

# Design of Disorder Treatment Expert System For Melon Integrated with Disorder Diagnosis

## 1. Introduction

This document contains the design of disorder treatment expert system for melon integrated with the disorder diagnosis (technical report TR/CLAES/155/2000.8). Section 2 represents the common knowledge base. Section 3 represents the modifications of the diagnosis expert system. Sections 4, 5, 6, and 7 represent treatment, user interface, overall control, and test cases respectively.

## 2. Common Knowledge

### 2.1. Domain Ontology

**Concept** disorders;  
**properties:**

value: disorder instances  
source of value: driven;  
cardinality: multiple;

suspected: disorder instances  
source of value: driven;  
cardinality: multiple;

**concept** plantation;  
**properties:**

date: date;  
source of value: user;  
cardinality: single;  
prompt:

type: { , }  
source of value: user;  
cardinality: single;  
prompt:

**concept** plant;  
**properties:**

complaint: universal;

observation: universal;

age: numeric  
source of value: driven;  
cardinality: single;

appearance: { }  
**differentiation-of** observation (plant);  
source of value: user;  
cardinality: single;

**concept** ;  
**sub-type-of:** disorders;

**concept** ;  
**sub-type-of:** disorders;

**concept** ;  
**sub-type-of:** disorders;  
**properties:**  
: numeric;  
source of value: user;  
cardinality: single;  
: numeric;  
source of value: user;  
cardinality: single;

**concept** ;  
**sub-type-of:** disorders;

**concept** ;  
**sub-type-of:** disorders;

**concept** ;  
**sub-type-of:** disorders;

**concept** ;  
**sub-type-of:** disorders;  
**properties:**  
: { };  
source of value: driven;  
cardinality: single;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of:** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;  
**properties:**

: numeric;

source of value: user;  
cardinality: single;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;  
**properties:**  
: numeric;  
source of value: user;  
cardinality: single;  
prompt:

**instance** ;  
**sub-type-of** ;  
**properties:**  
: numeric;  
source of value: user;  
cardinality: single;  
prompt:

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

**instance** ;  
**sub-type-of** ;

```

instance ;
  sub-type-of ;

instance ;
  sub-type-of ;

instance ;
  sub-type-of ;

instance ;
  sub-type-of ;

instance ;
  sub-type-of ;

instance ;
  sub-type-of ;

instance ;
  sub-type-of ;

```

### 3. Diagnosis Subsystem

This section contains the modifications of the technical report TR/CLAES/155/2000.8 due to the integration process and due to the comments in the technical report TR/CLAES/200/2001.2

#### 3.1 Domain Knowledge

##### 3.1.1 Ontology

- The concepts "plant", "plantation", and "disorders" are deleted because they become common concepts in section 2.1.1.
- The cardinality of the following properties is modified to multiple instead of single.

larva: multiple, root: appearance, root: color, plant: appearance,  
 leaves: spots\_appearance, leaves: spots\_position, leaves: color\_direction,  
 leaves: spots\_color, leaves: tunnel\_color, leaves: color\_position, leaves: color,  
 soil: appearance, and fruits: appearance

- The following properties of the "leaves" concept in page 4 of technical report TR/CLAES/155/2000.8 have been updated as follows:

```
spots color: {
    appearance: {
        ,
    }
};
```

- The typing error of the property "fruit: ppearance" in page 5 of technical report TR/CLAES/155/2000.8 has been updated to "fruit: appearance"

### 3.1.2 Domain Models

- The suspected disorder, in rule 1 page 7 of technical report TR/CLAES/155/2000.8, "disorders: suspected = " is duplicated, so one of them is deleted.
  - The property "leaves: appearance status" in pages 8, 11, 14, and 15 has been updated to "leaves: appearance\_status"
  - The property " stem: spot color " in pages 10, and 11 has been updated to " stem: spots color "
  - The property " stem: spot appearance " in pages 10, and 11 has been updated to " stem: spots appearance "
  - The property " stem spot: color " in page 11 has been updated to " stem: spots color "
  - The property " leaves tunnel: color " in page 15 has been updated to " leaves: tunnel color "
  - The value " stem: color = " in page 11 has been updated to " stem: color = "

- The value " leaves: color position = " in pages 12, and 13 has been updated to " leaves: color position = "
- The parentheses of rule 4 in page 9 has been corrected as follows:

```
(disorders: suspected = &
(leaves: spots color = OR
leaves: spots color = ) &
leaves: spots position = &
stem: color = &
(stem: spots color = OR
stem: spots color = ))
CONFIRM
(disorders: value = )
```

- The parentheses of rule 2 in page 10 has been corrected as follows:

```
(disorders: suspected = &
((leaves: spots color = &
leaves: spots position = &
leaves: spots appearance = ) OR
(leaves: spots color = &
leaves: spots position = &
leaves: spots appearance = )) &
(stem: color = OR
stem: color = ))
CONFIRM
(disorders: value = )
```

- The parentheses of rule 4 in page 14 has been corrected as follows:

```
(disorders: suspected = &
leaves: appearance status =
leaves: appearance = &
(leaves: appearance = OR
leaves: appearance = ))
CONFIRM
(disorders: value = )
```

- The parentheses of rule 2 in page 16 has been corrected as follows:

```
(disorders: suspected = &
(fruits: appearance = OR
```

```

fruits: appearance =
CONFIRM
(disorders: value =
)

```

- The parentheses of rule 5 in page 12 has been corrected as follows:

```

(disorders: suspected =
&
leaves: color position =
&
((leaves: appearance_status =
&
leaves: appearance =
) OR
(leaves: appearance_status =
&
leaves: color direction =
)))
CONFIRM
(disorders: value =
)

