Wish-4

A specification for a CASE-tool to wish for.

By Lars Olenfeldt, Nytänkarna AB

P.O Box 348, S-43124 Mölndal, Sweden

A shortened version of a paper from
the NordDATA 90 conference in Göteborg, Sweden.

If a fairy promised you a CASE-tool, what would you wish for?

Suppose that you'll find a CASE-tool that automatizes everything that can be automatized, will there remain anything to do for the systems engineer (SE)? If so, what will a CASE-tool look like that helps in this work? Perhaps something like Wish-4?

Why do we have two eyes pointing in the same direction?

Why do we not have our eyes on the sides of our heads, like birds have? (That would be fine when crossing the street). The answer is: by having two different pictures of the same thing, we experience depth. By having two (or more) "windows" we can understand things "deeper". WISH-4 has many windows.

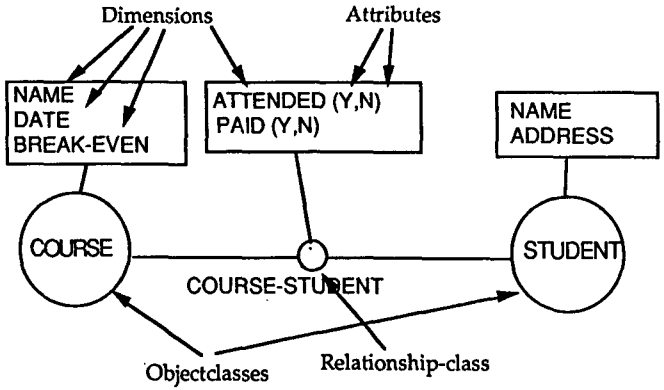
A CASE-tool should allow you to "see with your own eyes". Not just "think with your own brain".

Model

A datasystem is a model of something else. The datasystem shall reflect objects, relationships and attributes of these plus events that affect the objects and relationships.

Static model

Here we use the following notation:



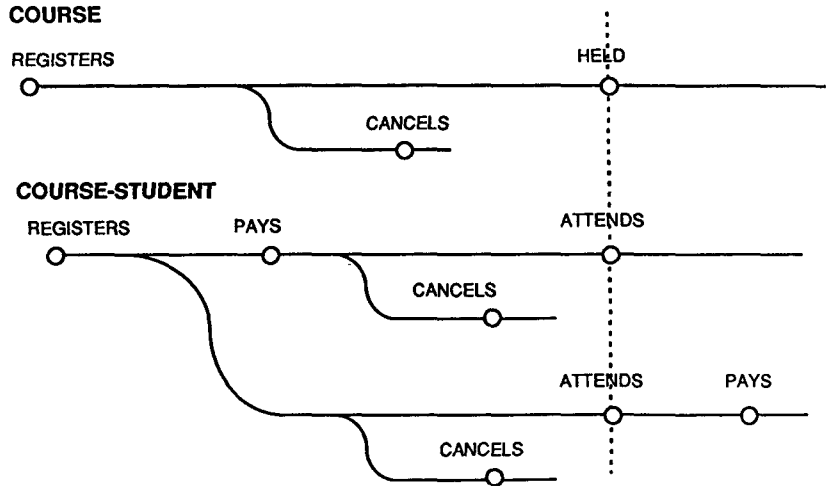
The same information can be represented as text in another window. The tool should allow me to enter text in one window and should automatically produce graphics in another. (Or vice versa).

```
COURSE                              COURSE-STUDENT
    (NAME          CHAR(20),            (COURSE      CHAR(20),
    DATE           DATE,                STUDENT      CHAR(20),
    BREAK-EVEN     INTEGER);            PAID         (Y,N),
                                        ATTENDED     (Y,N))
```

