Generic Integrated Design for Irrigation and Fertilization Applied on Tomato

1. Introduction

This report is intended provide a detailed generic design for the Fertigation system. Valid for any crop including tomato (our current system which is cultivated in different environments, open field, under high tunnel, under low tunnel). Both Irrigation and Fertilization systems have been designed and developed for this crop separately [TR/CLAES/72/99.5] [TR/CLAES/140/2000.5]. Therefore, the purpose of this report is to concentrate on the integration point of view taking into consideration also the comments in the updating and reviewing technical reports number TR/CLAES/195/2001.1, and TR/CLAES/189/2001.1

The report is organized as follows: section two provides an overview of the system architecture. Section 3 enumerates the irrigation and fertilization system common concepts and relationships (Common Knowledge). Section 4 and 5 provide the irrigation and fertilization system updating remakes (in terms of addition, deletion, or renaming of concepts, properties, and/or relations). Section 6 describes the integrated system global Interface. Section 7 shows the integrated system overall Control. Finally section 8 contains a description of the testing method.

2. System Architecture

The overall architecture of the Fertigation system is shown in figure 1. As shown in this figure the system consists of:

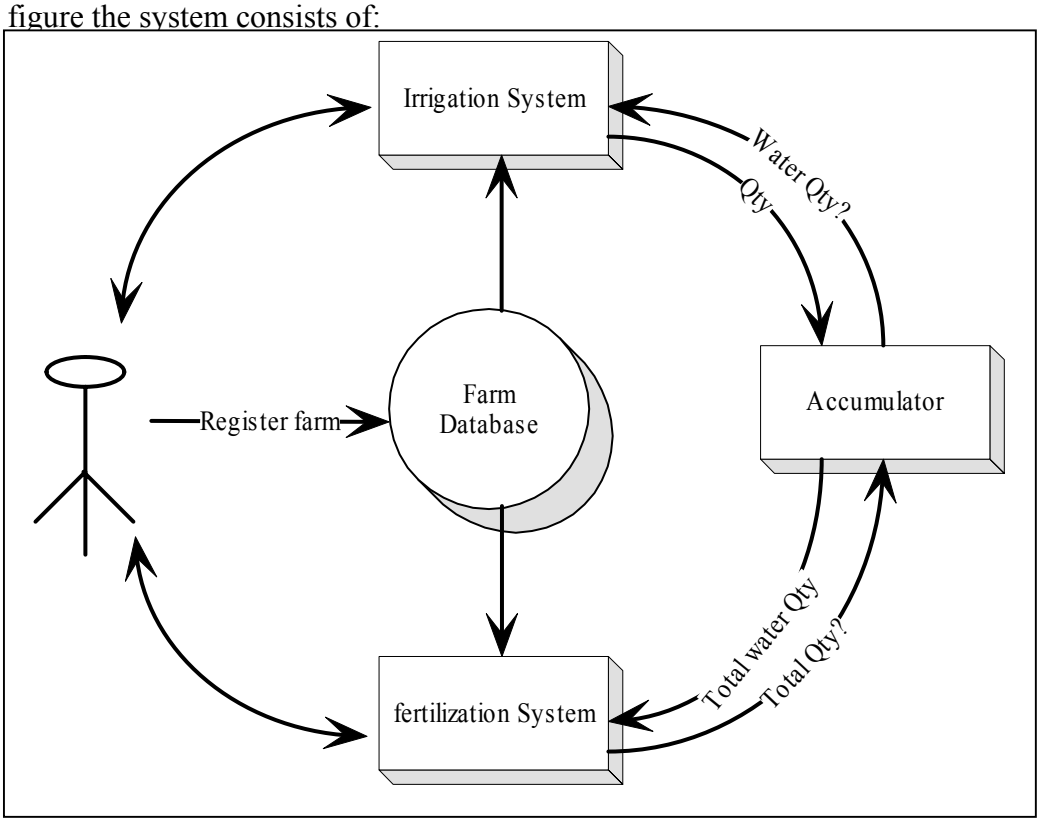

Figure 1: Fertigation System Architecture

1. Irrigation system that accept the farmer, and the weather data from either user or through database, and provide an irrigation schedule suitable for that specific farm.
2. Fertilization system, which acts the same way the irrigation system do, but in this case it will provide a fertilization schedule
3. Database: work as repository
    a. Farm data
    b. Weather data
4. Accumulator. Its job is to invoke the irrigation system several time to calculate the total irrigation water a specific fame may need. This information is sent to the fertilization system to start its function for calculating the appropriate nutrition quantities and schedule. The algorithm of this module is shown in figure 2.

```
accumulator :: {
water_requirment_total :-
                wr_total :: set(previous_wr(0)),
                :findall(W/D/Q/I/M1/N, irr_result(W, D, Q, I, M1, N), Irr_List1),
                :: wr_loop(Irr_List1)&
wr_loop([]) &
wr_loop([_/_/Q/_/_/_|List]) :-
water_requirement :: set(value(Q)),
        irrigation_task_user ::
mathematical_dependency_mechanism([wr_total]),
        wr_total :: get(value(W1)),
        wr_total :: set(previous_wr(W1)),
        wr_loop(List)&
wr_total/fed :-

        wr_total :: get(value(W1)),
                farm :: get(type(Type)),
        (       (Type ='open_field' ;  Type ='low_tunnel')  ->
                irrigation_task_user ::
mathematical_dependency_mechanism([wr_total_o]),
wr_total_o :: get(wr_value(M1))
        ;
                Type = 'high_tunnel' ->
                irrigation_task_user ::
mathematical_dependency_mechanism([wr_total_h]),
                wr_total_h :: get(wr_value(X1)
```

Figure 2: Accumulator Algorithm

3. Common Knowledge

3.1 Common Ontology

```
concept irrigation;
        sub-type-of : operations.
        properties :
                        method: NOMINAL,
                                VALUE-LIST(dripping; flooding)
                                SOURCE(D.B.)