```

- Rule 1 page 10 has been updated as follows:

```

(disorders: suspected =
&
leaves: spots color =
&
leaves: spots appearance =
&
leaves: spots position =
&
leaves: appearance status =
&
leaves: appearance =
)
CONFIRM
(disorders: value =
)

```

### 3.2 Inference Knowledge

The typing errors of the specification of inference steps predict, differentiate, and Generate observations in page 18 of technical report TR/CLAES/155/2000.8 are corrected as follows:

**inference:** predict

**operation-type:** predicts the suspected disorders.

**input-roles:** complaints, plant age.

**output-roles:** suspected disorders.

**static-roles:** SUSPECT  $\in$  prediction-model.

**spec:** the suspected disorders are predicted according to the complaints by applying " SUSPECT " relation.

**inference:** differentiate

**operation-type:** differentiate the confirmed disorders.

**input-roles:** suspected disorders, observations, plant age.

**output-roles:** confirmed disorders.

**static-roles:** CONFIRM  $\in$  differentiation-model.

**spec:** the confirmed disorders are obtained from the suspected disorders by obtaining observations on the plant from the user and by applying



“CONFIRM” relation.

**inference:** Generate observations

**operation-type:** generate observations related to the suspected disorders.

**input-roles:** suspected disorders.

**output-roles:** observations.

**static-roles:** CONFIRM  $\in$  differentiation-model

**spec:** the required observations are generated, to ask user about, by using “CONFIRM” relation. The generated observations are the L.H.S. of the rules that their R.H.S. are the suspected disorders.

### 3.3 Task Knowledge

The typing error in page 18 of technical report TR/CLAES/155/2000.8 is corrected as follows:

**task:** melon disorders diagnosis;

## 4. Treatment Subsystem

This expert system provides the user with the treatment of identified disorders, which are either the output of the diagnosis subsystem or the user assumption. Design of treatment expert system consists of 3 parts, domain knowledge, inference knowledge, and task knowledge. CommonKADS methodology [Wielinga, 1994] is used to represent these knowledge. Sections 4.1, 4.2, 4.3, and 4.4 represent domain, inference, task knowledge, and user interface respectively.

### 4.1 Domain Knowledge

#### 4.1.1 Domain Ontology

**concept** pesticide;

**properties:**

application\_method: universal

source of value: derived;

cardinality: single;

treat\_disorder: the disorders treated by the pesticide in the current situation;

source of value: derived;

cardinality: multiple;

concentration: universal;

source of value: derived;

cardinality: multiple;

**concept** fertilizer;

**properties:**

application\_method: universal  
source of value: derived;  
cardinality: single;

treat\_disorder: the disorders treated by the pesticide in the current situation;  
source of value: derived;  
cardinality: multiple;

concentration: universal;  
source of value: derived;  
cardinality: multiple;

**concept** irrigation;

**properties:**

type: {                    }  
source of value: user;  
cardinality: single;

**concept**                    ;  
**sub-type-of:** pesticide;

**concept**                    ;  
**sub-type-of:** pesticide;

**concept**                    ;  
**sub-type-of:** pesticide;

**concept**                    ;  
**sub-type-of:** pesticide

**instance**                    +                    ;  
**sub-type-of**                    ;  
**property-value:**  
concentration =                    100/                    150 +                    100  
application\_method =

**instance**                    ;  
**sub-type-of**                    ;  
**property-value:**  
concentration =                    100/                    100  
application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of**        : <sub>1</sub>  
  **property-value:** concentration =     100/ 250  
                          application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of**        : <sub>1</sub>  
  **property-value:** concentration =     100/ 250  
                          application\_method =

**instance**   +        : <sub>1</sub>  
  **sub-type-of**        : <sub>1</sub>  
  **property-value:** concentration =     100/     150 +     100  
                          application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of**        : <sub>1</sub>  
  **property-value:** concentration =     100/ 250  
                          application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of**        : <sub>1</sub>  
  **property-value:** concentration =     100/ 500  
                          application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of** fertilizer; <sub>1</sub>  
  **property-value:** concentration =     100/ 200  
                          application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of** fertilizer; <sub>1</sub>  
  **property-value:** concentration =     100/ 200  
                          application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of** fertilizer; <sub>1</sub>  
  **property-value:** concentration =     100/ 200  
                          application\_method =

**instance**            : <sub>1</sub>  
  **sub-type-of** fertilizer; <sub>1</sub>  
  **property-value:** concentration =     100/ 200

application\_method =

**instance**            :  
  **sub-type-of** fertilizer;  
  **property-value:** concentration =     / 400  
                  application\_method =

**instance**            :  
  **sub-type-of** fertilizer;  
  **property-value:** concentration =     / 50  
                  application\_method =

**instance**            :  
  **sub-type-of** fertilizer;  
  **property-value:** concentration =     100/ 200  
                  application\_method =

**instance**            :  
  **sub-type-of** fertilizer;  
  **property-value:** concentration =     100/ 200  
                  application\_method =

**instance**            :  
  **sub-type-of** fertilizer;  
  **property-value:** concentration =     100/ 200  
                  application\_method =

**instance**        % 95        :  
  **sub-type-of**            :  
  **property-value:**        100/ 1  
                  application\_method =

**instance**        + %40  
  **sub-type-of**            :  
  **property-value:** concentration =     / 25-20 + 1,25  
                  application\_method =

**instance**        + %40  
  **sub-type-of**            :  
  **property-value:** concentration =     / 20-15 + 1,25  
                  application\_method =

**instance** %57  
**sub-type-of** ;  
**property-value:** 100/ 312  
application\_method =

**instance** %94  
**sub-type-of** ;  
**property-value:** 100/ 1  
application\_method =

**instance** % 93 ;  
**sub-type-of** ;  
**property-value:** 100/ 625  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** 100/ 60  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** 100/ 100  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** 100/ 100  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** 100/ 100  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** 100/ 500  
application\_method =

