The AMODEUS Project
ESPRIT Basic ResearchAction 7040

MATIS:
A Multimodal Airline Travel Information System

Nigay L., Coutaz J. & Salber D.
Laboratoire de Genie Informatique, University of Grenoble

23rd February 1993

Amodeus Project Document: System Modelling/WP10

AMODEUS Partners:
MRC Applied Psychology Unit, Cambridge, UK (APU)
Depts of Computer Science & Psychology, University of York, UK. (YORK)
Laboratoire de Genie Informatique, University of Grenoble, France.(LGI)
Department of Psychology, University of Copenhagen, Denmark. (CUP)
Dept. of Computer & Information ScienceLinköping University, S. (IDA)
Dept. of Mathematics, University of the Aegean Greece (UoA)
Centre for Cognitive Informatics, Roskilde, Denmark(CCI)
Rank Xerox EuroPARC, Cambridge, UK.(RXEP)
CNR CNUCE, Pisa Italy (CNR,CNUCE)

MATIS :
A Multimodal Airline Travel Information System

Laurence Nigay
Joëlle Coutaz
Daniel Salber

February, 1993

Abstract

This document presents MATIS along the lines defined for the description of common exemplars: overview, reference material available, hardware and software platforms, usage, and future plans. MATIS allows an end-user to obtain information about flight schedules using speech, mouse, keyboard, or a synergistic combination of these techniques. At any time, the user can switch freely between these techniques: there is no prevailing modality. In addition, MATIS supports interleaving at the command level. This system runs under NeXTStep on a NeXT machine. Speech input is processed by Sphynx, a continuous, speaker independent recognition engine.

Table of contents

1. Overview

MATIS stands for Multimodal Airline Travel Information System. It allows an end-user to obtain information about flight schedules using speech, mouse, keyboard, or a combination of these techniques. User's requests are translated into the SQL formalism to access information stored in a data base. Currently, the database contains nine American cities, nine airline companies and miscellaneous information about each flight such as flight number, departure and arrival time, meals and prices. Speech input is processed by Sphynx, a continuous, speaker independent recognition engine developed at Carnegie Mellon University [Lunati 91]. MATIS runs under NeXTStep on a NeXT machine.

According to the framework described in [Coutaz 93], MATIS is a multimodal interactive system for input only. At the command level, it supports sequential as well as concurrent fusion of input data issued through two distinct communication channels (microphone on the one hand, mouse and keyboard on the other hand). It also performs fusion of data produced through a single channel such as speech or mouse only.

In other words, MATIS supports the synergistic use of two input modalities (mixing speech and mousing as in "show me USair flights from Pittsburgh to *this* city") as well as the exclusive use of these modalities using speech only (as in "show me USair flights from Pittsburgh to Boston") or filling a form using mouse and keyboard only. At any time, the user can switch freely between these techniques: there is no prevailing modality. Each modality offers the same power of expressiveness.

In addition, MATIS supports multithreading. It allows the user to work on multiple requests in an interleaved way: although a request may be partially formulated, the user may start a new one then come back to the previous one. To our knowledge, multimodality combined to multithreading is a unique feature.

The next paragraphs present the user interface of MATIS. Its software architecture is then sketched in terms of the PAC-Amodeus architectural model.

1.1. The Output Interface

As shown in figure 1, the graphical appearance of the output interface adheres to the NeXTStep look and feel. In the following, features inspired from NeXTStep are denoted by an asterix (*). Others ar inherited from the Sphynx user interface. These are denoted by (@).

The output interface is composed of a set of windows and panels:

- The left side of the screen is dedicated to the NeXTStep "application dock" (*). It is not part of MATIS per se.

- At the top left corner, the MATIS main menu offers five options (*)[1]. In particular, the option "Tools" provides access to four complementary panels: the Request Tools, the requests History, the Notepad, and the Speech panel.

[1] In NexTStep, every application has a main menu. This menu is equivalent to the menubar for the Macintosh.

• The Requests Tools panel (top left center of Figure 1) offers a set of tools to specify request slots using the mouse. Each tool is represented as an icon. When selecting such an icon, a form pops up whose content depends on the tool. For example, when selecting the City icon, a scrollable list of the available cities pops up. The user may then choose one item of the list to specify a departure or an arrival location.

Seven distinct forms are accessible through the icon tools: The City form, the Time form (departure/arrival time), the Flight Number form, the Miscellaneous form (number of stops required), the Airline Company form (list of airlines), the Fare form (ticket class), and the Meal form (dinner, breakfast, etc.).