                                SINGLE
                                NECESSARY.

                       schedule_type : NOMINAL,
                                VALUE-LIST(daily ; weekly),
                                SOURCE(User).


                        irrigation_efficiency: INTEGER,
                                NUMBER_RANGE(0;100),
                                SOURCE(D.B.,  derived[table,
                                      irrigation_efficiency_t])
                                SINGLE.

                       number_of_dripper : INTEGER,
                                NUMBER-RANGE(100,4000),
                                SOURCE(D.B.)
                                SINGLE.

                        rate_of_dripper_flow: REAL,
                                NUMBER-RANGE(1,100),
                                SOURCE(D.B.)
                                SINGLE
                                NECESSARY.
                        control_of_dripper: NOMINAL,
                                VALUE-LIST(yes;no),
                                SOURCE(D.B.)
                                SINGLE.

                        controled_water: NOMINAL,
                                VALUE-LIST(yes;no),
                                SOURCE(D.B.)
                                SINGLE
                                NECESSARY.

                        user_suggested_interval: integer,
                                NUMBER-RANGE(1;30),
                                SOURCE(D.B.)
                                SINGLE.
                       saa: REAL, %soil absorbed area
                                NUMBER-RANGE(0,1000),
                                SOURCE(D.B.; Derive[table,
                                        Saa_t])
                                SINGLE.
                       last_irr_date: DATE,  %last irrigation date
                                SOURCE(user)
                                SINGLE.
                       Last_wr: REAL, %last water requirement
```

NUMBER-RANGE(0,1000),
SOURCE(user)
SINGLE.

concept current_planting;
properties :
no_of_plants: INTEGER,
NUMBER-RANGE(1,2000),
SOURCE(D.B.)
SINGLE
NECESSARY.

date : DATE,
SOURCE(D.B.)
SINGLE
NECESSARY.

agriculture_method: NOMINAL
VALUE_LIST(seeding;transplanting)
SOURCE(D.B.)
SINGLE.
NECESSARY.
death_of_plants: NOMINAL
VALUE_LIST(yes,no)
SOURCE(User)
SINGLE.
NECESSARY.

concept water;
properties:
eciw: numeric;
source of value: DB;
cardinality: single;

Ca quantity: numeric;
source of value: user;
cardinality: single;

N quantity: numeric;
source of value: user;
cardinality: single;

P quantity: numeric;
source of value: user;
cardinality: single;

K quantity: numeric;
source of value: user;
cardinality: single;

Mg quantity: numeric;
source of value: user;
cardinality: single;

Fe quantity: numeric;
source of value: user;
cardinality: single;

Zn quantity: numeric;
source of value: user;
cardinality: single;

Mn quantity: numeric;
source of value: user;
cardinality: single;

Cu quantity: numeric;
source of value: user;
cardinality: single;

qty: numeric; % the irrigation volume in cubic meter/feddan
source of value: user;
cardinality: single;

concept material.

concept fertilizer;
sub-type-of : material

concept soil
properties:

texture: NOMINAL,
VALUE-LIST(heavy clay; silt clay; loam; silt loam; silt clayloam; sandy clay loam; sandy loam; loam fine sand; medium fine sand),
SOURCE(D.B.)
SINGLE.