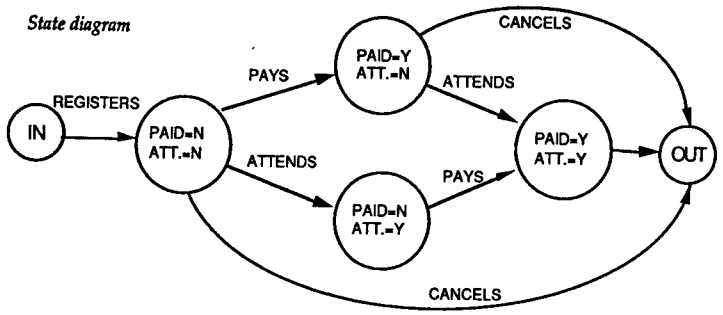
Dynamic model

For each object- and relationship-class we describe it's trajectory. In this example I will just show COURSE and COURSE-STUDENT. (The same principle holds for STUDENT).

We draw the trajectories. They are synchronized in the event COURSE-HELD. The graph below indicates among other things: A course cannot be cancelled after it has been held. A student can pay before or after he/she has attended.



The events are "pearls" on strings. Each piece of string between pearls represent a state of the object. This is an inverted state-diagram (where the states are the pearls). The state-diagram could also be automatically generated in another window.



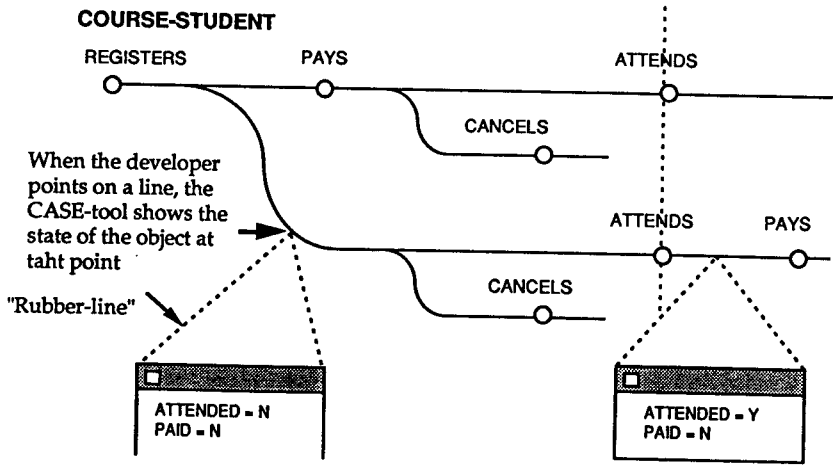
Now we can see that the attributes COURSE-STUDENT--ATTENDED (Y,N) iare affected by the event COURSE-STUDENT--ATTENDS. I many cases you will find events and states "hand-in-hand" together. You could not hope for finding all the (static) attributes first and all the (dynamic) events later.

An intelligent CASE-tool should be able to automatically find the name of a state from the name of the event. (If it could flex verbs, it should find "PAID" from "PAYS").

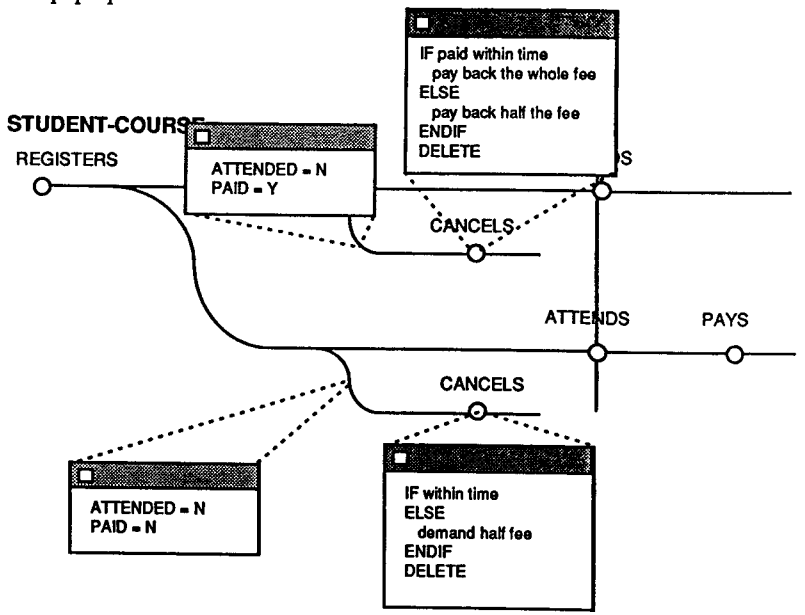
"Life-lines"

