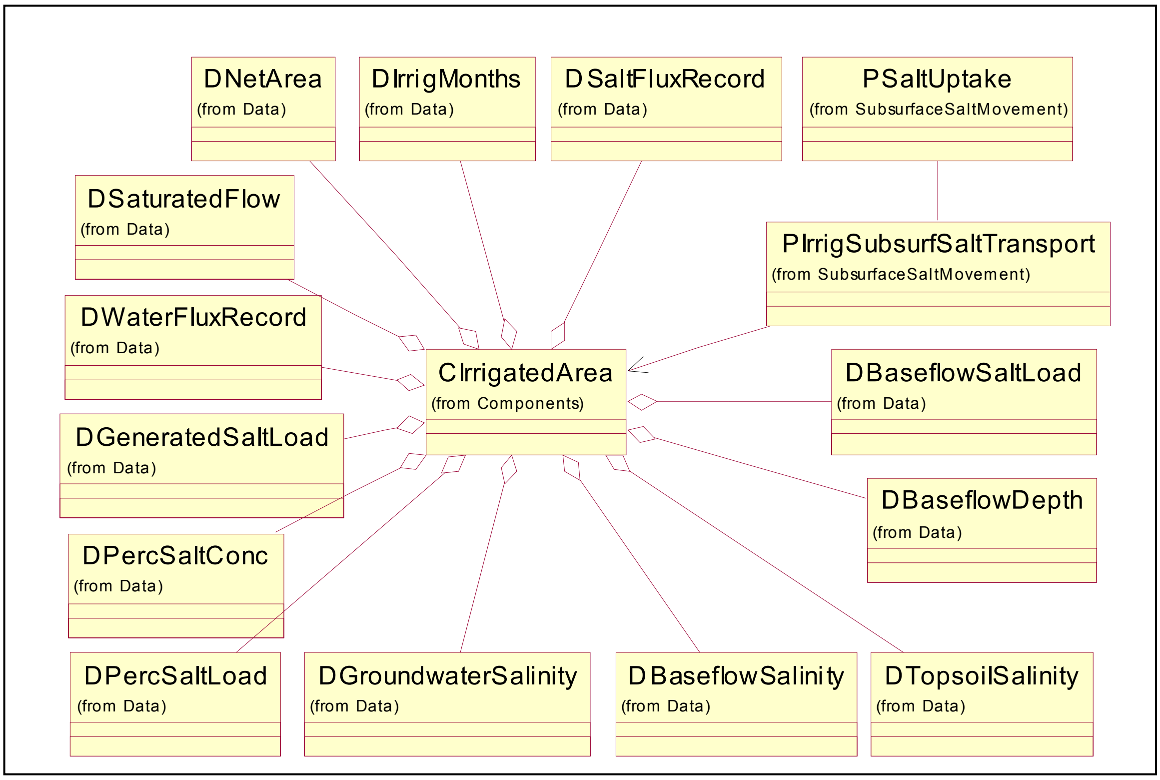

Figure 5.5 Class diagram of *PIrrigSubsurfSaltTransport* and its associated data and component objects

5.2.3.2 Upward subsurface salt movement

Upward salt movement through the soil profile and its influence on surface and subsurface salt balance is determined by the *PUpwardSaltTransport* Process. In this process, salt load moves from the bottom horizon through the overlying horizons to quickflow. The upward movement of salt is dependent on the moisture status and drainage of a particular layer. This process accounts upward subsurface salt movement only under a saturated condition. Hence, upward salt movement under this process occurs only if the rate of water recharge to a layer exceeds the rate of water loss from that particular layer. Information on the quantity of upward moving water is retrieved from the *DSatUpwardFlow* Data Object whose value is determined and assigned in *PSatUpwardFlow*. This data object is created in *ACRUSalinity* for hydrosalinity computations. Data and component objects associated with this process are shown below. Definitions for the data and component objects are given in Appendices A, B and C.

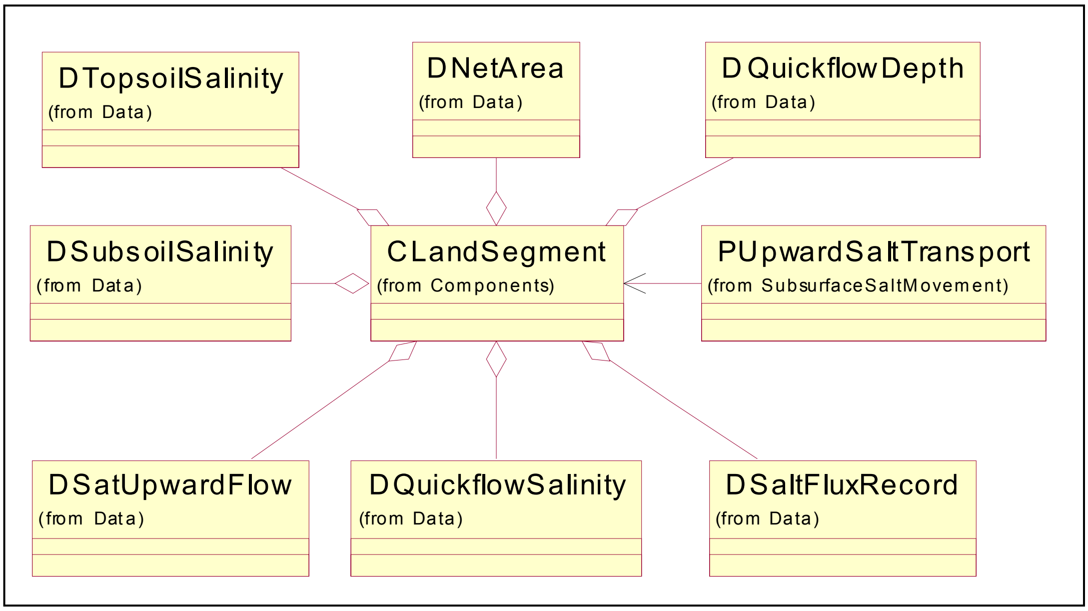

Figure 5.6 Class diagram of *PUpwardSaltTransport* Process and associated data and component objects

If, on any particular day, the upward movement of water from a given horizon is greater than zero, then the salt load entering an overlaying layer is expressed as the product of the volume of water entering the layer and the current salt concentration of the layer from which this volume of water commences. The upward flow of water is expressed in litre (l) and the salt concentration value in mg/l. Thus, the salt load entering the layer is expressed in milligram (mg). This is so in the case of salt movement being within the soil profile. If the origin of the upward moving salt is from the topsoil horizon, the salt load is added to the quickflow salt load. The TDS concentration of quickflow is subsequently updated under this process using Equation 5.11 in order to account for the effect of the salt flux from the topsoil associated with saturated upward flow. The concept of quickflow in *ACRU* model and determination of its initial salinity and salt load are discussed in Section 5.3.

$$C_{qf} = \frac{SL_{qf} + SL_{upf}}{QF} \tag{5.11}$$

where C_{qf} = the updated quickflow salinity (mg/l)

SL_{qf} = salt load of the quickflow before upward salt flux from the topsoil (mg)

SL_{upf} = upward salt flux from topsoil to quickflow (mg) and

QF = quickflow volume (l).

The following figure shows a flow diagram of an upward salt movement in non-irrigated lands as accounted in *ACRUSalinity*.

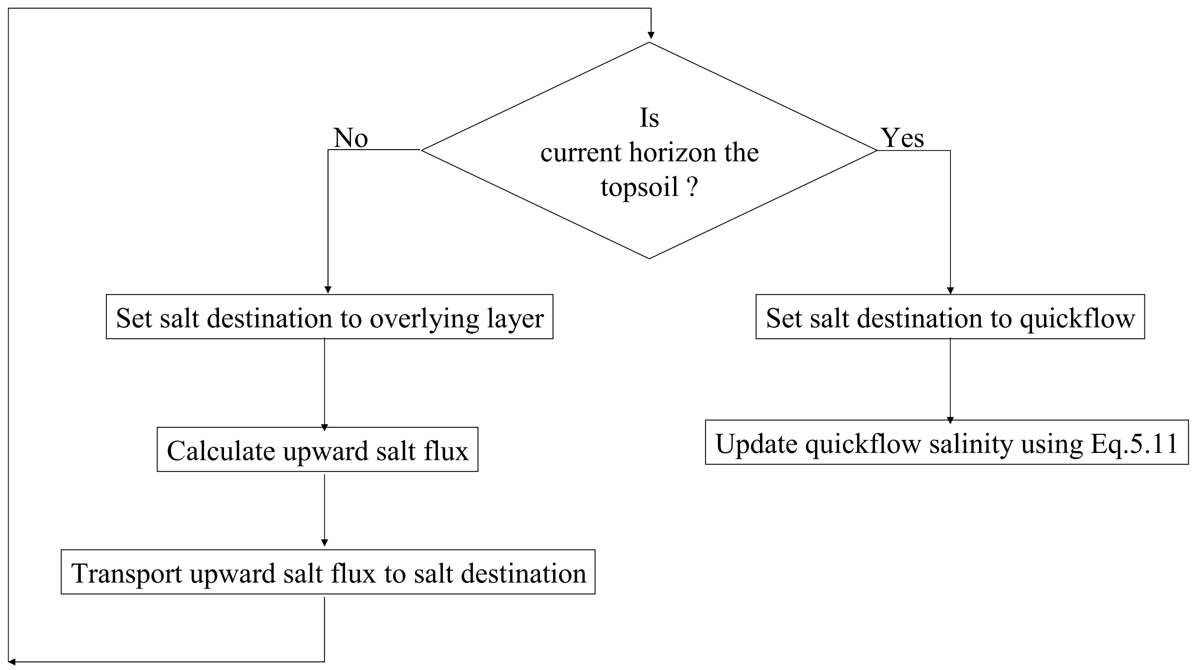

Figure 5.7 Flow diagram of upward salt movement in non-irrigated lands as represented in the *PUpwardSaltTransport* Process

The *PIrrigUpwardSaltTransport* Process also serves a similar task in irrigated lands. However, this process determines salt movement only from the topsoil to quickflow, with subsequent updating of the quickflow salinity and the salt load. This is so because of the presence of only a single layer as conceptualised in the *ACRU* model in the case of irrigated lands.

5.2.4 Salt generation

The process of salt generation involves complex weathering and soil-solute interaction mechanisms, which in turn are influenced by hydrological, climatic, geochemical and anthropogenic factors. Thus, among other factors, an adequate description of this process includes the detailed chemical reactions taking place at individual element level. This requires knowledge on such detailed processes and extensive data for each solute species, which is beyond the scope of this research. Hence, a simplified modelling approach of the salt generation mechanism is adopted in *ACRUSalinity*.

As mentioned in Section 5.3.2, computations to update TDS concentration of each horizon and the groundwater store to account for the increased salt concentration due to salt generation, take place in a separate process class. This allows for re-use of the same algorithm to determine the updated salt concentration of each soil horizon and the groundwater store both for irrigated and non-irrigated lands. Moreover, placing this process into a separate class minimises the complexity of processes performing subsurface salt balance and movement. Furthermore, any changes required for the future in the expressions describing the increased salt concentration as a result of salt generation processes can be made in this process only with little or no change to any of the other processes.

The salt generation process and subsequent update in salt concentration of each layer is carried out in the *PSaltUptake* Process. This process receives information from the calling class, which can either be the *PSubsurfaceSaltTra* or *PIrrigSubsurfSaltTransport*. The information includes identity of the layer in which salt generation is to take place, salt concentration of the layer before salt generation and area of the irrigated or non-irrigated land in which salt generation is to take place. This process then updates the salt concentration of the current layer based on the above and other relevant information according to first order rate kinetics as proposed by Ferguson *et al.* (1994).