**instance** %49 - ;

```

sub-type-of          ;
property-value: concentration =      100/  387
      application_method =

instance           ;
sub-type-of          ;
property-value: concentration =      100/  100
      application_method =

instance %10        ;
sub-type-of          ;
property-value: concentration =      /  13
      application_method =                                10

instance %10        ;
sub-type-of          ;
property-value: concentration =      /  13
      application_method =                                10

instance wp %50     ;
sub-type-of          ;
property-value: concentration =      100/  150
      application_method =

instance wp %50     ;
sub-type-of          ;
property-value: concentration =      100/  75
      application_method =

instance EC %50     ;
sub-type-of          ;
property-value: concentration =      100/  125
      application_method =

instance SP %90     ;
sub-type-of          ;
property-value: concentration =      100/  75
      application_method =

instance 2x        ;

```

**sub-type-of** ;  
**property-value:** concentration = 100/ 50  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 50  
application\_method =

**instance** WP % 25 ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 200  
application\_method =

**instance** EC %50 ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 387  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 250  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 10  
application\_method =

**instance** ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 100  
application\_method =

**instance** - ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 250  
application\_method =

**instance** + ;  
**sub-type-of** ;  
**property-value:** concentration = 100/ 250  
application\_method =

**instance**           % 95            ;  
**sub-type-of**            ;  
**property-value:**       100/   1  
                  application\_method =

**attribute** advice  
**value-set** universal;  
**cardinality:** multiple;

**relation:** treat;  
**inverse:** treated-by;  
**argument-1:** pesticide;  
          **argument-role:** remedy;  
**argument-2:** disorder;  
          **argument-role:** cause;

**relation:** recommend;  
**inverse:** is-recommended-when;  
**argument-1:** disorder;  
          **argument-role:** cause;  
**argument-2:** advice;  
          **argument-role:** remedial-advice;

## 4.1.2 Domain Models

**domain-model:** treatment-model;  
**parts:** tuple(treat), tuple(treated-by);  
**axioms:**           (           +           )  
                                  TREAT  
                                  (           &           )  
                                  (           )  
                                  TREAT  
                                  (           &           )  
                                  (           )  
                                  TREAT  
                                  (           &           &           &           &  
                                  &           &           )  
                                  (           )  
                                  TREAT  
                                  (           &           &           &           )



( ( + )  
TREAT  
& & &  
)

( ( )  
TREAT  
& & & &  
)

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( + %40 )  
TREAT  
( & )

( + %40 )  
TREAT  
( )

(%57 )  
TREAT  
( )

(%94 )  
TREAT  
( & & )

( % 95 )  
TREAT  
( & & )

( % 95 )  
TREAT  
( & & )

( %49 - )  
TREAT  
( & & )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( % 93 )  
TREAT  
( & & )

( )  
TREAT  
( & )

(%10 )  
TREAT  
( )

(%10 )  
TREAT  
( )

(wp %50 )  
TREAT  
( )

(wp %50 )  
TREAT  
( & & & )

(**EC** %50 )  
TREAT  
( & & )

(**SP** %90 )  
TREAT  
( )

(2x )  
TREAT  
( )

( )  
TREAT  
( )

(WP % 25 )  
TREAT  
( & )

(EC %50 )  
TREAT  
( & )

( )  
TREAT  
( )

( )  
TREAT  
( )

( )  
TREAT  
( )

( - )  
TREAT  
( )

(+ )  
TREAT  
( )

( % 95 )  
TREAT  
( & & )

( OR )  
TREATED-BY  
( + , , )

( &not( ) )  
TREATED-BY  
( , )

( & )  
TREATED-BY  
( )

not( ( ( OR OR OR ) & not( ) ) )  
TREATED-BY  
( , , + , )

( & not( ) )  
TREATED-BY  
( , + )

( )  
TREATED-BY  
( )

( )  
TREATED-BY  
(+ - )

( & plant:age < 30)  
TREATED-BY  
( )

( )  
TREATED-BY  
( )

( )  
TREATED-BY  
( )

( )  
TREATED-BY  
( )

( )  
TREATED-BY  
( )

( )  
TREATED-BY  
( )

```

( )
TREATED-BY
( )

( )
TREATED-BY
( )

( & irrigation: type = )
TREATED-BY
( )

( & irrigation: type <> )
TREATED-BY
( )

( & : = )
TREATED-BY
% 93 % 95 % 95 )
( % 95 %94 %49 -

( & : = )
TREATED-BY
( )

( & not( ) )
TREATED-BY
( + %40 )

( )
TREATED-BY
( + %40 )

( )
TREATED-BY
(%10 %10 )

( &
: 5 < &
not( ) )
TREATED-BY
(wp %50 wp % 50 )

( &
: 5 =< )

```

TREATED-BY  
(wp % 50 )

( & (( & : >= 5) OR  
( & : >= 5 )))

TREATED-BY  
(EC %50 )

( &  
: 5 < &  
(( & : >= 2 &  
: < 5 ) OR ( &  
: >= 2 &  
: < 5 )))

TREATED-BY  
( EC %50 )

( &  
: 5 < &  
(not( ) OR ( & : < 2))  
(not( ) OR ( &  
: < 2)))

TREATED-BY  
(2x SP%90 EC%50 )

( &(not( ) OR  
( & : < 2)) &  
: >= 2 & : < 5 &  
(not( ) OR ( &  
: 5 => )) &  
(not( ) OR( & : = )))

TREATED-BY  
( % 95 % 95 % 95  
% 93 %94  
%49 - )