• The Requests History (bottom left of Figure 1) displays all of the requests that have been submitted so far to the system. Through copy and paste operations, it is possible to reuse part or all of the content of previous requests as parameters for new requests. In particular, this facility provides a way to re-issue previous requests.

• The Notepad panel (bottom right of Figure 1) allows the user to gather data about interesting flights and type-in additional annotations.

• The Speech panel (not shown in Figure 1) presents the results of the syntactic analysis of each recognized utterance (@). This information is useful to check the parser and mapper.

• The Office Manager window (top center right of Figure 1) highlights the icon of the application currently connected to the speech recognition system (e.g., MATIS) (@). It also shows the last sentence recognized (currently, "USAIR FLIGHT FROM PITTSBURGH TO BOSTON SERVING A MEAL") and it pops up when a failure occurs (e.g., unknown words). These informations are very useful to check the robustness of the system.

• At the top right corner of the screen, an icon expresses the current state of the speech recognition engine: Ready, Listening or Searching. As shown in Figure 1, the system is currently ready (@).

• Request windows are used to specify requests to the system. There is one such window per on going formulation. A window request is composed of two parts: the button icons at the top (*), and the parameter slots.

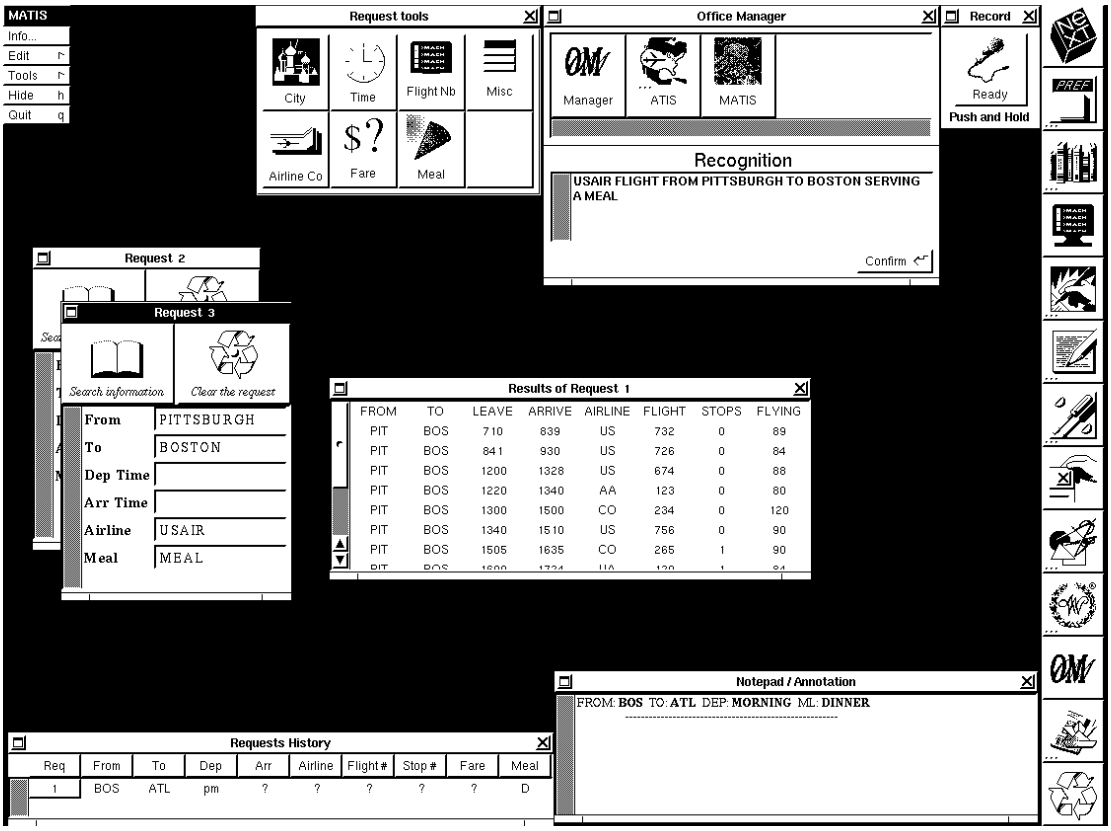
The icons are inspired from the NeXTStep environment. When the request is empty, the book is closed and the book button is inactive (i.e., greyed out). When enough slots have been specified by the user (i.e., the departure or the arrival city), the book opens automaticaly and becomes selectable. When pressed, the user's request is sent to the data base. The recycler button is always available. When selected, it clears up the request form.