As described in chapter 3, the first order rate kinetics equation assumes that the rate of increase over time in the concentration of a solute is proportional to how far the current concentration falls short of its equilibrium value. Equation 5.12 describes an initially rapid, but progressively slower, salt generation such that the concentration approaches the equilibrium value asymptotically. In the absence of any dilution by rainfall, irrigation water or percolation and "evapoconcentration", the increase in subsurface TDS concentration with time from its initial value (C_i) to the saturation (C_{sat}) value due to salt generation is represented as in Figure 5.8.

$$C_{upd\_i} = C_i + (C_{sat} - C_i)[1 - \exp(-k)] \tag{5.12}$$

where C_{upd_i} = updated salt concentration of the i-th horizon or groundwater store (mg/l)

C_i = salt concentration of the i-th horizon or groundwater store before salt generation (mg/l)

C_{sat} = the equilibrium value (mg/l) and

k = rate constant.

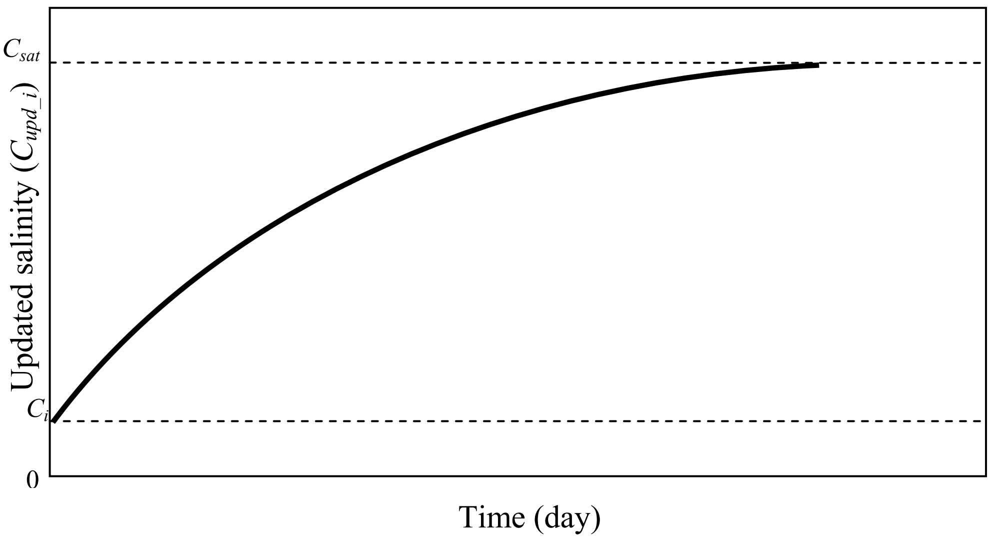

Figure 5.8 An increase in subsurface TDS concentration with time based on the first order rate kinetics

The first order rate kinetics equation is adopted in *ACRUSalinity* with some assumptions. Originally, this equation was proposed for use in estimating solute enrichment of the soil solution due to soil water uptake of individual solute species. However, in this case the equation is used for estimating the increased total dissolved solutes (TDS) value. This is based on the assumption that the quantity of total dissolved solutes, which is the salinity of a given layer, is the sum total of the major individual solute species in the soil solution. Thus, the increase in total dissolved solute concentration follows a trend similar to that of the individual solute species and hence it can be described by a similar equation. The time parameter (t) in the original equation is omitted in Equation 5.12, since in this case the time step between successive salt generation computations is fixed to a single day. Hence, its value is 1.

The equilibrium value and the rate constant for each horizon and groundwater are inputs to the model by the user and are stored in the *DSaltSat* and *DUptakeRateConstant* data objects. The equilibrium value refers to the maximum salt concentration of the layer at which saturation level is reached. The rate constant (k) controls the rate of salt generation from each horizon and the groundwater store. Although the rate constant is expected to vary between solute species, for this purpose its value is assumed to be similar for all solute species and is represented by a single averaged value.

Once the updated salt concentration is determined by this process, the generated salt load is calculated in *PIrrigSubsurfSaltTransport* and *PSubsurfaceSaltTra* for irrigated and non-irrigated lands respectively. It is calculated as the difference between the salt load of a particular layer before and after update of the salt concentration has taken place. The generated salt load is stored in *DGeneratedSaltLoad* Data Object.

5.2.5 Effect of total evaporation on subsurface TDS balance

Different studies have shown the presence of some chemical constituents in irrigation and rainwater. The loss of this water through evaporation from the soil and vegetation tend to increase salinity. Dissolved solids are added to agricultural land by way of irrigation and rainwater. However, neither surface evaporation nor absorption by plants appreciably reduces the amount of these salts added to the soil (Kay, 1986). Rather, the continuous upward movement of water from a subsurface system results in salt accumulation near the soil surface as water is lost by evaporation (Hoffman *et al.*, 1990). Thus *ACRUSalinity* also attempts to take into account the effect of total evaporation based on this concept. On each day of simulation, evaporation taking place both through the plant and from the soil surface is assumed to leave the salts behind, thereby resulting in an increased TDS concentration of the soil solution from which water is removed through this process. The removal of evaporated water from the soil is undertaken by the soil water budgeting modules of *ACRU*.

5.3 Determination of Surface Flow Salt Balance

The interactions between hydrological and geochemical processes are fundamental in determining the stream water chemistry of many catchments (Chen *et al.*, 2002). Hence, any process that affects runoff generation is likely to influence runoff salinity. As described in the following section, stormflow generation in *ACRU* and other functional models is often expressed by adaptation of the SCS equation. Similarly, baseflow release can be estimated by one dimensional Dupuit approximations (as used in DISA model), or as a product of groundwater store and baseflow response coefficient (as used in *ACRU* model). However, in most cases the salinity level of stormflow is assumed to be equal to rainfall or irrigation water salinity.

5.3.1 Stormflow generation mechanism in *ACRU*

In *ACRU* the term stormflow refers to the flow generated from a rainfall event. Thus the applied irrigation water does not contribute to stormflow generation. Part of the stormflow generated on non-irrigated lands leaves the land on the same day and hence is referred to as quickflow. Whereas, the remaining quantity is delayed for release on subsequent days and is called delayed stormflow.

5.3.1.1 Stormflow generation

The SCS equation is based on the principle that runoff potential is inversely related to the soil's relative wetness. The stormflow depth, Q (mm), is expressed as (Schulze *et al.*, 1992):

$$Q = \frac{(Pg - Ia)^2}{Pg - Ia + S} \qquad \text{for } Pg > Ia \tag{5.13}$$

where Pg = gross daily precipitation amount (mm)
Ia = initial abstractions (mm) before stormflow commences and
S = potential maximum retention of the soil (mm).

The initial abstraction, Ia consists mainly of interception, initial infiltration and depression storages. In the *ACRU* model (Schulze, 1995a), however, interception is abstracted separately and before the commencement of potential runoff producing rainfall. Initial abstraction is computed as a product of a variable coefficient and potential maximum retention (soil moisture deficit), i.e. $Ia = cS$. The regression coefficient c is referred to as the coefficient of initial abstraction and it depends on vegetation, site and management characteristics. The default value of the coefficient in *ACRU* is 0.2. However, it can be increased for conditions immediately after ploughing when surface roughness is high, or under forested conditions and reduced under conditions of soil compaction or convective rainfall.

The potential maximum retention (S) is taken as the difference between the soil water content at porosity and that held by a soil column of user specified critical depth prior to a rainfall event. The critical depth is a threshold depth for which the soil water deficit is calculated for stormflow generation. In *ACRU* the critical soil depth may vary and is used to account for different dominant runoff producing mechanisms prevailing in different climates, rainfall intensities, vegetation conditions and for different soil properties, as shown by the following diagram.

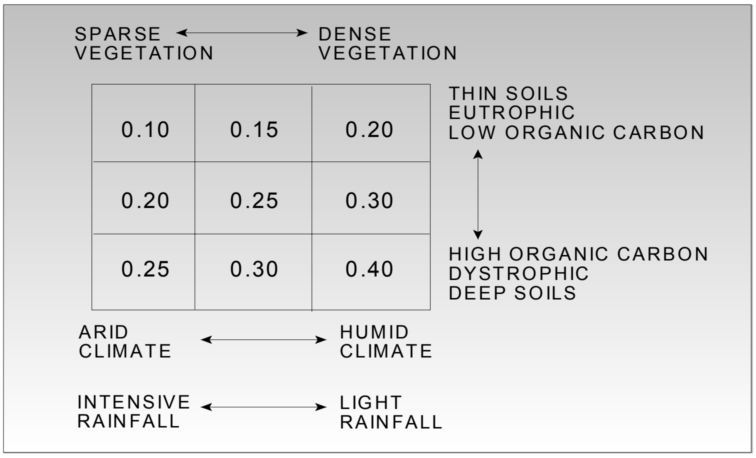

Figure 5.9 Suggested values of critical stormflow response soil depth (m) according to climatic, vegetation and soil characteristics (Schulze, 1995b)

5.3.1.2 The concept of delayed stormflow

In the *ACRU* model, the total stormflow generated from irrigated lands is assumed to leave the land on the same day. On the other hand, in non-irrigated areas a fraction of the generated flow is assumed to flow for subsequent days. According to Schulze (1995b), the generated total stormflow response from non-irrigated lands may be rapid or slow. Soils with a high interflow potential would respond rapidly, as would small and/or steep and/or urbanised catchments when compared with relatively larger ones and/or catchments with gentle gradients or dense land cover, where infiltration is high and lateral flow occurs more slowly. For this reason a stormflow response coefficient is included in *ACRU* model, which controls the "lag" of the delayed (interflow) component of stormflow on the day of the event and determines what fraction of stormflow generated from an event is same-day runoff. In this document this is referred to as "actual quickflow". The remaining stormflow is "retained" to the next day, when again the fraction of the remaining is discharged into the stream, giving rise to an exponentially declining recession limb of a hydrograph. The sum of the "actual quickflow" and the fraction of the delayed stormflow discharged in a particular day is referred to in this document as quickflow. Furthermore, it is assumed in this research that the delayed stormflow does not include any interflow from subsurface components and thus it is having the same salinity level as the rain falling over the area.

5.3.2 Stormflow and quickflow salinity

The origin of dissolved solutes in surface water is assumed to be mainly from wet atmospheric deposition, i.e. the salt load associated with the rain falling on the area, and from the applied irrigation water. Salt input from these sources to the topsoil and its impact on the subsurface salt balance are described in Section 5.2.2. The present section describes the influence of salt input from these sources on the surface flow TDS balance in general, and on stormflow and quickflow salinity in particular.

The source of TDS in quickflow is expected to be both from rainfall and applied irrigation water as well as enrichment from the soil surface. However, because of the difficulty involved in estimating the quantity of solutes diffused to stormflow and as a result of the insignificant effect on stormflow TDS balance of such contributions, most models tend to ignore the diffused salt load contribution from the soil surface. According to Rhoades *et al.* (1997), prevalent "textbook" logic would lead to the conclusion that salt pickup via stormflow should be negligible because the "leaching edge" of the water that flows over the soil is thought to infiltrate into the soil and to "carry" the readily soluble salt with it. The salt in the soil is not expected to diffuse upwards significantly when the water is percolating downwards. With this prevalent view of the transport processes, one would not expect to find a significant increase in the salinity of stormflow compared to that of the applied water, other than that which might be derived from the dissolution of suspended sediment gained through erosion. Mironenko and Pachepsky (1998) further suggest that ignoring vertical transport of the chemicals in surface water is obviously a viable approach for a thin water layer in rainfall-induced stormflow. Hence, the stormflow salinity is assumed to have the same value as the rainfall's average salt concentration on a site. However, quickflow salinity depends not only on the rainfall's salinity, but also on the salt concentration of delayed stormflow as well. The influence of delayed stormflow is described in the next section. The assumption that stormflow salinity is having the same TDS concentration as the rainfall average salinity may not hold true in areas characterised with significant salt crusts. In such areas the stormflow salinity may rise sharply with the rising limb of a hydrograph especially on the first flush of rainfall following a prolonged dry spell.

Determination of the salt balance and movement in stormflow generated from irrigated areas follows a similar approach to that from non-irrigated lands. However, in irrigated areas stormflow generated on a particular day is assumed to leave the area on the same day. Thus, unlike in non-irrigated lands, quickflow TDS concentration for irrigated lands is not influenced by TDS concentration of stormflow from previous days. In *ACRU2000,* stormflow generation from irrigated land is based on the assumption that it occurs only during a rainfall event and irrigation water *per se* does not make a direct contribution to stormflow generation. Therefore, in *ACRUSalinity* the stormflow generated on a particular day is assumed to have the same salinity as the average rainfall TDS concentration for the area. Salt concentration and salt load associated with the stormflow from irrigated lands is determined by the *PIrrigStormflowSalinity* Process.