type: (fine, medium, coarse);
source of value: Derived[soil_type relation];
cardinality: single;

sp : REAL, %soil saturated percentage
NUMBER-RANGE(0,1000),
SOURCE(D.B.);
Derived[table, sp_t],
SINGLE.

sbd: REAL, %soil bulk density
NUMBER-RANGE(0,1000),
SOURCE(D.B.;
Derived[table,sbd_t])
SINGLE.
ec: numeric; % in m. mhos
source of value: DB;
cardinality: single;

calcium carbonate: numeric; % percentage
source of value: user;
cardinality: single;

Ca quantity: numeric; % in ppm
source of value: user;
cardinality: single;

N quantity: numeric;
source of value: user;
cardinality: single;

P quantity: numeric;
source of value: user;
cardinality: single;

K quantity: numeric;
source of value: user;
cardinality: single;

Mg quantity: numeric;
source of value: user;
cardinality: single;

Fe quantity: numeric;
source of value: user;
cardinality: single;

Zn quantity: numeric;
source of value: user;
cardinality: single;

Mn quantity: numeric;
source of value: user;
cardinality: single;

Cu quantity: numeric;
source of value: user;
cardinality: single;

concept organic manure;
sub-type-of : fertilizer.
properties:
name: {chicken magazer, chicken bayade, cattle, horse, pigeon};
source of value: DB;
cardinality: single;

weight: numeric; % weight of 1 cubic meter in Kg
source of value: derived;

cardinality: single;

quantity : numeric, /* the volume of organic manure in cubic meter for the whole area */

source of value: DB;
cardinality: single;

ratio of N: numeric;
source of value: derived;
cardinality: single;

ratio of P: numeric;
source of value: derived;
cardinality: single;

ratio of K: numeric;
source of value: derived;
cardinality: single;

ratio of Ca: numeric;
source of value: derived;
cardinality: single;

ratio of Mg: numeric;
source of value: derived;
cardinality: single;

concept chicken magazer;
sub-type-of: organic manure;
properties:
weight: {250};
ratio of N: {0.015};
ratio of P: {0.012};
ratio of K: {0.005};
ratio of Ca: {0.0};
ratio of Mg: {0.0};

concept chicken bayade;
sub-type-of: organic manure;
properties:
weight: {575}
ratio of N: {0.013};
ratio of P: {0.007};
ratio of K: {0.005};
ratio of Ca: {0.0};
ratio of Mg: {0.0};

concept cattle;

sub-type-of: organic manure;
properties:
weight: {700}
ratio of N: {0.005};
ratio of P: {0.0014};
ratio of K: {0.004};
ratio of Ca: {0.0};
ratio of Mg: {0.0};

concept horse;
sub-type-of: organic manure;
properties:
weight: {250}
ratio of N: {0.02};
ratio of P: {0.0066};
ratio of K: {0.016};
ratio of Ca: {0.0};
ratio of Mg: {0.0};

concept pigeon;
sub-type-of: organic manure;
properties:
weight: {300}
ratio of N: {0.04};
ratio of P: {0.036};
ratio of K: {0.026};
ratio of Ca: {0.0};
ratio of Mg: {0.0};

concept plant;
properties:
name: {tomato, cucumber, melon, rice, bean, fababean, …}; / This property will only be used when we integrate more than one crop/
source of value: user;
cardinality: single;

init_stage : INTEGER,
NUMBER-RANGE(0,1000),
SOURCE(User.; Derived[relation, growth_stages])
SINGLE.

ve_stage : INTEGER, %vegetative stage
NUMBER-RANGE(0,1000),
SOURCE(User.; Derived[relation, growth_stages])
SINGLE.
fl_stage : INTEGER, %flowering stage
NUMBER-RANGE(0,1000),
SOURCE(User.; Derived[relation, growth_stages])
SINGLE.
fr_stage : INTEGER, %fruiting stage
NUMBER-RANGE(0,1000),
SOURCE(User.; Derived[relation, growth_stages])

```
                SINGLE.
        growth_period : INTEGER,
                NUMBER-RANGE(0,1000),
                SOURCE(Derived[function, growth_period_f])
                SINGLE.
        init_ve_stage : INTEGER,
                NUMBER-RANGE(0,1000),
                SOURCE(Derived[function, init_ve_stage_f])
                SINGLE.
        init_ve_fl_stage: INTEGER,
                NUMBER-RANGE(0,1000),
                SOURCE(Derived[function, init_ve_fl_stage_f])
                SINGLE.

elements: nominal;
      source of value: derived;
      cardinality: multiple;

variety: nominal;
      source of value: derived;
      cardinality: single;

predicted yield factor: numeric;     /* a coefficient when multiply by the
                                     optimum yield gives the predicted yield */
      source of value: derived;
      cardinality: single;

N ratio: real number;
      source of value: derived;
      cardinality: single;

       P ratio:  real number;
      source of value: derived;
      cardinality: single;

K ratio: real number;
      source of value: derived;
      cardinality: single;

Ca ratio:  real number;
      source of value: derived;
      cardinality: single;

       Mg ratio: real number;
      source of value: derived;
      cardinality: single;

       Fe ratio: real number;
      source of value: derived;
      cardinality: single;

Mn ratio: real number;
```