Because slots are too numerous to be listed in every request, they have been divided into two categories based on their frequency of use. Those that are frequent or mandatory (e.g., From, To, Dep. Time, Arr. Time) are automatically listed in the form when a new empty form is proposed to the user. The user can then add extra slots in the form using the Request Tools or using speech.

An empty request form pops up at system start up as well as when an answer is provided to a previous request.

MATIS supports multithreading: it is not mandatory to finish a request to start another one. Figure 1 illustrates this facility: two requests (middle left of the screen) are currently being formulated. To make an old request active, the user has to select the corresponding window. The request will come to the front and will constitute the new current context of interaction.

- The Results window displays the answer received from the database. There is one such window per request submitted to the system. Any data from any results window can be selected to be copied elsewhere, for example, into the Notepad.



Recognition: USAIR FLIGHT FROM PITTSBURGH TO BOSTON SERVING A MEAL

Request 3:

From	PITTSBURGH
To	BOSTON
Dep Time	
Arr Time	
Airline	USAIR
Meal	MEAL

Results of Request 1

FROM	TO	LEAVE	ARRIVE	AIRLINE	FLIGHT	STOPS	FLYING
PIT	BOS	710	839	US	732	0	89
PIT	BOS	841	930	US	726	0	84
PIT	BOS	1200	1328	US	674	0	88
PIT	BOS	1220	1340	AA	123	0	80
PIT	BOS	1300	1500	CO	234	0	120
PIT	BOS	1340	1510	US	756	0	90
PIT	BOS	1505	1635	CO	265	1	90
PIT	BOS	1600	1724	UA	120	1	84

Notepad / Annotation: FROM: **BOS** TO: **ATL** DEP: **MORNING** ML: **DINNER**

Requests History

Req	From	To	Dep	Arr	Airline	Flight#	Stop#	Fare	Meal
1	BOS	ATL	pm	?	?	?	?	?	D

Figure 1. A snapshot from the MATIS application.

1.2. The Input Interface

MATIS supports multiple input modalities. The user can use each of these modalities independently or combine them in a synergistic way. To clarify the discussion, we define a modality as the couple <d, f> where d denotes the physical input device used to specify an expression to the system, and f defines the formalism (i.e., the interaction language) interpreted (i.e., understandable) by the system.

MATIS supports the following modalities:

- <keyboard, English natural language>: the user can type-in English pseudo-natural language in the Recognition slot of the Office Manager Window. By pseudo-natural language we mean “natural language constrained by a grammar and a vocabulary”.

- <microphone, English natural language>: the user can voice the request in an English pseudo-natural language.

- <mouse, graphical formalism>: the user can fill-in request forms by using the Request Tools with the mouse. For example, selecting the Meal button in the Request Tools panel results in popping up a form that contains a list of possible meals (i.e., breakfast, lunch, and dinner) as well as the usual Cancel and OK buttons. Selecting the breakfast item and the OK button, automatically fills in the Meal slot in the current Request window.

- <keyboard, graphical formalism>: the user can fill in the request forms by typing directly in the request slots.

At any time, the user can switch freely between these modalities. In MATIS, all of the modalities have the same power of expressiveness. Thus, any request can be specified using either:

- one modality only: for example, a complete request may be formulated using <keyboard, English natural language> only, <microphone, English natural language> only, <keyboard, graphical formalism>, or <mouse, graphical formalism>. In particular, to submit a request to the system, the user can either say "show me" or click the Book button in the current request window.

- multiple modalities combined in a sequential way: the user may specify a couple of slots using the <microphone, English natural language>, then complete the form using the <mouse, graphical formalism>. For example, it is possible to say "flights from Pittsburgh to Boston". As a feedback, the system shows what has been understood in the Recognition window, and fills in the From and To slots of the current request window). Then the user can complete the form by typing in "morning" in the Dep. Time slot. As another example, the user can say "flights from", then selects Pittsburgh in any of the Results, History or NotePad windows, then say "to", then selects Boston in any of the Results, History or NotePad windows.

- multiple modalities combined concurrently. This facility supports deictic expressions mixing speech and gesture. For example, it is possible to say "flights from this city to this city arriving in the afternoon" while performing mouse clicks to denote the two cities referred to in the utterance.

In summary, according to the classification of multimodal interactive systems presented in [Nigay 93], MATIS supports the exclusive, alternate, concurrent and synergistic use of input modalities. Switching between these possibilities is at the user's will.

1.3. The software architecture

The software architecture of MATIS relies on the PAC-Amodeus model. One of our motivations for developing MATIS was to check the capacity of PAC-Amodeus to support the software design of multimodal systems. PAC-Amodeus is fully described in [Nigay 93] and [Nigay 91]. Figure 2 presents the implementation components of MATIS.