5.3.3 The effect of delayed stormflow on TDS balance determination

The concept of delayed stormflow in *ACRU* complicates determination of TDS concentrations and salt loads associated with quickflow. Thus, to accommodate this concept an algorithm is included in *PStormflowSalinity*, where quickflow salinity is determined through simple mixing of the fraction of delayed stormflow and the fraction of generated stormflow leaving the area on the same day. Mathematically this is expressed as:

$$C_{qf} = \frac{(QF_a * C_{sf}) + (SF_d * C_{dsf})}{QF_a + SF_d} \tag{5.14}$$

where C_{qf} = quickflow salinity (mg/l)

QF_a = actual quickflow, i.e. fraction of the stormflow leaving the land on the same day (l)

C_{sf} = stormflow salinity (C_{sf} = rainfall average salinity)(mg/l)

SF_d = fraction of delayed stormflow contributing to quickflow (l) and

C_{dsf} = salt concentration of delayed stormflow (mg/l).

Once the quickflow TDS concentration is determined, the salt load associated with quickflow is estimated by the following equation:

$$SL_{qf} = QF * C_{qf} \tag{5.15}$$

where SL_{qf} = salt load associated with the total quickflow volume for the day (mg) and

QF = total quickflow volume, i.e. QF_a + SF_{cd} (l).

In *ACRU2000* a fraction of the stormflow delayed on a particular day is referred to as Carryover. The salt load associated with this Carryover is calculated based on the stormflow salinity and Carryover volume. The salt load associated with the Carryover is then added to the *DSurfaceSaltFluxRecord* Data Object to be released for subsequent days, in proportion to the discharged fraction of delayed stormflow volume.

As mentioned in Section 5.2.3.2, upward salt transport from the topsoil horizon to quickflow can take place when upward saturated flow occurs due to low drainage of the soil horizons and subsequent filling of topsoil pore spaces to their saturation. Thus, if on a particular day, a salt contribution occurs to quickflow due to upward flow, the salt concentration of quickflow is updated in the *PUpwardSaltTransport* and *PIrrigUpwardSaltTra* processes to account for the effect of this phenomenon.

5.3.4 Runoff salinity and salt load

The salt concentration and salt load of runoff water from non-irrigated lands is determined by the *PRunoffSalinity* Process. This process obtains the required data such as baseflow and quickflow depths and their associated salinity from relevant data objects as shown in Figure 5.10. Once this process has received the required data, it determines runoff salinity through simple instantaneous baseflow and quickflow mixing using Equation 5.16. The associated salt load is then calculated using Equation 5.17. It stores runoff salinity and salt load in *DRunoffSalinity* and *DRunoffSaltLoad* data objects respectively for subsequent computations and for an output at the end of each day. Thus,

$$C_{run} = \frac{(BF * C_{bf}) + (QF * C_{qf})}{QF + BF} \tag{5.16}$$

$$SL_{run} = C_{run} * (BF + QF) \tag{5.17}$$

where C_{run} = salt concentration of runoff water (mg/l) and

SL_{run} = the salt load associated with runoff water (mg).

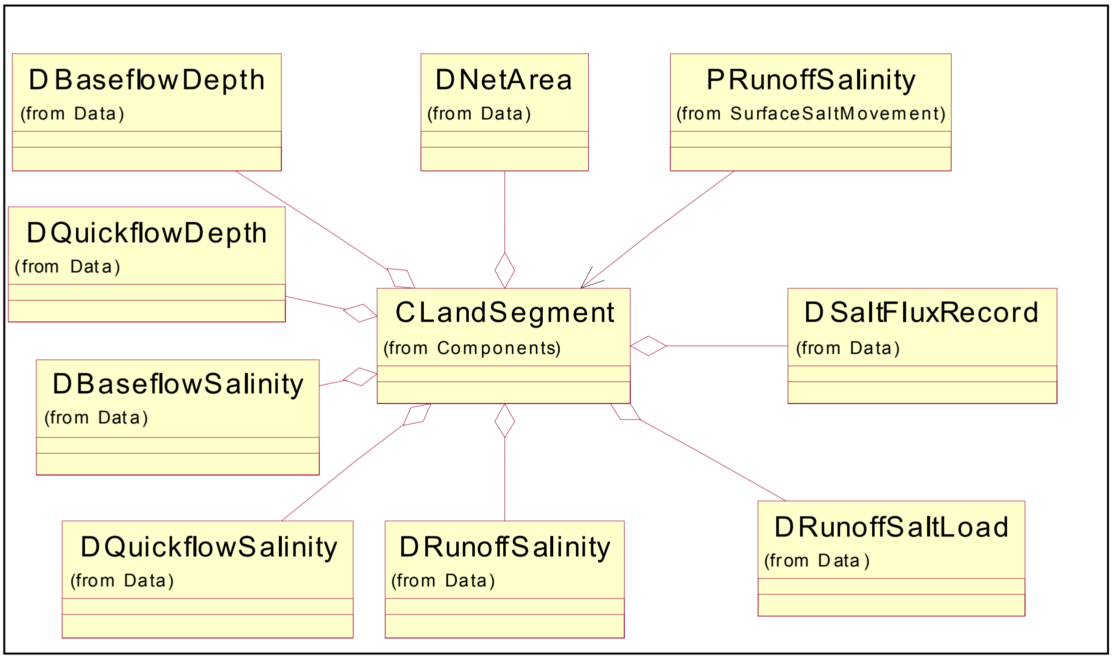

Figure 5.10 Class diagram of *PRunoffSalinity* Process and its associated component and data objects

Runoff salinity and the salt load from irrigated areas are determined in the *PIrrigRunoffSalinity* Process. This process generally follows a similar approach to the one already described to compute salt concentration and salt load associated with runoff water in non-irrigated lands.

5.4 Salt Distribution to Reservoir and Channel Reaches

Salt load associated with runoff water from non-irrigated land, irrigated land and impervious areas is distributed to an appropriate destination component based on the direction of flow configured by the user for water quantity. Thus, the salt allocating processes distribute the stored salt load onto a particular component after they receive relevant information regarding the direction of water flow from the component object under consideration.

5.4.1 Runoff distribution in *ACRU*

The mass of salt load to be allocated to a particular outflow reach depends on the volume of runoff entering the reach and its salt concentration. Runoff volume in turn is a function of area of the contributing land. In the *ACRU* model a reservoir may be situated within or at the outlet of a sub-catchment. If it is located at the outlet of the sub-catchment, the entire sub-catchment area is assumed to contribute its flow to the dam. If, on the other hand, it is located within the sub-catchment, it functions as an internal dam and only a fraction of the sub-catchment area contributes to the dam. In the case of irrigated areas, the total flow from the land enters either the dam or the channel reach. In the case of non-irrigated lands and adjunct impervious areas only part of the total flow enters an internal dam while the remaining fraction enters the channel reach. Thus, to determine the volume of runoff contributing to a particular outflow reach, the net area (non-irrigated area) of a particular sub-catchment (land segment) is divided into an upper and lower net land segment area. In this case, the total net land segment area refers to the gross sub-catchment area excluding the dam, irrigated and impervious areas. The upper net land segment area refers to the fraction of net land segment area upstream of an internal dam, whereas, the lower net land segment area refers to that fraction of the total net land segment area downstream of an internal dam. In *ACRU2000* the upper and lower net land segment areas are calculated based on the following three possible sub-catchment configurations.

I. If no irrigated area exists in the sub-catchment

$$UNLSA\_1 = (A_{gls} - A_{adj} - A_{disj})*f_{dam} - A_{dam} \tag{5.18}$$

$$LNLSA\_1 = (A_{gls} - A_{adj} - A_{disj})*(1-f_{dam}) \tag{5.19}$$

II. If an irrigated area exists in the sub-catchment and irrigation return flows enter the system downstream of an internal dam

$$UNLSA\_2 = UNLSA\_1$$
$$LNLSA\_2 = (A_{gls} - A_{adj} - A_{disj})*(1-f_{dam}) - A_{irrig} \tag{5.20}$$

III. If an irrigated area exists in the sub-catchment and irrigation return flows enter the system upstream of an internal dam

$$UNLSA\_3 = (A_{gls} - A_{adj} - A_{disj})*f_{dam} - A_{dam} - A_{irrig} \tag{5.21}$$
$$LNLSA\_3 = LNLSA\_1$$

where A_{gls} = gross net land segment area (km^2)
A_{dam} = dam area (ha)
A_{irrig} = irrigated land area (ha)
f_{dam} = fraction of the gross catchment area contributing its flow to the dam
A_{adj} = adjunct impervious area (km^2) and
A_{disj} = disjunct impervious area (km^2).

UNLSA_1, *UNLSA_2* and *UNLSA_3* refer to upper net land segment areas under the three different sub-catchment configurations in km^2, whereas, *LNLSA_1*, *LNLSA_2* and *LNLSA_3* refer to lower net land segment areas under the three different sub-catchment configurations in km^2.

The volume of runoff from non-irrigated areas entering an internal dam is expressed as the product of the upper net land segment area and the runoff depth from that area. The volume of runoff entering to the channel reach, in turn, is expressed as the product of the lower net land segment area and the runoff depth.

The preceding concepts for distribution of runoff water from non-irrigated and irrigated lands to the dam and/or channel reaches can be represented by a flow diagram as in Figure 5.11.

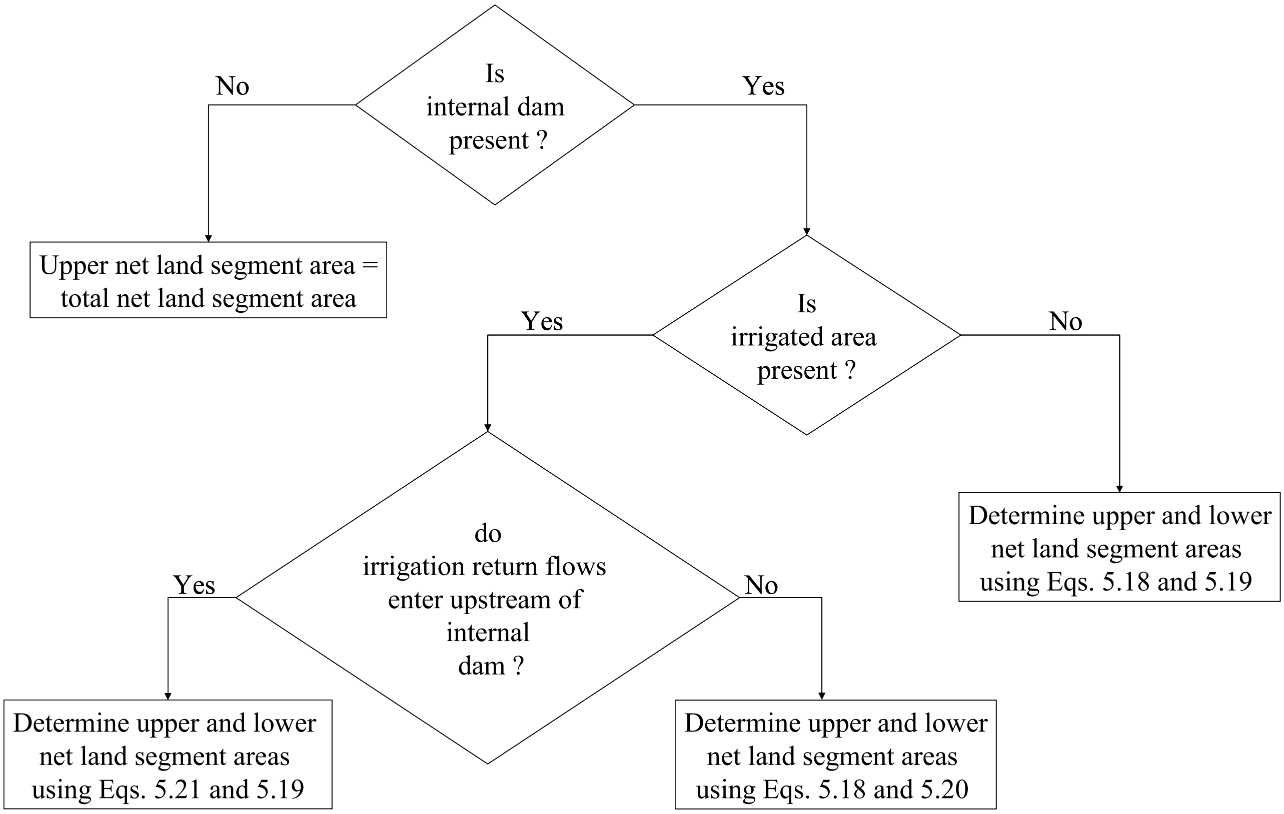

Figure 5.11 Flow diagram for determination of upper and lower net land segment areas

5.4.2 Salt distribution from non-irrigated lands

The process of salt distribution from non-irrigated areas to the appropriate outflow component is carried out by the *PLandSegSaltMovement* Process Object. This process distributes the runoff salt load from non-irrigated areas in a way similar for runoff distribution from these areas. In this process, runoff salt load ends up in a channel reach and/or in a reservoir. However, it assumes that only a single reservoir exists within a sub-catchment (land segment) to which a fraction of the runoff salt load is allocated. This process also assumes that the salt load from non-irrigated land is not distributed to more than one channel reaches. Hence, it considers only a single channel reach at a sub-catchment's outlet, where the remaining runoff salt load ends up. The *PLandSegSaltMovement* and its relationship with the various data and component objects are depicted in Figure 5.12.