( & : >= 2 &  
: < 5 &  
(not( ) OR ( & : < 5)) &  
(not( ) OR ( &  
: 5 => ))&  
(not( ) OR( & : = )))  
TREATED-BY

```

( % 95 % 95 % 95 %94
  % 93 %49 - )
      ( & not( ) & not( ) &
        : >= 5)
      TREATED-BY
(EC %50 ,EC %50 %57 WP % 25 , wp %50 )
      ( & not( )&
        : >= 5)
      TREATED-BY
( EC %50 , EC %50 ,wp % 25 ,WP %50 )
      ( & not( )&
        : < 5 &
        & : 5 =< )
      TREATED-BY
(EC %50 EC %50 , WP % 25 , wp %50 )

```

**domain-model:** advice-model;  
**parts:** tuple(recommend); : tuple(is-recommended-when)  
**axioms:** (advice = )

```

      IS-RECOMMENDED-WHEN
      ( & )
(advice = 21 )
      IS-RECOMMENDED-WHEN
      ( )
(advice = )
      IS-RECOMMENDED-WHEN
      ( )
(advice = %25 )
      IS-RECOMMENDED-WHEN
      ( , )
(advice = )
      IS-RECOMMENDED-WHEN
      ( )
(advice = %25 )
      IS-RECOMMENDED-WHEN
      ( )

```



(advice = / 50 - 20 )  
IS-RECOMMENDED-WHEN  
( )

(advice = 200 )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

(advice = )  
IS-RECOMMENDED-WHEN  
( )

```

(advice =
    IS-RECOMMENDED-WHEN
    (
)

(advice =
)
    IS-RECOMMENDED-WHEN
    (
    (advice =
    )
    IS-RECOMMENDED-WHEN
    (
)

(advice =
)
    IS-RECOMMENDED-WHEN
    (
)

(advice =
    %25
)
    IS-RECOMMENDED-WHEN
    (
)

(advice =
)
    IS-RECOMMENDED-WHEN
    (
)

    (
    )
    RECOMMEND
(advice =
)

    (
    &
    plantation: type =
    )
    RECOMMEND
(advice =
    21
)

    (
    )
    RECOMMEND
(advice =
    &
    advice =
)

    (
    & plantation: type =
    )
    RECOMMEND
(advice =
    %25
)

    (
    & plantation: type =
    )

```

```

      RECOMMEND
(advice =
      )

      & plantation: type =
      RECOMMEND
(advice =
      %25
      )

      & plantation: type =
      RECOMMEND
(advice =
      )

      )
      RECOMMEND
(advice =
      %25
      )

      & plantation: type =
      RECOMMEND
(advice =
      / 50 - 20
      )

      & plantation: type =
      RECOMMEND
(advice =
      200
      )

      )
      RECOMMEND
(advice =
      &
      advice =
      )

      & : = & plant: age > 60)
      RECOMMEND
(advice =
      &
      advice =
      )

      )
      RECOMMEND
(advice =
      )

      &
      : 5 =>
      RECOMMEND
(advice =
      )

      &
      : 5 >
      RECOMMEND

```

(advice = )  
 ( &  
 : 5 =< )  
 RECOMMEND  
 (advice = )

( &  
 : <= 5 &  
 (not( ) OR ( & : < 2)) &  
 (not( ) OR ( &  
 : < 2)))  
 RECOMMEND  
 (advice = )

( & : < 2 &  
 (not( ) OR ( &  
 : < 2)) &  
 (not( ) OR ( &  
 : 5 => )  
 RECOMMEND  
 (advice = )

( & : < 2 &  
 (not( ) OR ( &  
 : < 2)) &  
 (not( ) OR ( &  
 : 5 => )  
 RECOMMEND  
 (advice = )

( )  
 RECOMMEND  
 (advice = )

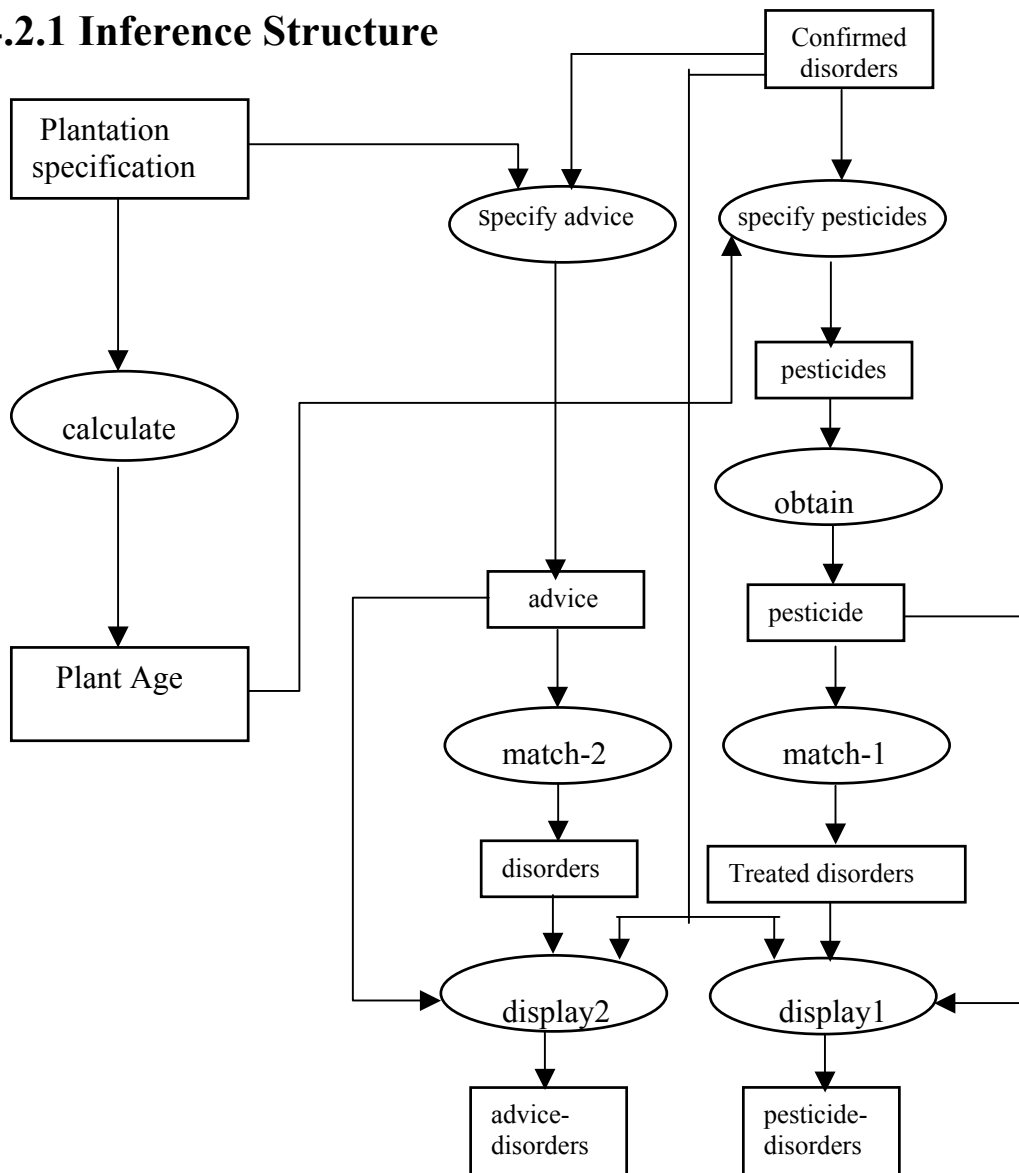
( & irrigation: type = )  
 RECOMMEND  
 (advice = )