source of value: derived;
cardinality: single;

Cu ratio: real number;
source of value: derived;
cardinality: single;

Zn ratio: real number;
source of value: derived;
cardinality: single;

N content: real number;
source of value: derived;
cardinality: single;

P content: real number;
source of value: derived;
cardinality: single;

K content: real number;
source of value: derived;
cardinality: single;

Ca content: real number;
source of value: derived;
cardinality: single;

Mg content: real number;

source of value: derived;
cardinality: single;

Fe content: real number;
source of value: derived;
cardinality: single;

Mn content: real number;
source of value: derived;
cardinality: single;

Cu content: real number;
source of value: derived;
cardinality: single;

Zn content: real number;
source of value: derived;
cardinality: single;

concept farm;
properties :

latitude: REAL
NUMBER-RANGE(0,1000)
SOURCE(D.B.)
SINGLE.

altitude : REAL
NUMBER-RANGE(0,1000)
SOURCE(D.B.)
SINGLE.
area: REAL,
NUMBER-RANGE(1,2000),
SOURCE(D.B.)
SINGLE
drainage_system: NOMINAL,
VALUE-LIST(good;medium;bad),
SOURCE(D.B.)
SINGLE
type: NOMINAL
VALUE_LIST([open_field,low_tunnel,
High_tunnel])
SOURCE(D.B.)
SINGLE.
NECESSARY.

concept tomato;
properties:
elements: {N, P, K, Ca, Mg, Fe, Zn, Cu, Mn};
variety: { al_wadey, alex_61, alex_63, astren_b, beto_86, beto_bride, brigad,
c_1_150, casel_rok, dora, extra_emar_mind,facolta_38, floraide,
'gs-12', hagen_6130, jakal, mader,mar_mind, neam_1400,
nemarok, sarea, super_astren_b, super_mar_mind, taefon, ty20,
ty70_70, ty70_84, ty_70, u_c_973, v_f_n_8, veuna, lamec,
gc_779, terkewaza, kowmat, nomy, crystena,};
source of value: DB;
cardinality: single;

N ratio: {0.05};
P ratio: {0.005]
K ratio: {0.04]
Ca ratio: {0.02}
Mg ratio: {0.005}
Fe ratio: {0.00014}
Mn ratio: {0.000145}
Cu ratio: {0.00002}
Zn ratio: {0.00004}

concept plantation;
properties:
cultivation capability: {yes, no};

source of value: drived;
cardinality: single;

optimum-yield: numeric;
source of value: drived;
cardinality: single;

expected-yield: numeric;
source of value: drived;
cardinality: single;

3.2 Common Relations

relation soil_type;
properties: texture, type;
argument-1: soil;
argument-2: soil;
axioms :

R1: If (texture of soil = "clay; clay loam; silty clay; silty clay loam"
Then
type of soil = fine

R2: If (texture of soil = "sandy clay; sandy clay loam; silt loam; silty loam"
Then
type of soil = medium

R3: If (texture of soil = "sandy loam; sand; loamy sand"
Then
type of soil = coarse

4. Irrigation System

4.1 Added Items

4.1.1 Ontology

concept : wr_total;
sub_type_of : domain_class
properties:

value : numeric;
source of value: drived;
cardinality: single;

previous_wr : numeric;

source of value: drived;
cardinality: single;

concept : wr_total_h;
sub_type_of : domain_class
properties:

wr_value : numeric;
source of value: drived;
cardinality: single;

concept : wr_total_o;
sub_type_of : domain_class
properties:

wr_value : numeric;
source of value: drived;
cardinality: single;

4.1.2 Function

1. wr_total_f = (value of water_requirement + previous_wr of wr_total)
2. wr_total_h _f = (value of wr_total * 4200)/area of farm)
3. wr_total_o_f = (value of wr_total/ area of farm)