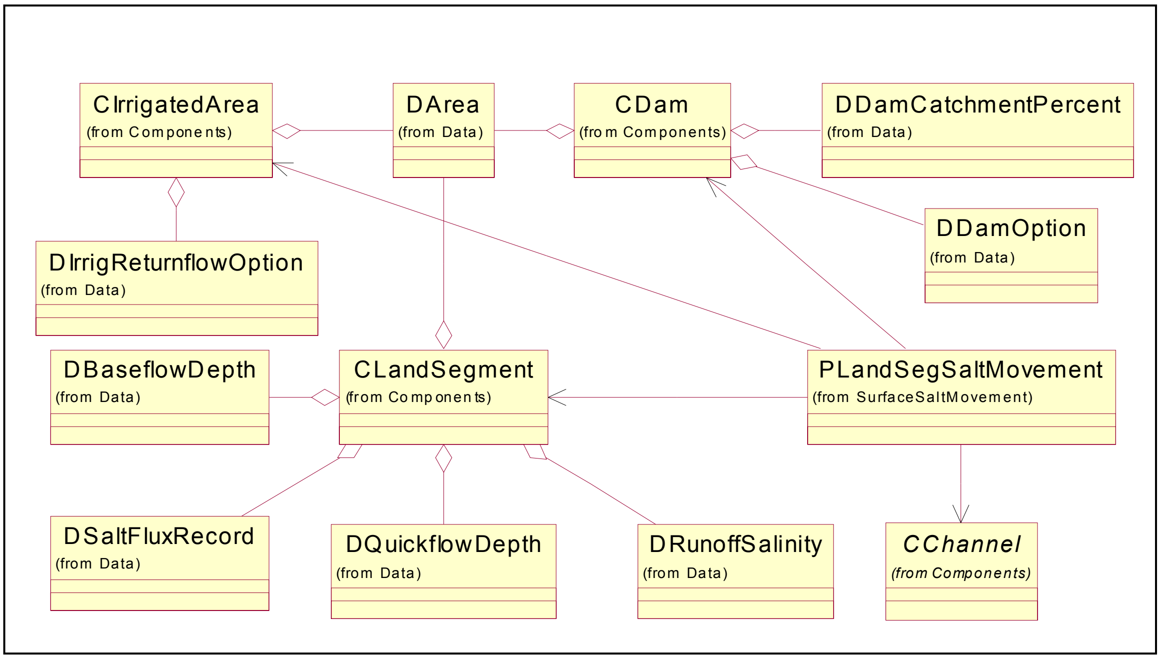

Figure 5.12 Class diagram of the *PLandSegmentSaltMovement* Process and associated component and data objects

Partitioning of the total non-irrigated area in a sub-catchment into that contributing its flow to the dam (upper net land segment area) and that contributing its flow to the channel reach (lower net land segment area) is carried out using a similar algorithm to that used for runoff distribution in hydrological process objects of *ACRU* (Section 5.4.1). After determination of the areas of non-irrigated land contributing to channel and dam reaches, the volume of runoff and its associated salt load entering the dam are calculated as follows:

$$RUN_{dam} = (BF + QF) * UNLSA\_i * 10^9 \quad (5.22)$$

$$SL_{dam} = RUN_{dam} * C_{run} \quad (5.23)$$

where RUN_{dam} = runoff volume entering to the dam (l)

$UNLSA\_i$ = upper net land segment area (km^2) under the i-th sub-catchment configuration and

SL_{dam} = salt load inflowing to the dam (mg).

The other variables have been defined in the previous sections, but *BF* and *QF* in this case represent baseflow and quickflow depths (m) respectively. Similarly, the remaining volume of runoff and its salt load flowing into the channel reach are calculated using the following equations:

$$RUN_{chnl} = (BF + QF) * LNLSA\_i * 10^9 \quad (5.24)$$

$$SL_{chnl} = RUN_{chnl} * C_{run} \quad (5.25)$$

where RUN_{chnl} = runoff volume inflowing to the channel (l)
$LNLSA\_i$ = lower net land segment area (km^2) under the i-th sub-catchment configurations and
SL_{chnl} = salt load entering to the channel (mg).

The calculated masses of salt load flowing into the reservoir and/or channel reaches are then transported from the non-irrigated land to their respective destination components. However, if no dam exists in the sub-catchment or if the dam is situated at the sub-catchment's outlet, the total runoff generated from the sub-catchment flows into the channel reach. Hence, the salt load associated with this runoff is expressed as the product of the total runoff volume from non-irrigated lands and its salinity, as follows:

$$RUN = (BF + QF) * NLSA * 10^9 \quad (5.26)$$

$$SL = RUN * C_{run} \quad (5.27)$$

where RUN = the total runoff volume from non-irrigated land in a sub-catchment (l)
$NLSA$ = total area of the non-irrigated land in a sub-catchment (km^2) and
SL = total salt load associated with runoff from the non-irrigated land (mg).

Figure 5.13 shows the flow diagram of runoff salt load allocation from non-irrigated areas into a dam and/or channel reaches as represented in the *PLandSegSaltMovement* Process.

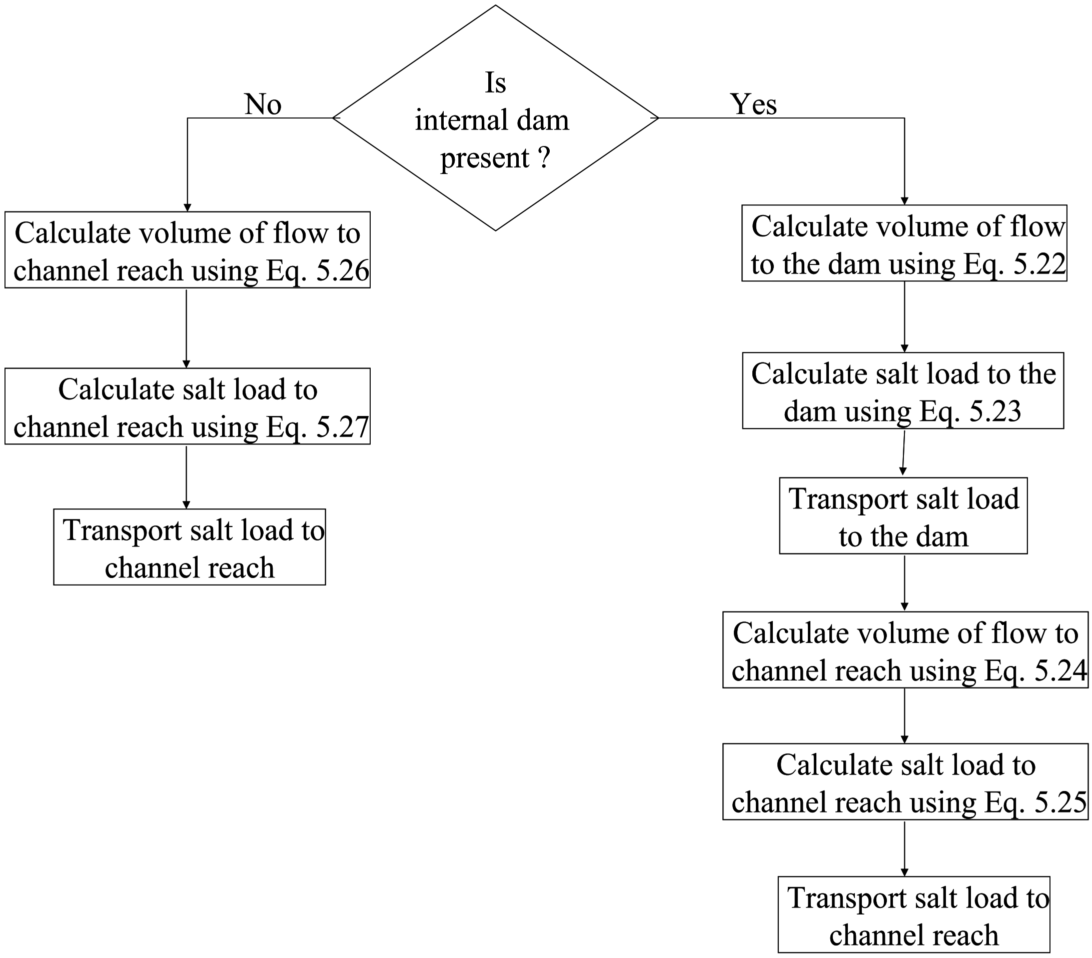

Figure 5.13 Flow diagram of runoff salt load allocation from non-irrigated areas

5.4.3 Salt distribution from irrigated lands

The process of salt distribution from irrigated lands is carried out by the *PIrrigatedAreasSaltMovement* Process. In this process, salt distribution follows the direction of runoff flow from the irrigated land. The allocation of runoff salt load from irrigated lands does not involve the partitioning of salt load to that flowing into the reservoir and channel reaches. Rather, it either ends up in a reservoir or in a channel reach, depending on the location of the irrigated land in relation to an internal dam. If the irrigated land is situated upstream of the dam, then the total salt load associated with runoff water from the irrigated land enters to the dam. If it is located downstream of an internal dam, the total salt load from the irrigated land enters to the channel reach. The *PIrrigatedAreasSaltMovement* and its relationship with the various data and component objects are shown in Figure 5.14.