Behind the scene, (i.e., as a third dimension to the description presented in Figure 2), we have implemented a general mechanism for performing the fusion of incoming data. These data are provided by the Presentation or the Interaction components. Our fusion engine comes into play within the Dialogue Controller component.

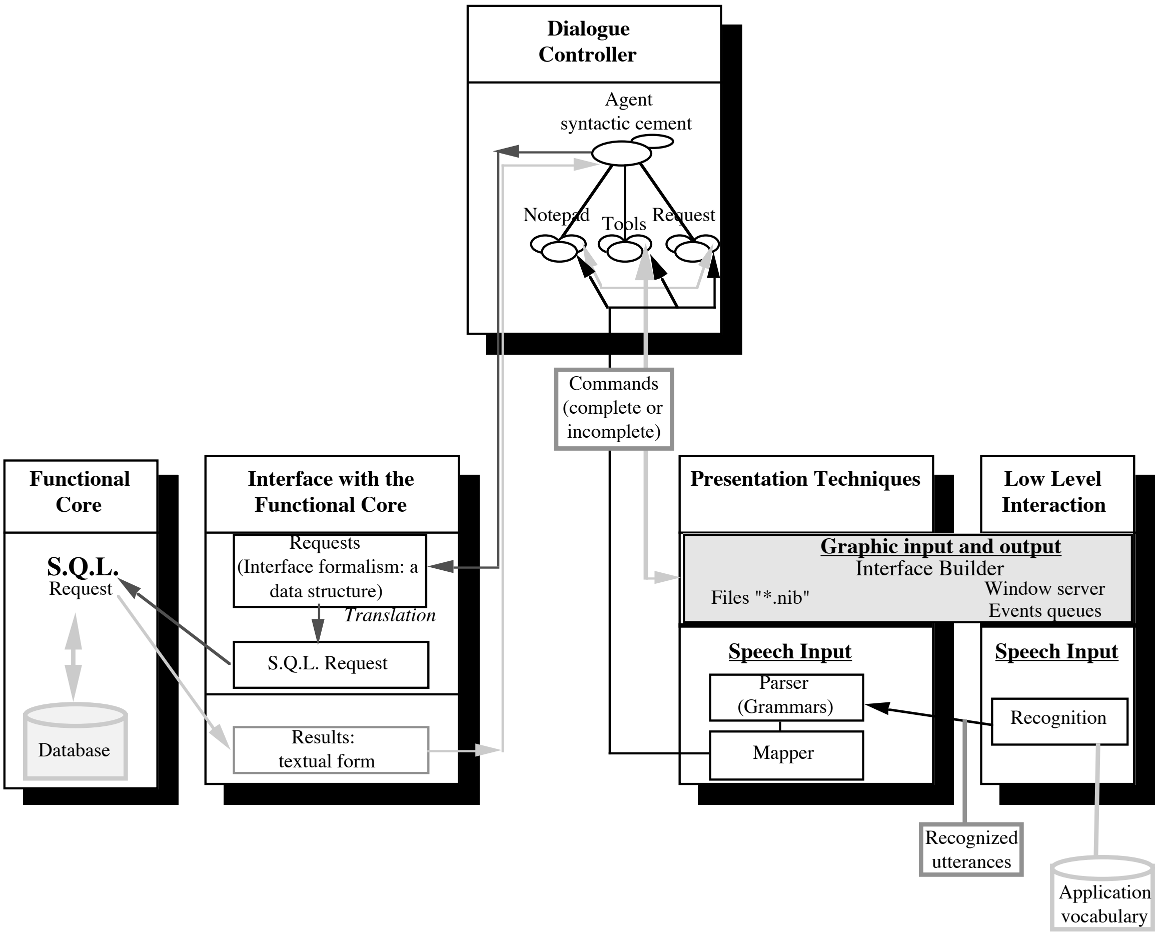


Figure 2. The software components of the MATIS application.

The fusion mechanism is domain independent and adopts the "eager" strategy: it never stops waiting for further information and always makes attempts to combine input data. It is then possible that incorrect fusions are performed and must be undone. On the other hand, it has the advantage of performing immediate feedback. Fusion is performed within melting-pot objects, that is, a common representation used at different levels of abstraction [Nigay 93]. The criteria for triggering fusion are the logical structure of commands, time and context. The fusion engine implements three types of fusion: the macro, the micro and the contextual fusions.

- **Micro fusion** is based on the structure of the data to be combined and time. Micro fusion combines data if they have been produced in parallel or in pseudo-parallelism. (Intersection of time intervals.)