4.2 Deleted Items

concept eggplant_family
sub-type-of (vegetable)

concept tomato
sub-type-of (eggplant_family)

concept tomato_open_field
sub-type-of (tomato)

concept tomato_early_summer
sub-type-of (tomato_open_field)

concept tomato_summer
sub-type-of (tomato_open_field)

concept tomato_autumn
sub-type-of (tomato_open_field)

concept tomato_winter
sub-type-of (tomato_open_field)

concept tomato_nile
sub-type-of (tomato_open_field)

concept beto_86
sub-type-of (tomato_early_summer)
concept u_c_973
sub-type-of (tomato_early_summer)

concept floraide
sub-type-of (tomato_early_summer)

concept alex_63
sub_type_of (tomato_early_summer)

concept beto_bride
sub_type_of (tomato_early_summer)

concept brigad
sub_type_of (tomato_early_summer)

concept mader
sub_type_of (tomato_early_summer)

concept hagen_6130
sub_type_of (tomato_early_summer)

concept v_f_n_8
sub_type_of (tomato_early_summer)

concept nemarok
sub_type_of (tomato_early_summer)

concept astren_b
sub_type_of (tomato_summer)

concept zena_692
sub_type_of (tomato_summer)

concept super_astren_b
sub_type_of (tomato_summer)

concept csael_rok
sub_type_of (tomato_autumn)

concept alex_61
sub_type_of (tomato_autumn)

concept ty70_84
sub_type_of(tomato_autumn)

concept ty70_70
sub_type_of (tomato_autumn)

concept ty20
sub_type_of (tomato_autumn)

concept facolta_38
sub_type_of (tomato_autumn)

concept veuna
sub_type_of (tomato_autumn)
concept jakal
sub_type_of (tomato_autumn)

concept e445
sub_type_of (tomato_autumn)

concept tomanour
sub_type_of (tomato_autumn)

concept sarea
sub_type_of (tomato_autumn)

concept taefon
sub_type_of (tomato_autumn)

concept dora
sub_type_of (tomato_autumn)

concept super_mar_mind
sub_type_of (tomato_winter)

concept mar_mind
sub_type_of (tomato_winter)

concept extra_emar_mind
sub_type_of (tomato_winter)

concept c_l_150
sub_type_of(tomato_winter)

concept neam_1400
sub_type_of (tomato_nile)

concept tomato_low_tunnel
sub_type_of (tomato)

concept tomato_high_tunnel
sub_type_of (tomato)

concept cherry_tomato_high_tunnel
sub_type_of (tomato)

concept al_wadey
sub_type_of (tomato_low_tunnel)

concept ty_70
sub_type_of (tomato_low_tunnel)

concept gs_12
sub_type_of (tomato_low_tunnel)

concept orient

sub_type_of (tomato_low_tunnel)

concept model
sub_type_of (tomato_low_tunnel)
concept lamec
sub_type_of (tomato_high_tunnel)

concept gc_779
sub_type_of (tomato_high_tunnel)

concept terkewaza
sub_type_of (tomato_high_tunnel)

concept kowmat
sub_type_of (tomato_high_tunnel)

concept nomy
sub_type_of (tomato_high_tunnel)

concept sweet_million
sub_type_of (cherry_tomato_high_tunnel)

concept pink_debut
sub_type_of (cherry_tomato_high_tunnel)

concept yellow_debut
sub_type_of (cherry_tomato_high_tunnel)

4.3 Updating legal values

Old Item			New Item		
Concept	**Property**	**Value**	**Concept**	**Property**	**Value**
Irrigation	Method	VALUE-LIST(drip; flooding)	irrigation	method	VALUE-LIST (dripping; flooding)
Organic_ manure	name	VALUE-LIST(cattle, hog, chicken, chicken bayade, chicken magazer, sheep, hourse, pigeon)	organic_ manure	name	VALUE-LIST (chicken magazer, chicken bayade, cattle, horse, pigeon)
Vegetable	variety	al_wadey, alex_61, alex_63, astren_b, beto_86, beto_bride, brigad, c_1_150, casel_rok,	tomato	variety	al_wadey, alex_61, alex_63, astren_b, beto_86, beto_bride, brigad, c_1_150, casel_rok, ,dora,

		dora, e445, crystena extra_emar_min d, facolta_38, floraide, gc_779, 'gs-12', hagen_6130, jakal, kowmat, lamec, mader, mar_mind, model, neam_1400, nemarok, orient, nomy, pink_debut, sarea, super_astren_b, sweet_million, super_mar_mind ,taefon, terkewaza, tomanour, ty20, ty70_70, ty70_84, ty_70, u_c_973, v_f_n_8, yellow_debut, veuna, 'zena-692'			extra_emar_min d,facolta_38, floraide, 'gs-12', hagen_6130, jakal, mader,mar_mind , neam_1400, nemarok, sarea,super_astre n_b, super_mar_mind , taefon, ty20, ty70_70, ty70_84, ty_70, u_c_973, v_f_n_8, veuna, lamec, gc_779, terkewaza, kowmat, nomy, crystena,