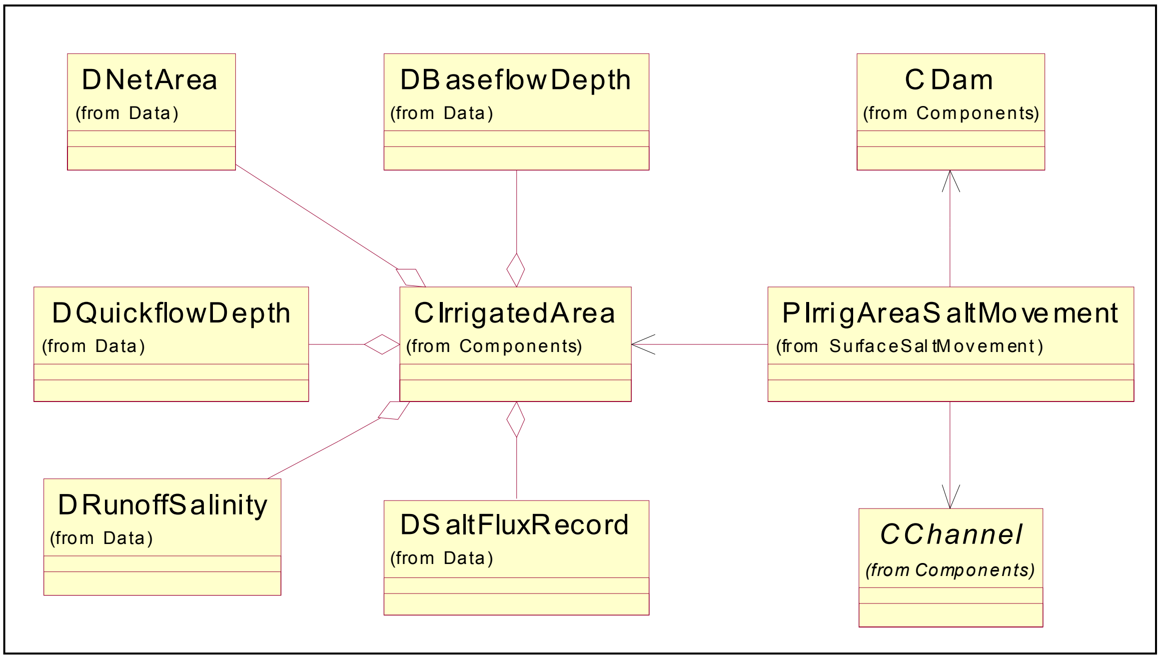

Figure 5.14 Class diagram of *PIrrigatedAreaSaltMovement* Process and associated data and component objects

5.4.4 Salt distribution from impervious areas

The increased need for applying the *ACRU* model to entirely urbanised catchments or to areas where the urban component is significant enough to influence runoff responses initiated the development of a routine that enables the model to simulate runoff from such areas (Schulze and Tarboton, 1995). This routine considers a number of urban land use categories such as business district, industrial and residential areas. These land use types vary according to their percentages of impervious areas to the total area. Since runoff salt load and concentration are directly influenced by runoff volume, the various urban land use categories and their hydrological responses are also likely to have an impact on hydrosalinity processes of these areas.

5.4.4.1 Hydrological responses of impervious areas as conceptualised in *ACRU* model

Impervious areas in the *ACRU* model are described as either adjunct or disjunct. Adjunct impervious areas are those parts of impervious areas adjunct, i.e. connected directly, to a water course, storm water drain or channel. Thus, runoff from adjunct impervious areas contribute directly to streamflow or the storm water system. On the other hand, disjunct impervious areas represent those parts of the impervious area that are disconnected from the water course. Runoff from disjunct impervious areas initially flows into a pervious area and thereby contribute to the soil water budget and runoff response of that pervious area.

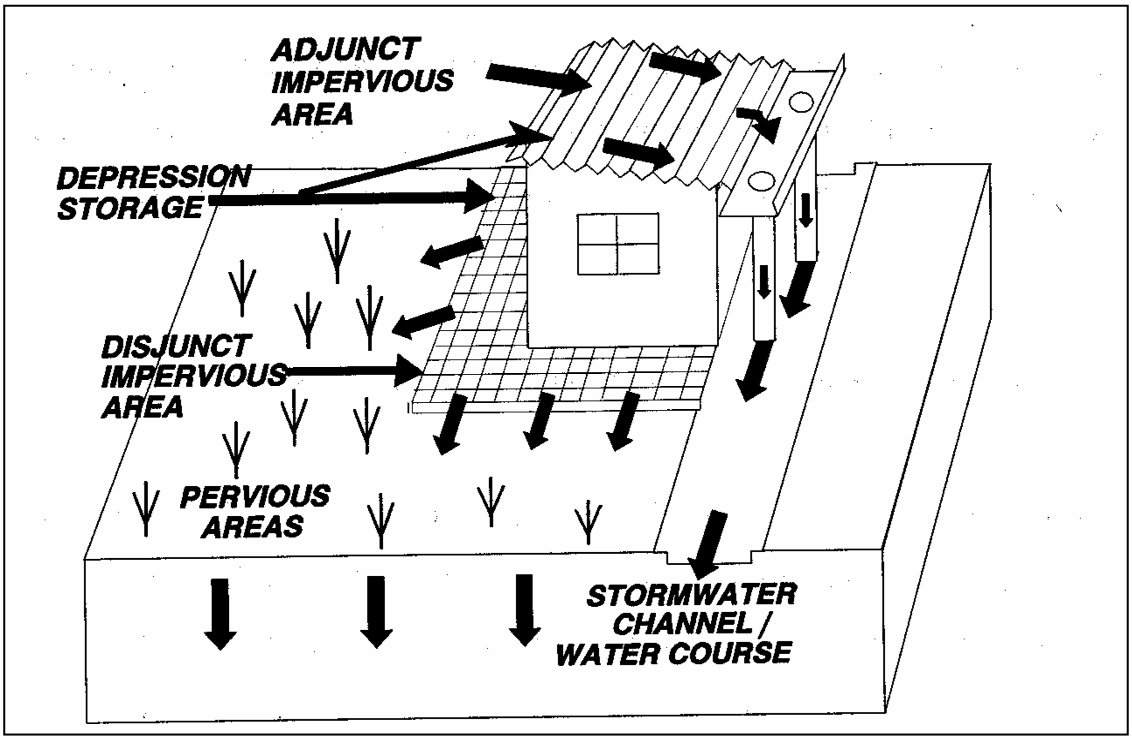

Figure 5.15 Runoff generation from impervious areas as conceptualised in *ACRU* model (Schulze and Tarboton, 1995)

5.4.4.2 Determination and allocation of runoff salt load from impervious areas

Determination of runoff salt loads from adjunct and disjunct impervious areas and their subsequent allocation to an appropriate destination reach is carried out in a single process object, using *PImperviousAreaSaltMovement*. Figure 5.16 shows a class diagram of this process and associated objects. As it can be seen from that figure, both adjunct and disjunct impervious areas are "types of" impervious area (*CImperviousArea*). Hence, it is possible and consistent to represent processes taking place in these areas in a single process object, as done for the runoff component. This process is responsible for distributing salt loads from an impervious area into a channel reach, in the case of adjunct impervious areas, or onto the surrounding non-irrigated area in the case of disjunct impervious areas. Once runoff depth is retrieved from the relevant data object, runoff volume from adjunct and disjunct impervious areas is expressed by two distinct equations. If the impervious area under consideration is disjunct, runoff from such areas is calculated using Equation 5.28.

$$RUN_{dis} = RUN_{imp} * A_{dis} * 10^6 \tag{5.28}$$

where RUN_{dis} = runoff volume from disjunct impervious area (l)
RUN_{imp} = depth of runoff from impervious area (mm) and
A_{dis} = disjunct impervious area (km^2).

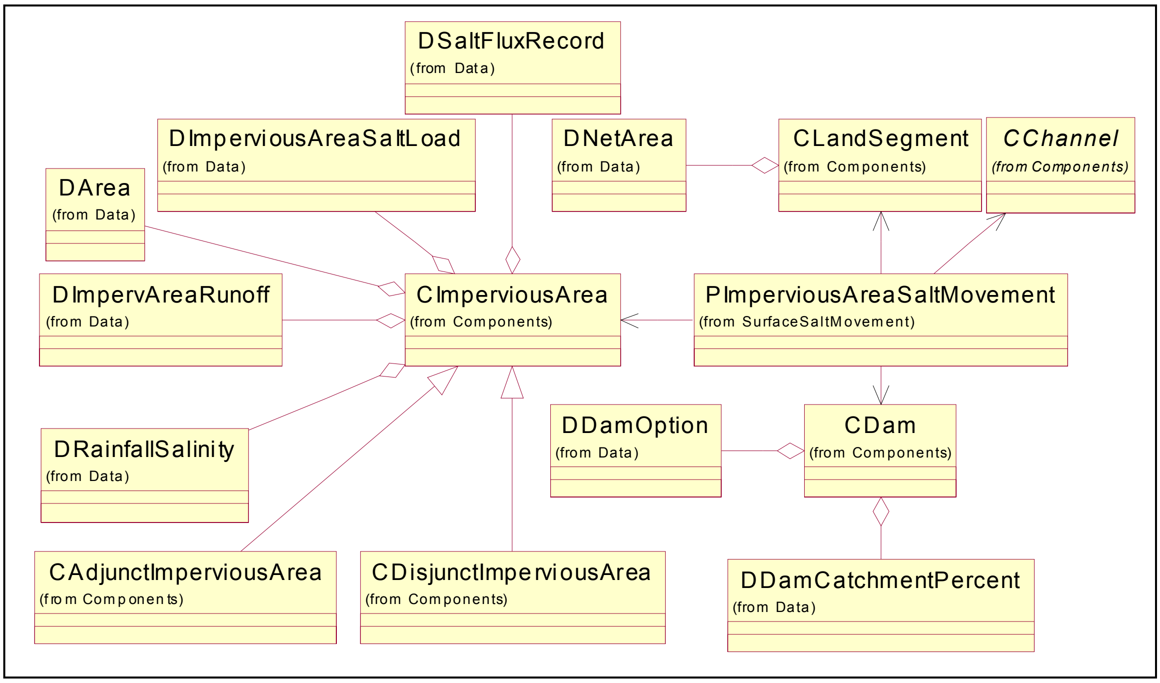

Figure 5.16 Class diagram of *PImperviousAreaSaltMovement* Process and associated objects

In impervious areas the only source of quickflow considered in *ACRU* is from rainfall i.e. there is no contribution from the topsoil as a result of saturated upward, and hence overland, flow. Some urban impervious areas may experience significant salt input due to dry atmospheric deposition with subsequent salt enrichment of the quickflow from such sources. However, due to the complications involved in programming, the present hydrosalinity module of *ACRU* does not account for this phenomenon. Thus, the salt concentration of quickflow from impervious areas is assumed to have the same TDS concentration as that of the rain falling on the area. Similarly, runoff from impervious areas does not include any baseflow component in *ACRU*. Therefore, salt concentration of runoff water from impervious areas is assumed to have the same salinity as the quickflow from the area, which in turn has the same TDS concentration as the rain falling on that area. The salt load associated with runoff from disjunct impervious areas is calculated using Equation 5.29. The calculated salt load is then stored in *DImperviousAreaSaltLoad.*

$$SL_{dis} = RUN_{dis} * C_{run\_imp} \tag{5.29}$$

where SL_{dis} = salt load associated with runoff from disjunct impervious areas (mg) and

C_{run_imp} = salt concentration of runoff water from impervious areas (mg/l), with C_{run_imp} having the same value as the average TDS concentration of rainfall.

The effect of this salt load on surface and subsurface salt balance of the adjacent non-irrigated land is accounted indirectly in the *PStormflowSalinity* and *PSaltInput* processes. The daily stormflow in the hydrological modules of *ACRU* and hence the associated salt load in *ACRUSalinity*, are computed not only from the rain falling on the area for the day but also it includes the surface flow from disjunct impervious areas. Similarly, the daily infiltration to the topsoil of non-irrigated areas is partly comprised of the flow from adjunct impervious areas. Hence, the salt load associated with the infiltration water is partly comprised of the salt load from disjunct impervious areas. On the other hand, if the impervious area is an adjunct type, runoff volume is calculated as in Equation 30 by

$$RUN_{adj} = RUN_{imp} * A_{adj} * 10^6 \tag{5.30}$$

where RUN_{adj} = runoff volume from the adjunct impervious areas (l) and

A_{adj} = adjunct impervious area (km^2).

Furthermore, in the presence of an internal dam, the volume of runoff from adjunct impervious areas, and thus the total salt load, are partitioned into that entering the dam and channel reaches. In this case, the volume of runoff from adjunct impervious areas flowing into the dam (RUN_{adj_dam}) is calculated as the product of the total runoff volume from the adjunct impervious area and the fraction of the land segment contributing to the dam (f_{dam}), as specified by the user. The remaining volume then enters the channel (RUN_{chnl}). The salt load flowing into the dam (SL_{adj_dam}) and channel (SL_{adj_chnl}) reaches are expressed using Equations 5.33 and 5.34.

$$RUN_{adj\_dam} = RUN_{adj} * f_{dam} \tag{5.31}$$

$$RUN_{adj\_chnl} = RUN_{adj} * (1 - f_{dam}) \tag{5.32}$$

$$SL_{adj\_dam} \quad = RUN_{adj\_dam} * C_{run\_imp} \tag{5.33}$$

$$SL_{adj\_chnl} \quad = RUN_{adj\_chnl} * C_{run\_imp} \tag{5.34}$$

The calculated salt load is then transported to its destination component (to the dam or channel reach). However, if no dam exists in the sub-catchment, the total runoff salt load from the adjunct impervious area is calculated using Equation 5.35 and is transported to the channel reach.

$$SL_{adj\_chnl} = RUN_{adj} * C_{run\_imp} \tag{5.35}$$

The preceding concepts of salt load determination and distribution from impervious areas are summarised in the following flow diagram.