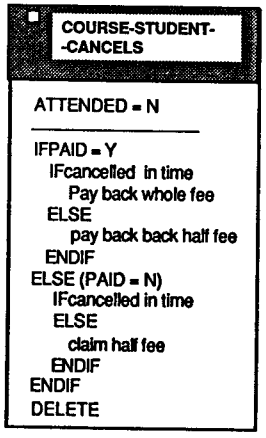
The lines in the diagram are live.

By the SE's clicking on a line (with a mouse) a window will open up showing the state of the object at that point.



When you point at an event, a window showing the processing of the event will pop up.





The graph above shows that the event (signal) COURSE-STUDENT--CANCELS can hit COURSE-STUDENT in two different states.

Wish-4 will compile information on all CANCELS into one list.

To the left you can see the window for COURSE-STUDENT--CANCELS. It shows that CANCELS is defined only when ATTENDED = N (The CASE-tool has deduced that from other windows

Events

Events in the object world.

The objects themselves generate and signal events, e.g. a student registers for a course.

Signals from objects that affect relationships

Suppose that a whole course is cancelled for some reason. The signal *COURSE--CANCELLED* can *"hit" the relationship COURSE-STUDENT* anywhere in the trajectory of the latter. Like a bolt from a clear sky. Then we want Wish-4 to display all possible states for COURSE-STUDENT so that the SE can specify the appropriate actions for each state.

```
WHEN COURSE--CANCELLED
   DELETE COURSE
   FOR ALL COURSE-STUDENTs          plural -s added by WISH-4
    {ATTENDED =N}                   comment by WISH-4
     send message "Course cancelled"   < specified by developer
    IF PAID = N
     do nothing                     < specified by developer
    ELSE {PAID =Y}
     pay money back (always whole sum)   < specified by developer
    ENDIF
    DELETE COURSE-STUDENT
   ENDFOR                           (the rest is specified by WISH-4)
ENDWHEN
```

Events not signalled from objects

The developer decides a) when to enter a plan, b) when (and how) to request output, and c) perform checks and supervision.

The developer specifies that these (non-object-signalled) events start on specific points of time or intervals or could be requested by the end user.

For example, you want to check if a sufficient number of students have registered for a course. If that is not the case you may want to cancel the course. The check is to be performed 14 days before the course is to start.

```
WHEN TODAY = COURSE--DATE - 14
    IF NUMBER(COURSE-STUDENT) < COURSE--BREAK-EVEN
        TRIG COURSE--CANCEL
    ENDIF
ENDWHEN
```


The Object List

The Object List is a useful window

OBJECT	DIMENSION or CREATE/DELETE	VALUE	EVENT condition-A condition-B etc
COURSE-STUDENT			
	CREATE		REGISTER
	DELETE		ATTENDS IF PAID = Y
			PAYS IF ATTENDED = Y
			CANCEL
	ATTENDED (N,Y)		
		→ N	REGISTER
		→ Y	ATTENDS
	PAID (N,Y)		
		→ N	REGISTER
		→ Y	PAYS

Here you can check "Is there another instance where COURSE-STUDENT--ATTENDED is set to N?" Or: "Do we have to handle the case where a student pays part of the fee as a registration fee?"

If an object in one dimension has the values, say, Y and N, then Wish-4 will force you to specify at least one event that sets the value to Y, and at least one event that sets it to N.

Wish-4 will detect that there are dimensions where the attribute never change in the trajectory. Then you'll have to ask yourself "Have I forgotten something? Or is this deadwood that can be removed?"

Standard CASE facilities

Of course Wish-4 automatically generates database descriptions, screen layouts and suggests layouts for output listings etc. These things are handled by today's CASE tools.

This is the CASE tool I wish for.

See also:

Olenfeldt: Object/Event Analysis. (SEN Jan 1985).

End 93.01.13