5. Fertilization System

5.1 Updating legal values

Old Item			New Item		
Concept	**Property**	**Value**	**Concept**	**Property**	**Value**
Organic_ manure	name	chicken manure for meat product, chicken manure	organic_ manure	name	chicken magazer, chicken bayade, cattle, horse, pigeon

		for egg product, cow manure, residual farm manure, horse manure, sewage sludge manure, town refuse manure, pigeon manure			
Soil	type	clayey, loamy, sandy	soil	type	fine, medium, coarse
Plantation	type	Open field, tunnels, low tunnels	farm	type	open_field, high_tunnel low_tunnel,
Plantation	irrigation type	flooding, dripping, sprinkling, pivot	irrigation	method	flooding, dripping
Tomato	variety	Peto 86, UC 97-3, Cast rock, Floradade, Alex 63, Peto pride, Prigade, Madir, H-6130, Estrine-B, Super estrine-B, Alex 61, TY, Facolta-38, Fiona, Jacal, Saria, Taifon, Dora, Super marmend, Marmend, Extra marmend, H-CL-150, CL-150, Valley, Ben Shifar, Orite, H-5656, H-G-S-12, V.F.N-8, H-Nema Rock, H-Nema-1400,	tomato	variety	al_wadey, alex_61, alex_63, astren_b, beto_86, beto_bride, brigad, c_1_150, casel_rok, ,dora, extra_emar_mind,facolta_38, floraide, 'gs-12', hagen_6130, jakal, mader,mar_mind, neam_1400, nemarok, sarea,super_astren_b, super_mar_mind, taefon, ty20, ty70_70, ty70_84, ty_70, u_c_973, v_f_n_8, veuna, lamec, gc_779, terkewaza, kowmat, nomy, crystena,

		under tunnel variety, other			

5.2 Updating properties

Organic manure: volume → organic manure: quantity
Plantation: date → current_planting: date
Plantation: type → farm: type
Soil: salinity → Soil: ec
Water: salinity → Water: eciw

5.3 Deleted concepts

residual farm manure
town refuse manure
sewage sludge manure

5.4 Renamed concepts

chicken manure for meat product → chicken magazer
chicken manure for egg product → chicken bayade
cow manure → cattle
horse manure → horse
pigeon manure → pigeon

5.5 Updating rules

The following rules in the technical report number TR/CLAES/140/2000.5 pages 23, 24 have been updated, the bolded items are added, the italic are modified to the items follows the arrow. These updating is due to that the volume of organic manure is in cubic meter for the whole farm area in the irrigation system, and also the farm area is in square meter in irrigation system while the volume of organic manure is in cubic meter per feddan in fertilization system.

(plant: name = X &
X: element = Ca &
soil: Ca quantity = SCaQ &
water: Ca quantity = WcaQ &
water: qty = WQ &
organic manure: name = M &
farm: area = A &
M: ratio of Ca = MRCa &
M: volume = MV & → M: quantity = MQ
M: weight = MW)
CALCULATE ELEMENT IN ENVIRONMENT
(environment: Ca quantity = SCaQ + WcaQ * WQ/1000 + MRCa * **(MQ*4200/A)** * MW)

(plant: name = X &
X: element = N &
soil: N quantity = SNQ &
water: N quantity = WNQ &

water: qty = WQ &
organic manure: name = M &
farm: area = A &
M: ratio of N = MRN &
M: volume = MV & → M: quantity = MQ
M: weight = MW)
CALCULATE ELEMENT IN ENVIRONMENT
(environment: N quantity = SNQ + WNQ * WQ/1000 + MRN * **(MQ*4200/A)** * MW)

(plant: name = X &
X: element = P &
soil: P quantity = SPQ &
water: P quantity = WPQ &
water: qty = WQ &
organic manure: name = M &
farm: area = A &
M: ratio of P = MRP &
M: volume = MV & → M: quantity = MQ
M: weight = MW)
CALCULATE ELEMENT IN ENVIRONMENT
(environment: P quantity = SPQ + WPQ * WQ/1000 + MRP * **(MQ*4200/A)** * MW)