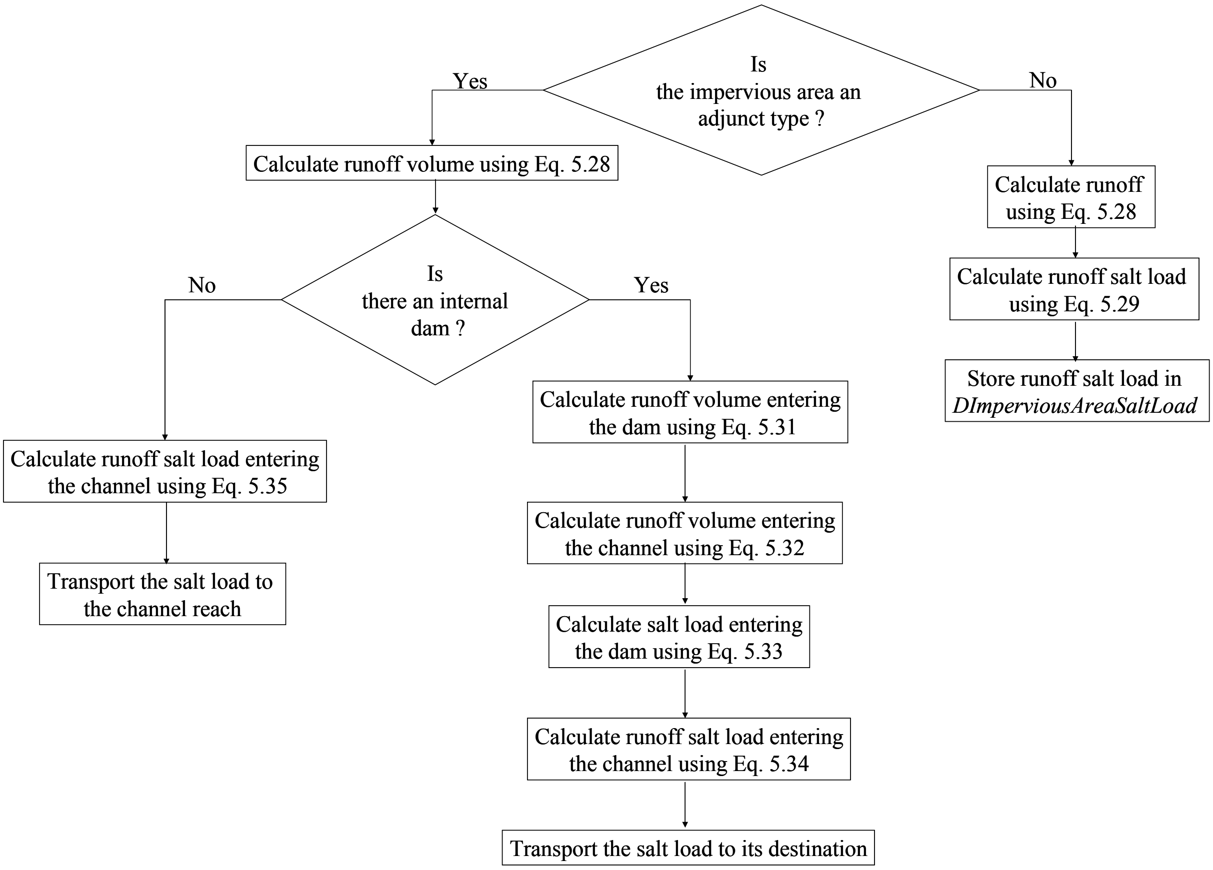

Figure 5.17 Flow diagram of salt load determination and distribution from impervious areas

5.5 Reservoir Salt Budget and Salt Routing

According to Bath *et al.* (1998), water quality models for reservoirs are developed for two main reasons. First, they can be used as research tools to establish an understanding of the complex interactions taking place between various processes. Secondly, these models are used as management and planning tools to provide necessary information for decisions on the abatement of water quality problems. These include short-term operational decisions to provide water quality and hydrological information and long term planning or design decisions where information is required on the influence of water resources developments and blending options.

Reservoir hydrosalinity models can be used to provide information on the governing processes and their influences on salt concentrations of reservoir storage and the various outflows. Such information is in turn frequently used to provide information on the design of water treatment works and design operational and management guidelines for the reservoirs and its associated upstream areas. These include extracting useful information from model output that can be employed on timing of freshwater imports and downstream releases.

5.5.1 Reservoir water budgeting in *ACRU*

As accounted in the reservoir yield analysis section of the *ACRU* model (Schulze *et al.*, 1995d), the main components of reservoir water budget can be broadly classified as gains (inflows) and losses (outflows) from the system. Gains to the system include streamflows, inter-catchment transfers and precipitation onto the reservoir water surface. The loss component comprises evaporation from the water surface, legal water releases, seepage losses, overflows and irrigation as well as other abstractions. The various gains and losses from a reservoir system as conceptualised in *ACRU* model are shown in Figure 5.18.

5.5.1.1 Gains to the system

The following are gains to the reservoir system:

i. Streamflow: This is usually the major gain to the system and includes both stormflow and baseflow.
ii. Precipitation: This constitutes a second gain to the system. In *ACRU* all precipitation falling onto the entire surface area at full capacity is added. This is based on the assumption that when the reservoir is not at full storage, the adjacent dry parts are compacted and surface sealing has taken place. In some hydrosalinity models the salt contribution of precipitation is neglected, however, it can have significant effect particularly in areas where atmospheric deposition is dominant source of salt input.
iii. Inter-catchment transfers to the reservoir: This input to the reservoir is from outside the catchment which contributes streamflow. The present reservoir salt budget and salt routing routine of *ACRUSalinity*, however, does not account for inter-catchment salt load transfers.

5.5.1.2 Losses from the system

The following are losses from the reservoir system:

i. Abstractions from the reservoir: These include losses from the system and they include irrigation, domestic and other abstractions.
ii. Seepage losses: Daily seepage from earth-walled and unlined reservoirs may be estimated as equivalent to 0.0006 of the storage capacity (Schulze *et al*., 1995d). This approximates that the reservoir empties about once in every five years as a result of seepage losses, disregarding any other losses or gains.
iii. Evaporation losses: Reservoir evaporation takes place from a large and usually relatively deeper water body, while the A-pan equivalent evaporation is subjected to local climatic and advective fluctuations. Therefore, in order to account for the varying relationship between a large water body and the A-pan, *ACRU* uses seasonally and regionally dependent month by month adjustment coefficients that are input by the user.
iv. Legal flow releases: This is a legal release of water from the reservoir for downstream riparian and other users.
v. Overflow: *ACRU* treats the temporal distribution of overflow in two ways. When the hydrograph routing option is not invoked, the reservoir water budget is calculated on a daily basis. In this case, the assumption is made that storage in excess of the maximum capacity spills from the reservoir on the same day. Thus, the maximum storage possible at the end of the day is equal to maximum capacity. If the hydrograph routing option is invoked, the upstream hydrograph, and the other gains to the system, are routed in sub-daily time steps through the reservoir using the storage indication method. Since this method accounts for storage above the full capacity level, for large dams the storage in the dam at the end of the day may be greater than the maximum reservoir capacity (Schulze *et al*., 1995d). The current reservoir salt budget routine in *ACRUSalinity,* however, considers salt routing only at a daily time step and not at sub-daily ones.

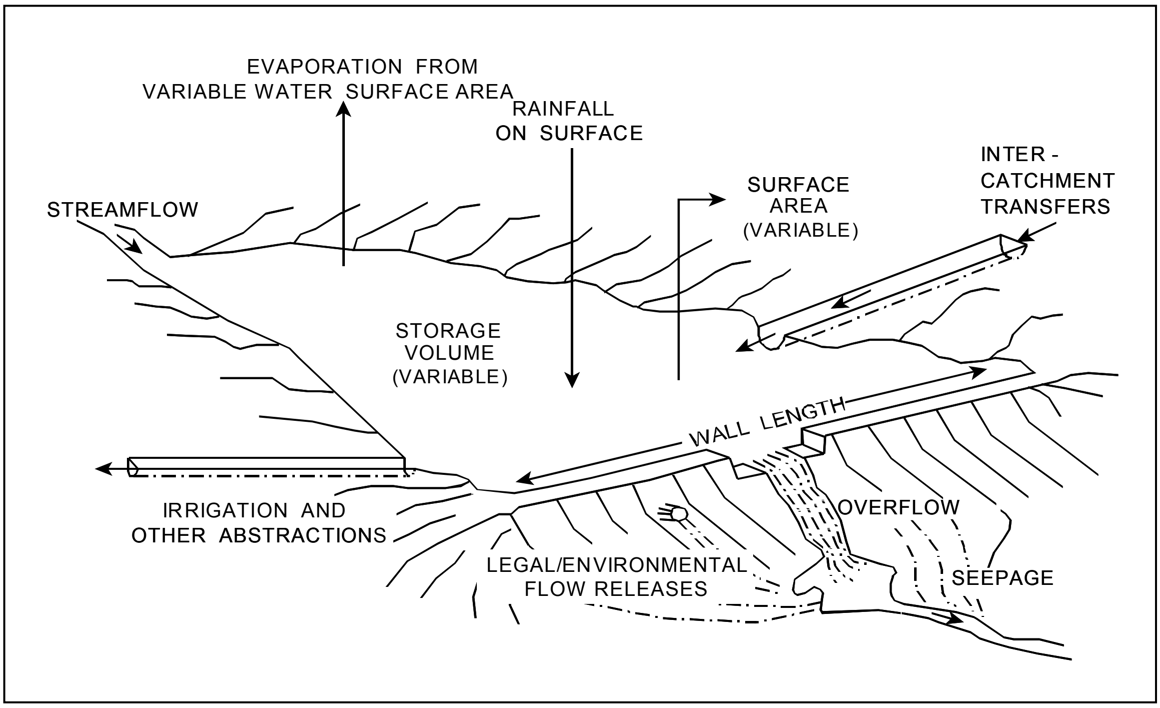

Figure 5.18 Reservoir water budget as conceptualised in *ACRU* model (Schulze and Smithers, 2002)

5.5.1.3 Surface area to storage relationship

Computations on the evaporation loss from a reservoir system and thereby its effect on salt concentration require knowledge of the surface area at various storage volumes of the reservoir. This can be estimated in *ACRU* with and without reservoir basin survey information (Schulze *et al.*, 1995d). The surface area to storage volume relationship, when the reservoir basin has been surveyed, is determined from the relationship that $A_s = a\ (Sv)^b$, where A_s stands for surface area of water (m^2) on a given day, Sv = storage (volume) of water (m^3) calculated from the previous day's final reservoir water budget, and "*a*" and "*b*" are the constant and the coefficient of the equation respectively, determined from the survey. According to Arnold *et al.* (1996), the coefficient "*b*" is a fairly constant parameter (0.9) and thus the constant "a" can be determined using maximum surface area and maximum storage of the reservoir. On the other hand, if no basin survey exists for the reservoir, *ACRU* computes the default surface area to storage relationship for different shapes of surface area from the reservoir wall length.

5.5.2 Determination of TDS concentration of reservoir storage and outflows

The reservoir salt budgeting computations are carried out by the *PReservoirComponSalinty* Process. This process operates in conjunction with the *PSaltStacking* Process to determine the reservoir's current storage salinity and salt load as well as TDS concentration of the various outflows from the reservoir system.