( & irrigation: type = )  
 RECOMMEND

(advice =  
)

## 4.2. Inference Knowledge

### 4.2.1 Inference Structure



### 4.2.2 Inference specification

**inference:** calculate

**operation-type:** calculate the plant age.

**input-roles:** plantation specification . % plantation: date, plantation:type

**output-roles:** plant age % plant: age.

**static-roles:** there is no static roles

**spec:** plant age is calculated in days by subtracting the plantation date from the session date.

**inference:** specify-pesticides

**operation-type:** specifies pesticides suitable for the existing disorders

**input-roles:** confirmed disorder, /\* the output of the diagnosis system or the user assumption\*/

plant: age  
**output-roles:** pesticides % *pesticide* concept  
**static-roles:** treated\_by  $\in$  treatment-model  
**spec:** a disorder is treated by a pesticide, or alternatives. The pesticides are to be specified by applying TREATED\_BY relation.

**inference:** obtain  
**operation-type:** it's a transfer task to obtain a pesticide  
**input-roles:** pesticides  
**output-roles:** pesticide  
**static-roles:** no static role  
**spec:** a disorder is treated by a pesticide, or alternatives. In the case of alternatives, the user selects one of them.  
formally:  
 $\forall (p): p \in \text{pesticides}$   
IF no-of-elements(p) > 1  
THEN  
BEGIN  
ask-user-to-select-one-pesticide(p, p'),  
obtain(p, p')  
END  
ELSE obtain(p, p)

**inference:** specify-advice  
**operation-type:** specifies advice suitable for the existing disorders  
**input-roles:** confirmed disorders, plantation specification  
**output-roles:** advice  
**static-roles:** recommend  $\in$  advice-model  
**spec:** the user is recommended by an advice if a disorder exists. The advice is to be specified by applying RECOMMEND relation.

**inference:** match-1  
**operation-type:** matches a pesticide to the corresponding disorders  
**input-roles:** pesticide  
**output-roles:** treated disorders  
**static-roles:** treat  $\in$  treatment-model  
**spec:** a pesticide treats a set of disorders. The disorders are to be matched to a pesticide by applying TREAT relation using rule interpreter .

**inference:** match-2  
**operation-type:** matches an advice to the corresponding disorders  
**input-roles:** advice  
**output-roles:** disorders  
**static-roles:** is-recommended-when  $\in$  advice-model  
**spec:** an advice is recommended when there are a set of disorders. The disorders are to be matched to an advice by applying IS-RECOMMENDED-WHEN relation using rule interpreter.

**inference:** display-1

**operation-type:** transfer task that display a pesticide and its treated disorders in the current situation

**input-roles:** confirmed disorders, treated disorders, pesticide

**output-roles:** pesticide-disorders

**static-roles:** no static role

**spec:** a pesticide treats a set of disorders.

formally:

[ $d' = d \cap d''$  % where  $d''$  is the disorders in the 'confirmed disorders' role  
% and  $d$  is the disorders in the 'treated disorders' role

assert( $d'$ ,  $p'.treat\_disorder$ ) % where  $p'$  is the pesticide in the  
%'pesticide' role

PRESENT ( $p'$ ,  $p'.concentration$ ,  $p'.application\_method$ ,  
 $p'.treat\_disorder$ )

display-1(( $d$ ,  $d''$ ,  $p'$ ), PRESENT( ))]

**inference:** display-2

**operation-type:** matches an advice to the corresponding disorders

**input-roles:** confirmed disorder, advice, disorders

**output-roles:** advice-disorders

**static-roles:** is-recommended-when  $\in$  advice-model

**spec:** an advice is recommended when there are a set of disorders.

formally:

[ $d' = d \cap d''$ , % where  $d''$  is the disorders in the 'confirmed disorders' role  
% and  $d$  is the disorders in the 'disorders' role

PRESENT( $a$ ,  $d'$ ) % where "a" is the advice in the 'advice' role

display-2(( $a$ ,  $d$ ,  $d''$ ), PRESENT( ))]