(plant: name = X &
X: element = K &
soil: K quantity = SKQ &
water: K quantity = WKQ &
water: qty = WQ &
organic manure: name = M &
farm: area = A &
M: ratio of K = MRK &
M: volume = MV & → M: quantity = MQ
M: weight = MW)
CALCULATE ELEMENT IN ENVIRONMENT
(environment: K quantity = SKQ + WKQ * WQ/1000 + MRK * **(MQ*4200/A)** * MW)

(plant: name = X &
X: element = Mg &
soil: Mg quantity = SMgQ &
water: Mg quantity = WMgQ &
water: qty = WQ &
organic manure: name = M &
farm: area = A &
M: ratio of Mg = MRMg &
M: volume = MV & → M: quantity = MQ
M: weight = MW)
CALCULATE ELEMENT IN ENVIRONMENT

(environment: Mg quantity = SMgQ + WMgQ * WQ/1000 + MRMg * **(MQ*4200/A)** * MW)

6. Global Interface

The main screen is shown in figure(3). About system button display a brief description about the expert system as follows: " This expert system contains two subsystems, fertilization and irrigation for tomato plant cultivated in open field, under low tunnels or high tunnels. When clicking on the "participants" button a pull down menu appears which contains the items [Agricultural Experts, Claes Staff]. When choosing either of them, a second screen appears containing the persons name according to the item selected.

Irrigation button invokes the irrigation system. Fertilization button invokes the fertilization system. Database buttons invokes the database.

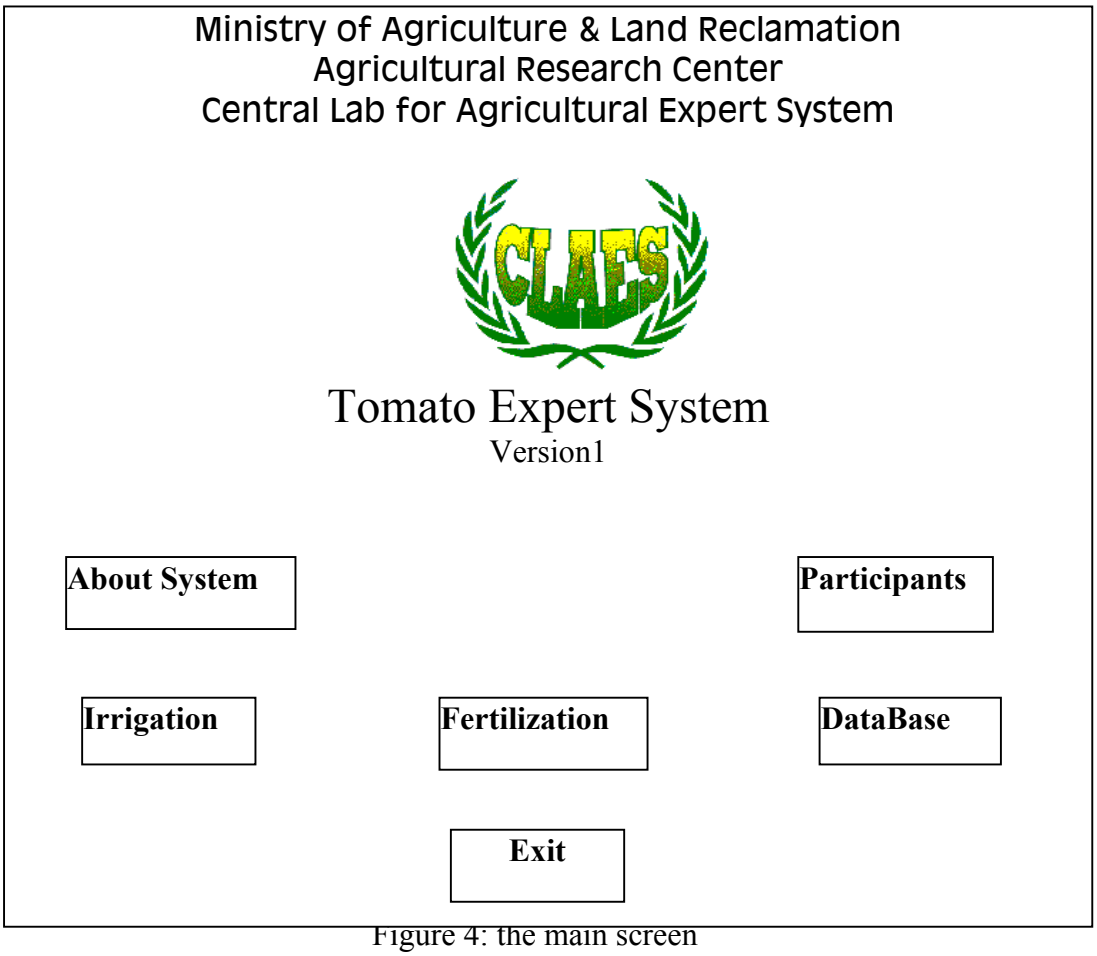

Figure 4: the main screen

7. Overall Control

The system overall control in the sequence diagram shown in figure 4.