This process prepares the main data input requirements of the *PSaltStacking* Process. These include total volume of water flowing into the reservoir and its salinity, as well as total volume of water flowing out from the reservoir, excluding evaporation losses. The total volume of water flowing into the reservoir system, which comprises runoff from irrigated and non-irrigated lands, adjunct impervious areas as well as rain falling on the surface of a reservoir, is obtained from the daily total water influx record of the reservoir, as determined by the hydrological modules of *ACRU*. However, the salt load associated with the various inflow sources varies depending on the flow volume and salinity of each source. Hence, the required data for these flow components are also retrieved from the relevant individual data objects, as shown in Figure 5.19. As in the case of salt input to irrigated and non-irrigated lands, here again, a steady state rainfall salt concentration is assumed. The average TDS concentration of the total inflow from the various sources is then determined through instantaneous mixing of the different inflows using Equation 5.36. The salinity level of reservoir inflows is computed not only for use as a major input to the advection model, but also to help the user to anticipate the average salt concentration of reservoir inflows under different combinations of hydrological, climatic and catchment conditions, including upstream land use practices. Therefore, the average TDS concentration and salt load of the total reservoir inflow are stored in the *DResInflowSalinity* and *DInflowSaltLoad* data objects respectively for use in other computations and an output at the end of the day. Thus,

$$Cin_i = \frac{(RUN_{ni} * C_{run\_ni}) + (RUN_{irr} * C_{run\_irr}) + (RFL_{dam} * C_r) + (RUN_{adj\_dam} * C_{run\_adj})}{I_{dam}} \qquad (5.36)$$

where Cin_i = average salt concentration of water flowing into the dam (mg/l)

RUN_{ni} = runoff flowing into the dam from non-irrigated lands (l)

C_{run_ni} = Salt concentration of runoff from non-irrigated lands (mg/l)

RUN_{irr} = runoff from irrigated areas (l)

C_{run_irr} = salt concentration of runoff water from irrigated areas (mg/l)

RFL_{dam} = volume of rain falling on the dam surface (l)

C_r = rainfall salt concentration (mg/l)

RUN_{adj_dam} = runoff from adjunct impervious areas inflowing to the dam (l)

C_{run_adj} = salt concentration of runoff from adjunct impervious areas (mg/l) and

I_{dam} = total water inflow to the dam on the day including rain falling on surface of the dam (l).

The total outflow from the reservoir system that comprises legal flow releases (normal flow), abstractions from the reservoir, spillway overflow, seepage and evaporation from reservoir surface is obtained from the total outflux record of the reservoir, as determined by the hydrological processes of *ACRU*. This record includes evaporation from the reservoir surface. However, this process assumes that evaporation losses from the reservoir system have a salt concentrating effect by leaving the salts behind. Therefore, in order to accommodate this assumption, the total outflow from the reservoir which influences the salt load released from the system is reduced as described by the following equation:

$$Total\ outflow = total\ reservoir\ out\,flux\ record - reservoir\ evaporation \qquad (5.37)$$

The TDS concentration at reservoir current storage is then computed in the *PSaltStacking* Process based on the information sent from this process on total inflow and outflow volumes, as well as the average salt concentration of the total inflow to the reservoir, as described by equation 5.36. One of the basic assumptions in the reservoir salt budget computations is a complete mixing of the reservoir at the end of each time step. Thus, no stratification in salt concentration is assumed to occur through out the depth of the reservoir. Hence, the outflow components that include legal flow releases, abstractions from the reservoir, spillway flow and seepage are assigned an average TDS concentration value as determined by the *PSaltStacking* Process. The corresponding salt load associated with the various outflow components is then calculated in this process as the product of the volume of water in the particular outflow component and the outflow salinity as determined by the *PSaltStacking* process. The *PReservoirComponSalinty* and its relationship with the various component, data and process objects is depicted in Figure 5.19.

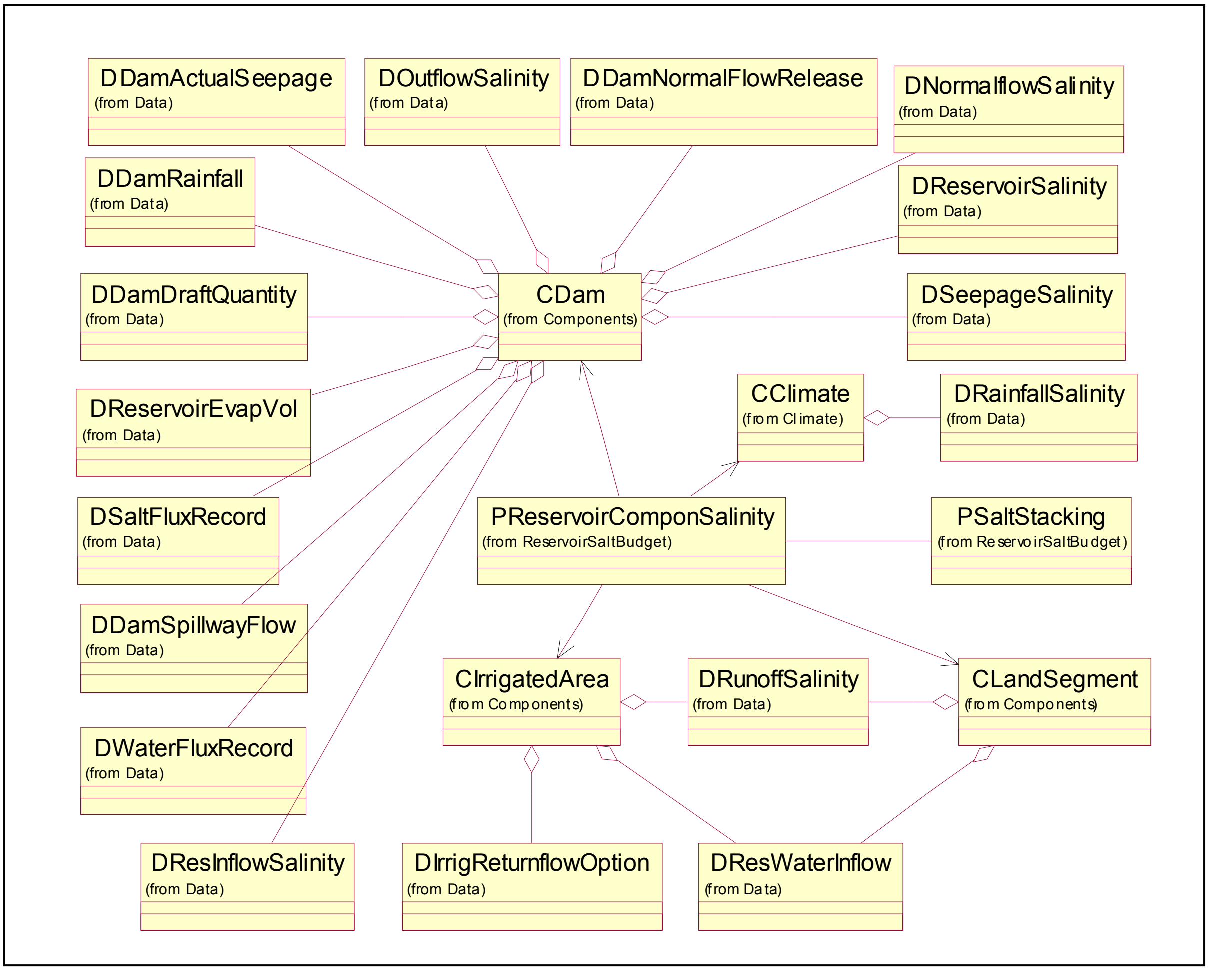

Figure 5.19 Class diagram of *PReservoirComponSalinity* Process and associated data and component objects

5.5.3 The *PSaltStacking* Process Object

Transport and mixing mechanisms of water and solutes in a stream or reservoir can be categorised as advective and dispersive. According to Michael (1997b), a stream that exhibits purely advective flow is said to undergo "plug-flow". Under ideal plug-flow, the length, shape and peak concentration of a dye tracer cloud will remain unchanged during transport downstream. On the other hand, dispersive transport moves solutes from areas of higher concentration to areas of lower concentration. Thus, with dispersive transport a dye tracer cloud will expand in length and reduce in peak concentration over time. This section describes reservoir storage salinity and salt concentration of an outflow from the dam based on the reservoir water budget and the two-cell plug-flow models.

The *PSaltStacking* Process Object in *ACRUSalinity* determines the average TDS concentration of the current storage and outflows from the reservoir. This process performs its computations when invoked by the *PReservoirComponSalinity* Process. It is thus not a stand-alone process. However, the main reasons for making this process take place in a separate class are to avoid complexity of the *PReservoirComponSalinity* and to facilitate re-use of the algorithm in the future for other processes that involve advective transport and mixing of salts. As mentioned in the previous section, this process obtains daily data on the volume of water inflowing to the dam and its salinity, as well as an outflow volume from the dam as determined by the *PReservoirComponSalinity* Process. Furthermore, this process also obtains other required data for reservoir storage salinity and outflow salinity computations from relevant data objects. The bi-directional association between the two process objects is shown in the UML diagram making up Figure 5.20.

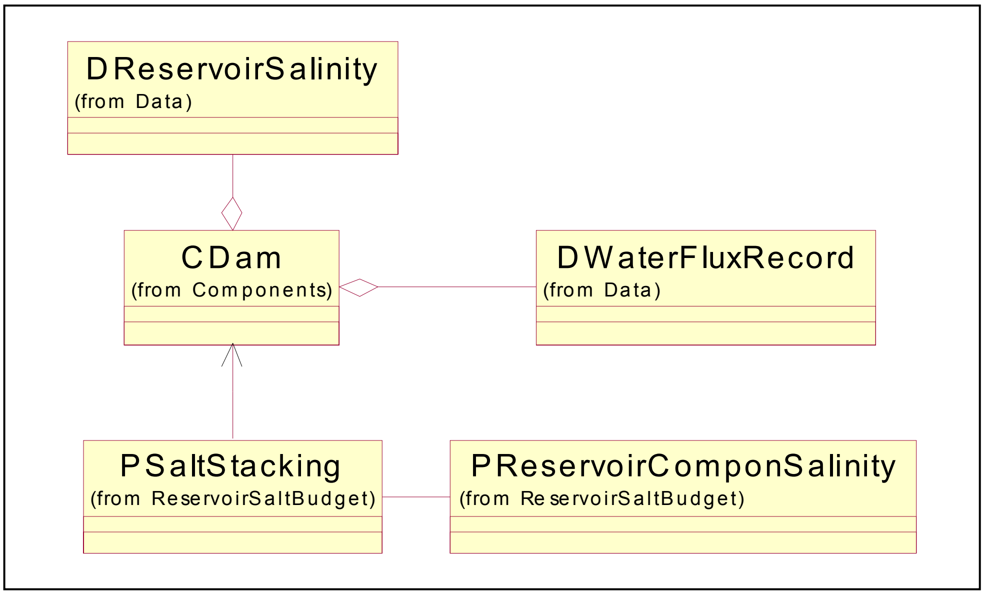

Figure 5.20 Class diagram of *PSaltStacking* Process and associated data and component objects

TDS concentration computations for the reservoir's current storage and outflows are accomplished using a simplified mixing and routing procedure as employed by Herold (1980). The method is based on the assumption that complete mixing occurs within the time step and advection is described by means of a two-cell plug-flow model. The first cell contains the mixed contents of the reservoir at the end of the previous day, while the second cell comprises all the inflows to the reservoir during the day being simulated. This process considers two cases. The first arises when outflow of water from the dam is less than the storage at the end of previous time step (Figure 5.21 a). In that case, the salinity of water leaving the reservoir is set equal to reservoir salinity at the end of the previous day and the reservoir salinity at the end of the day is calculated as follows:

$$C_i = \frac{Qin_i * Cin_i + C_{i-1} * (S_{i-1} - Qout_i)}{S_i} \tag{5.38}$$

where C_i = reservoir salinity at the end of the current day of simulation (mg/l)

Qin_i = water inflow to the reservoir on the current day of simulation (l)

Cin_i = salt concentration of inflowing water on the current day of simulation (mg/l)

C_{i-1} = reservoir salinity at the end of the previous day (mg/l)

S_{i-1} = volume of water stored in the reservoir at the end of the previous day (l)

$Qout_i$ = water outflow from the reservoir for the current day of simulation (excluding evaporation loss) (l) and

S_i = volume of water stored in the reservoir at the current day of simulation (l).