## 4.3 Task Knowledge

**task:** melon disorder treatment;

**task-definition:**

**goal:** get treatment and recommendation of the infested plant;

**input:** confirmed disorders: the disorders infected the plant. It is either the output of the diagnostic system or the user assumption;

plantation specification: plantation: date, plantation:type;

**output:** pesticide-disorders: the disorders treated by a pesticide;

advice-disorders: the recommendation corresponding to the infected disorders;



**task-body:****type:** composite**subtasks:** calculate, specify pesticides, obtain, specify advice, match-1, match-2, display-1, display-1;**additional-roles:** pesticides: the all alternative pesticides used in a disorder treatment;

pesticide: a pesticide used in a disorder treatment;

advice: a recommendation needed for a treatment operation;

treated-disorders: the disorders treated by a pesticide;

disorders: the disorders related to a specific advice;

plant age: the plant age at the session date;

**control-structure:**

specify-pesticides(d: confirmed disorders → p:pesticides)

obtain( p → p': pesticide)

specify-advice(d → a:advice)

 $\forall (e) : e \in p' \text{ DO}$ 

Begin

match-1(e → d': treated-disorders)

display-1(e, d', d → pd: pesticide-disorders)

End

 $\forall (ad) : ad \in a \text{ DO}$ 

Begin

match-2(ad → ds: disorders)

display-2(ad, ds, d → add: advice-disorders)

End

## 4.4. User Interface

### 4.4.1. Input

If the treatment of the confirmed disorder has alternative pesticides, the system asks the user to select one by displaying the list of alternatives as shown in following screen.

The screenshot shows a user interface window. At the top, there is a label 'المواد:' (Al-Mawad) followed by a colon. Below this is a rectangular list box with a grey header bar and a white body. In the bottom-left corner of the window, there is a square button containing the Arabic word 'تم' (Tam), which means 'Done' or 'Finish'.

## 4.4.2. Output

The output of the treatment system consists of two screens. The first screen (treatment screen) shows the pesticides used and their concentrations, application methods, and the disorders in the current situation that treated by each pesticide. The pesticides are collected in a list box, when the user click on a pesticide, its concentrations, application method, and the disorders in the current situation that treated by this pesticide are displayed in entry boxes as shown in the following figure.

**نظام العلاج**

الآفات المعالجة	التركيز	المواد
	%0.25	
	الطريقة	
جلسة أخرى		النصائح

The second screen shows the advice related to each disorder (if any) as shown in the following figure. The disorders appear in the entry box in the left side are related to the highlight advice in the list box in the right side.

الآفات

1. جلسة أخرى	خروج

## 5. Interface

The Interface contains two types of screens: introductory screens and subsystems screens. The main screen of the introductory screens is shown in figure 2.



Fig.(2): introductory screen

"About system" button activates the screen in figure 3. "Domain Expert" button activates the screen that contains the names of the domain experts participated in that system, figure 4 shows this screen. "CLAES" Staff button activates the screen that contains the names of the CLAES staff participated in that system, figure 5 shows this screen. "Diagnosis" button activates the diagnosis subsystem. "Treatment" button activates the treatment subsystem.