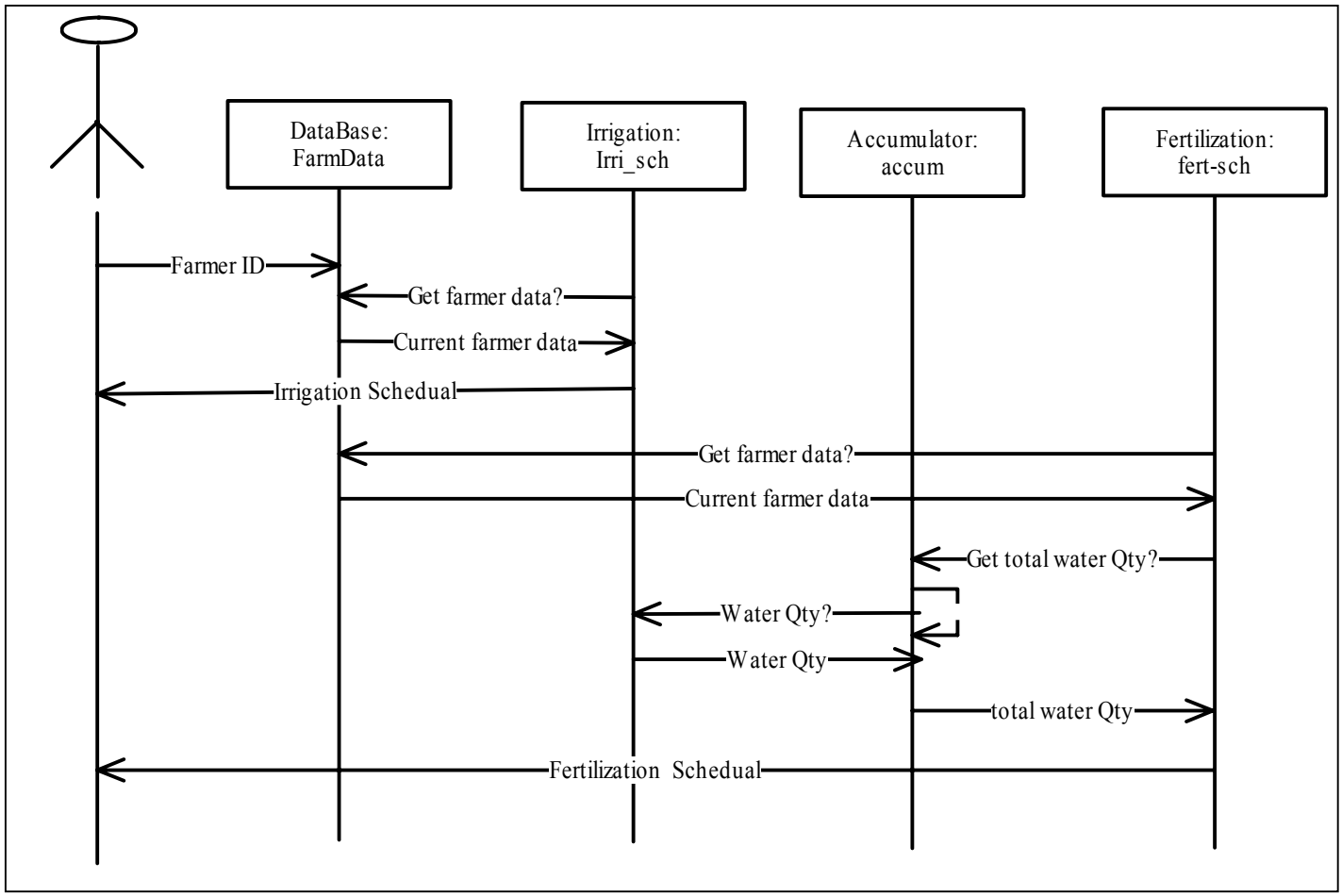

Figure 4: Fertigation Sequence Diagram

8. Testing Method

The integrated Fertigation system should run all the test cases provided for both standalone irrigation and fertilization systems and provide he same results approximately.