The second case arises when outflow of water from the dam is greater or equal to the storage at the end of the previous day (Figure 5.21 b). In that case, the average TDS concentration of an outflow from the reservoir is described by Equation 5.39 and the reservoir salinity at the end of the day is calculated using Equation 5.40.

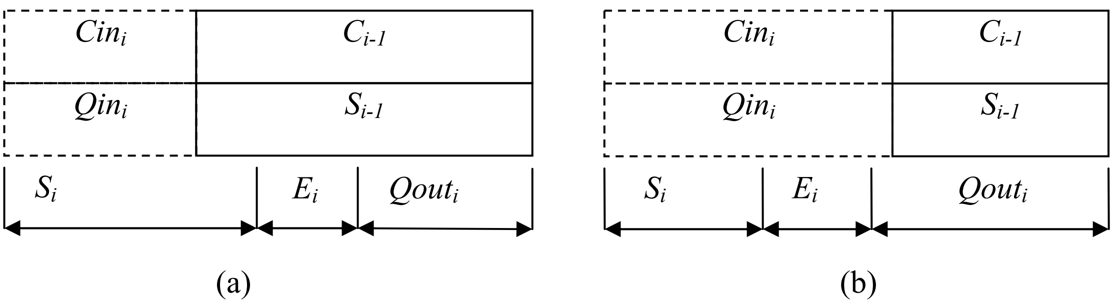

Figure 5.21 Plug-flow cells for the cases (a) when outflow is less than storage and (b) when outflow is greater or equal to storage (after Herold, 1980)

$$Cout_i = \frac{C_{i-1} * S_{i-1} + Cin_i * (Qout_i - S_{i-1})}{Qout_i} \tag{5.39}$$

$$C_i = \frac{Cin_i * (Qin_i - (Qout_i - S_{i-1}))}{S_i} \qquad (5.40)$$

This process generally assumes that evaporation losses take place at the end of the day. It is also based on the assumption that evaporation losses from the reservoir surface tend to concentrate the salts in the reservoir by leaving the salts behind. The basic steps followed by this process are shown in the following flow diagram.

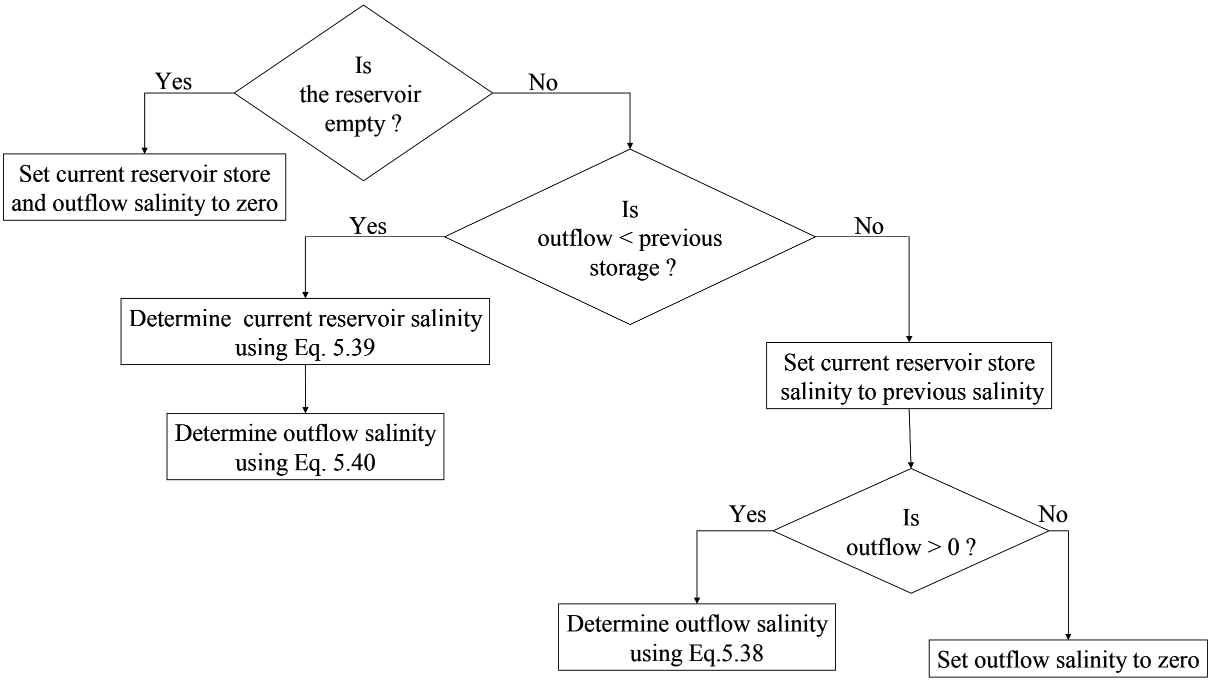

Figure 5.22 Flow diagram of the *PSaltStacking* Process

5.6 Channel Salt Movement and Distributed Hydrosalinity Modelling

As mentioned in Section 5.1.3, *ACRU2000* is structured in a way that, on each day of simulation, all processes taking place in *CReach* type (for example, in channel reaches) are executed only after processes for the land segment have been executed, starting from the edge towards an exit of the simulated catchment. Processes for each *CReach* type in turn are executed starting with reaches on the edge of the flow network and moving progressively downstream. Thus, salt balance computations for a channel reach of a particular sub-catchment are also carried out after all other *ACRUSalinity* processes operating in the sub-catchment have been executed.

The *PCatchmentSalinity* Process determines the salt load and concentration of the water flowing out of a particular channel reach. The salt load from the different sources in a sub-catchment such as irrigated and non-irrigated lands, reservoirs as well as impervious areas, entering the channel reach is determined and transported to the channel reach in relevant process classes. This process then determines the total salt load stored at the channel reach at the end of the day and the streamflow salinity as calculated in the reach. The average TDS concentration of streamflow at the channel reach is determined by using Equation 5.41:

$$C_{chnl} = \frac{SL_{\inf\_t}}{STFL} \tag{5.41}$$

where C_{chnl} = TDS concentration of flow at the channel reach (mg/l)

SL_{inf_t} = daily total salt load stored in the channel reach (mg) and

$STFL$ = volume of streamflow at the channel reach (l).

In the case of distributed hydrosalinity modelling, where more than one sub-catchment is considered, the channel reach in a sub-catchment receives salt load not only from sources within the sub-catchment, but also from upstream sub-catchments. Therefore, in the case of distributed hydrosalinity modelling, the salt load stored in a particular channel reach is transported as salt influx to a downstream reach. The *PCatchmentSalinity* Process carries out the transport of salt load in the channel reach to a downstream channel or reservoir reaches. The *PReservoirComponSalinity* Process also performs a similar function if a particular reservoir is situated at sub-catchment outlet and a downstream reach exists in the catchment being simulated. In this case the total salt load associated with overflow, seepage and legal flow release from the reservoir is computed using Equation 5.42 and is transported to an appropriate downstream reach.

$$SL_{dam\_of} = (OF_{dam} * C_{of}) + (SEEP_{dam} * C_{seep}) + (NF_{dam} + C_{nf}) \tag{5.42}$$

where SL_{dam_of} = total salt load released from the dam to downstream reaches (mg)

OF_{dam} = overflow volume (l)

C_{of} = salt concentration of overflow (mg/l)

$SEEP_{dam}$ = volume of seepage water from the dam (l)

C_{seep} = salt concentration of seepage water (mg/l)

NF_{dam}		= legal (normal) flow release volume (l) and
C_{nf}		= salt concentration of legal flow (mg/l).

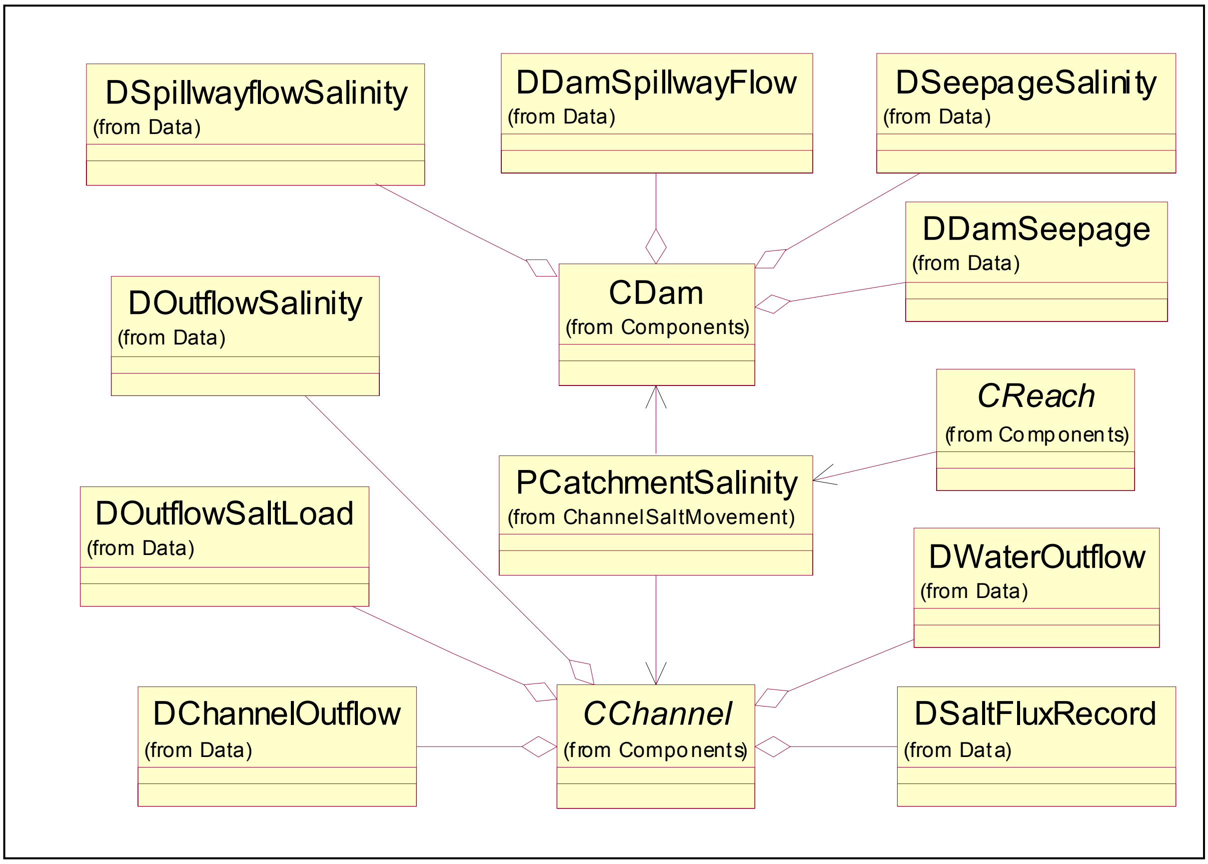

Figure 5.23 Class diagram of *PCatchmentSalinity* and associated component and data objects

To sum up, one of the basic assumptions in *ACRUSalinity* is that solutes are transported along with the moving water and in the direction of water flow (advection). Therefore, salt load transport in subsurface components, i.e. from rainfall salt input through soil horizons to the groundwater store and runoff, as well as the allocation of this runoff salt load to various destination components within a sub-catchment and to downstream reaches follows the direction of water flow as specified by the user and / or as determined by the hydrological modules of *ACRU2000*.

This chapter has reviewed the development of the hydrosalinity module of *ACRU* with special emphasis on how the various hydrosalinity processes are represented in the module. Following the development of *ACRUSalinity*, a verification study was carried out to see how the new module performed under catchment conditions. This phase involved two main steps. The first step was code validation and the second step was comparison of module outputs against observed data. The validation and verification procedures, and subsequent sensitivity analysis of the major inputs to the module as well as a case study that shows some potential applications of the module for planning, design and management of water resource developments will be the subjects of discussion of the next two chapters.