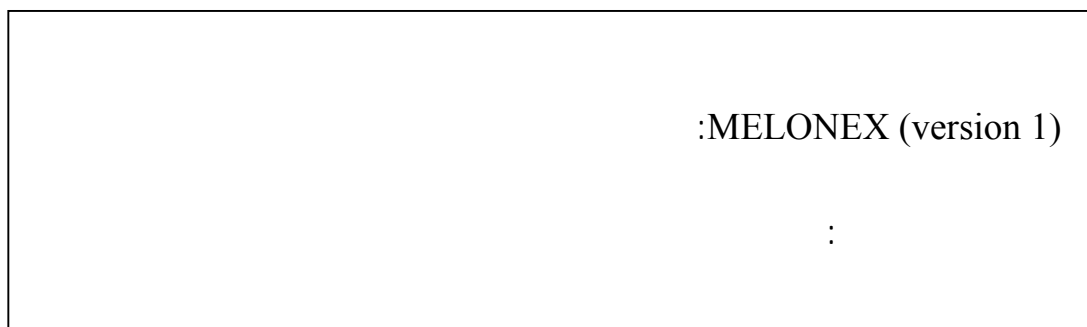


Fig.(3): about system screen

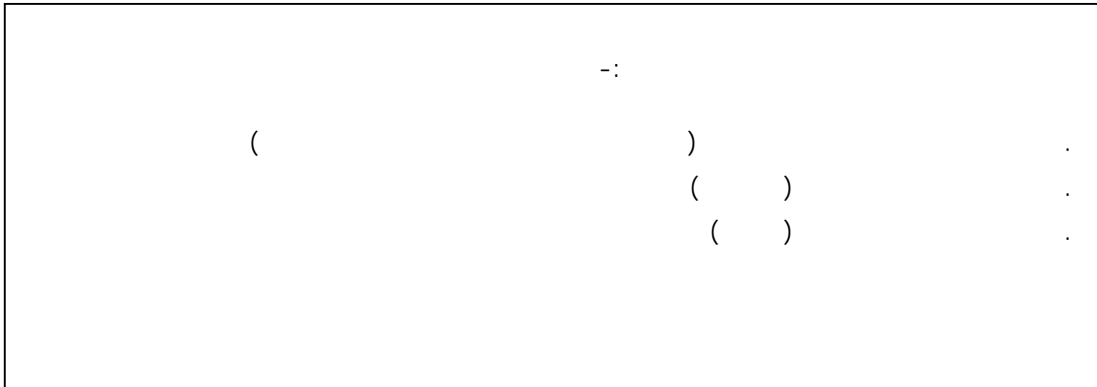


Fig.(4): domain experts screen

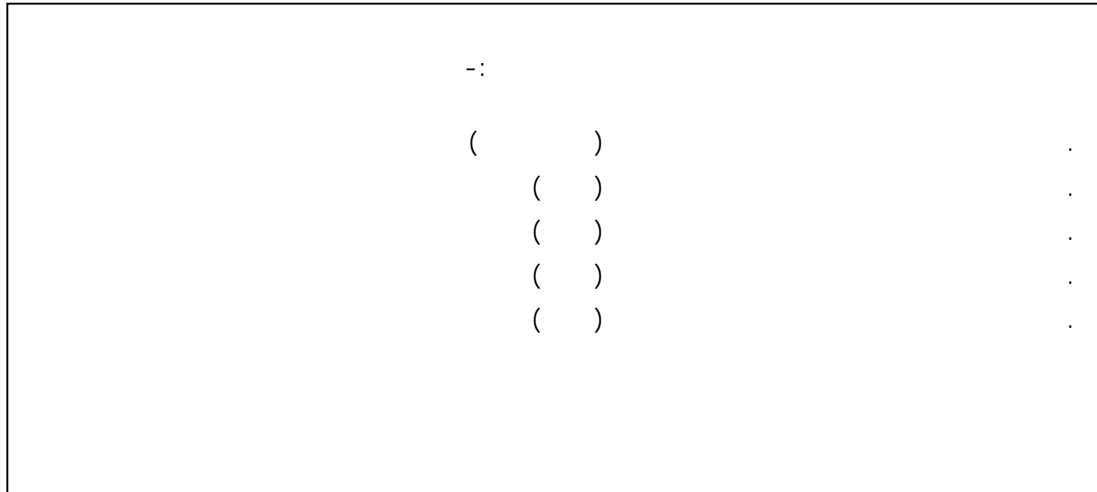


Fig.(5): CLAES staff screen

**6. Overall Control**

The control function consists of two layers. The first layer governs the execution of the whole system. The second layer is included within each subsystem, which is controlled by its own control program (task layer).

## 6.1. Control Structure

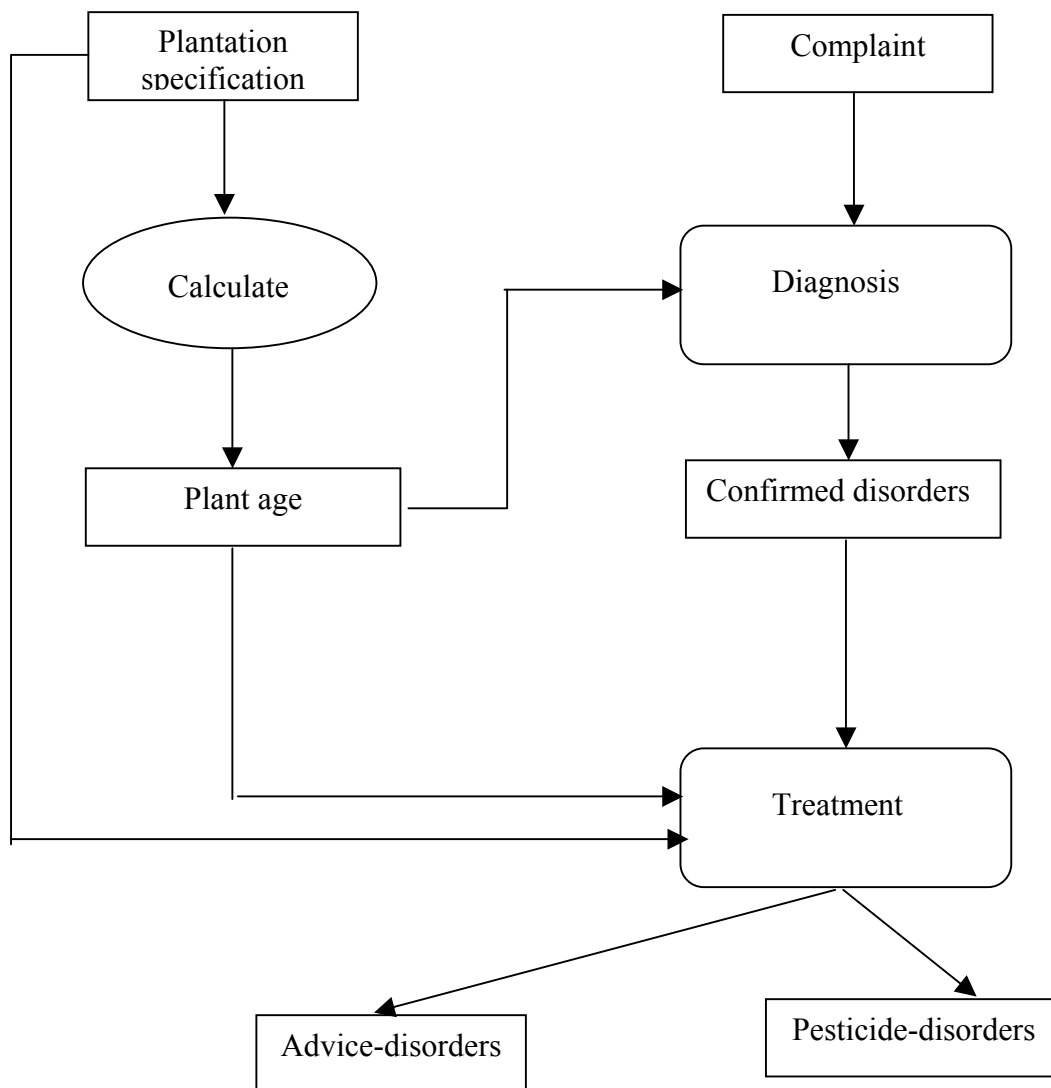


Fig. (6): Control structure of the integrated system

**inference:** calculate

**operation-type:** calculate the plant age.

**input-roles:** plantation specification; (plantation: date, plantation:type)

**output-roles:** plant age (plant: age);

**static-roles:** there is no static roles

**spec:** plant age is calculated in days by subtracting the plantation date from the session date.

## 6.2 Task

**task:** melon disorders diagnosis and treatment;

**task-definition:**

**goal:** 1- get the causes of the abnormal observations;  
2- give the disorder treatment

**input:** complaints: (plant: complaint);  
plantation specification: (plantation: date, plantation:type)

**output:** confirmed-disorders: (disorder: confirmed)  
pesticide-disorders: (pesticide: treat\_disorder)  
advice-disorders: (advice & the related disorders).

**task-body:**

**type:** composite  
**subtasks:** diagnosis, treatment, calculate  
**additional-roles** plant age: (plant: age)

**control-structure:**

```
PRESENT(welcome screen)
IF user selection is about system
THEN PRESENT(about system screen)
IF user selection is domain expert
THEN PRESENT(domain experts screen)
IF user selection is claes staff
THEN PRESENT(claes staff screen)
IF user selection is diagnosis
THEN
IF plantation: date is UNKNOWN THEN Begin OBTAIN (plantation: date) End
calculate (P: plantation specification → P: plant age),
OBTAIN (plant: complaint) % from the user
Diagnosis (complaint, P → confirmed disorders)
IF user selection is treatment
IF plantation: date is UNKNOWN
THEN OBTAIN (plantation: date)
calculate (P: plantation specification → P: plant age),
IF disorders: confirmed = ∅
THEN OBTAIN (assumption), % from the user
assert(assumption, disorders: confirmed)
Treatment (P, confirmed disorders → Pesticide-disorders, advice-disorders)
```

## 7. Test Cases

### Test No: 1

User request: diagnosis

Plantation date: 15/2/2001

session date: 31/3/2001

Leaves color:

leaves: color position =

leaves: color direction =

Root appearance:

#### Diagnosis:

User request: treatment

=  
= 0.2 %  
=  
=

### Test No: 2

plantation: type =

User request: diagnosis

Plantation date: 15/2/2001

session date: 31/3/2001

Leaves color:

leaves: appearance status =

leaves: appearance =

root: appearance =

#### Diagnosis:

User request: treatment

=  
21

### Test No: 3

User request: diagnosis  
Plantation date: 15/2/2001      session date: 31/3/2001

Leaves color:  
leaves: spots color =  
leaves: spots position =  
leaves: spots appearance =

**Diagnosis:**

User request: treatment  
10 = (user)  
=  
**EC** %50      **EC** %50      ,wp % 25      ,**WP** %50  
The user chose **EC** %50  
= %50  
= 0.387 %  
=

**Test No: 4**

User request: diagnosis  
Plantation date: 15/2/2001      session date: 31/3/2001

Leaves color:  
leaves: spots color =  
leaves: spots position =  
leaves: spots appearance =  
leaves: appearance status =  
leaves: appearance =

**Diagnosis:**

User request: treatment  
= (+ - )  
The user chose -  
= -  
= 0.25 %  
=

**Test No: 5**

User request: diagnosis  
Plantation date: 15/2/2001      session date: 31/3/2001  
stem: color-status =



stem: color =

root: color =

**Diagnosis:**

User request: treatment

= ( , , + , )

The user chose , +

= +

= 100/ 150 + 100

=

**Test No: 6**

User request: diagnosis

Plantation date: 15/2/2001 session date: 31/3/2001

leaves: appearance status =

leaves: appearance =

leaves: tunnel color =

**Diagnosis:**

User request: treatment

5 =

=

**Test No: 7**

User request: diagnosis

Plantation date: 15/2/2001 session date: 31/3/2001

leaves: appearance status =

leaves: color =

leaves: spots color =

leaves: spots appearance =

leaves: appearance =

**Diagnosis:**

User request: treatment

: = (user)

= - % 93 % 95 % 95 )

( %49

The user chose % 95

= % 95  
= 1 %  
=  
=

### Test No: 8

User request: diagnosis  
Plantation date: 15/2/2001      session date: 30/4/2001  
fruits: appearance status =  
fruits: appearance =  
fruits: appearance =

#### Diagnosis:

User request: treatment  
: = (user)  
= (wp %50      wp % 50      )

The user chose      wp % 50

= wp% 50  
= 0.075 %  
=

### Test No: 9

Plantation date: 15/2/2001      session date: 30/4/2001  
User request: treatment  
Disorder name: (      &      )

=  
= 0.25 %  
=

### Test No: 10

Plantation date: 15/2/2001      session date: 30/4/2001  
User request: treatment  
Disorder name: (      )

=  
= 0.2 %

=

**Test No: 11**

Plantation date: 15/2/2001

session date: 30/3/2001

User request: treatment

Disorder name: (            )

irrigation: type =

=

=     /            400

=

**Test No: 12**

Plantation date: 15/2/2001

session date: 30/3/2001

User request: treatment

Disorder name: (            &            &            )

= (            ,            ,            +            ,            )

The user chose

=

=     0.25   %

=

**Test No: 13**

Plantation date: 15/2/2001

session date: 30/3/2001

User request: treatment

Disorder name: (     &            &            )

= 10

4 =

8 =

= EC   %50

=     0.125   %

=

**Test No: 14**

Plantation date: 15/2/2001

session date: 30/3/2001

User request: treatment

Disorder name: (     &            &            )

```

                = 3
            4 =
                3 =
        = (      % 95      % 95
%94      ,      % 93      %49 -      )

```

The user chose

```

=
= 0.1 %
=

```

## References

**[Wielinga, 1994]** Bob J. Wielinga, Expertise Model Definition Document, ESPRIT Project P5248 KADS-II, Document Id. : KADS-II/M2/UvA/026/5.0, University of Amsterdam, 1994.