- **Macro fusion** is based on the same criteria as the Micro Fusion but combines data belonging to a given temporal window. (Temporal proximity.)

- **Contextual fusion** is based upon the structure of the data to be combined and the current context. The context here is reduced to the current request. Contextual fusion combines a new input data with the current request if their respective structures are compatible.

2. Reference material available

MATIS is described in a short report written in French in October 1992.

A twelve minute NTSC video has been done at CMU. This video is divided into three parts. The first part presents how to use MATIS using keyboard and mouse only. The second part shows how to use MATIS with speech only. And the third part presents the various way of combining the input modalities to specify a request.

Each part is summarized by a list of the key points we wanted to make.

3. MATIS and the criteria for exemplar selection

1) Real world validity

Although MATIS has been used so far by members of the speech recognition team at CMU and by our team in Grenoble, we plan to experiment MATIS in real conditions, asking volunteers to realize a scenario. The scenario is provived in the Annex.

The voice interface of MATIS results from several years of study performed by the CMU speech group. Within CMU, the project is called ATIS. For two years, experiments with representative volunteers have been conducted to elaborate a realistic vocabulary. These experiments have used the Wizard of Oz technique to record behavioral data.

2) Design process as well as product

About the software design of MATIS, we followed the "V" software engineering life cycle: In particular we designed the external specification as a prototype using Interface Builder [Webster 89] and we then designed the software architecture applying the PAC-Amodeus model before the coding phase.

3) Influence over design

Does not apply so far.

4) Available design knowledge

MATIS has been developed by the Grenoble team only.

5) Domains of interest

MATIS provides two interesting features:
- multithreading at the request level,
- multiple ways of using modalities (alternate, concurrent, exclusive, synergistic).

Thus MATIS can be used to observe how this type of flexibility is good to the user?

6) Support development of integration

One of our main goal for developing MATIS is the study of the fusion mechanism of input data coming from different modalities. In particular, this study gave rise to two problems: the common data representation and the criteria for triggering fusion.

MATIS supports :

- multiple resource types,
- multiple channels,
- multiple interaction styles,
- multiple tasks (assuming that one task is one request for example).

7) Familiarity
The domain of MATIS is airline travel information. It is a familiar domain.

8) Communicable
The reference material available is described in Section 2 of this document.

A running version of MATIS is available using the NeXTStep operating system..

9) Coverage of models
To be discussed. From the system point of view, MATIS illustrates a number of interesting phenomena: parallelism at the physical level, fusion at various levels of abstraction, interleaving at the command level, support for deictic expressions.

10) Commonality of analysis
Does not know. To be discussed.

Figure 3 summarizes MATIS and the criteria for exemplars selection.

1. Real-World validity	YES
2. Design process as well as product	YES
3. Influence over design	?
4. Available design knowledge	YES
5. Domains of interest	YES
6. Support development of integration	YES
7. Familiarity	YES
8. Communicable	YES
9. Coverage of models	SYSTEM+?
10. Commonality of analysis	?

Figure 3. MATIS and the criteria for exemplars selection.

4. Hardware and Software Platforms

MATIS is been developed on the NeXT workstation using the interface generator, Interface Builder [Webster 89]. Interface Builder is a tool that allows the developer to define the graphical behaviour by direct manipulation.

The voice interface is the Carnegie Mellon Spoken Language Shell (CM-SLS) [Lunati 91]. It provides continuous, speaker independent voice recognition. Speech input is constrained by a grammar and a vocabulary. This grammar is application-dependent, and is used to parse recognized utterances.

The current version of MATIS that runs in Grenoble has a very limited “faked” data base. The database is simulated as one file that contains descriptions for flights from Pittsburgh to Boston only. It is however easy (but boring) to extend it.

5. What we have done with MATIS and future plan

The orginality of MATIS is the multi-threading. MATIS allows the end-user to start a request, switch to an another request and come back to the previous one. Multi-threading makes the fusion mechanism more complex, allowing multiple contexts where the fusion can be performed.

MATIS has been used and evaluated by the members of our team only. MATIS is a platform to study:

- the software architecture of multimodal systems and in particular the correctness of PAC-Amodeus,
- the strategy and criteria for the fusion mechanism,
- the common representation formalism of data to be combined.

Although technically MATIS is a challenging case study, we do not know whether multithreading and multimodality are useful concepts with regard to the user's task. In particular we would like to study the synergistic use of modalities: is it used? is it usable? etc. To do this study, we plan a set of experiments asking volunteers to realize a scenario. This scenario is presented in the Annex.

Behavioral data recorded during these experiments will denote usability and learnability problems. As software designers, we will be able to check how the PAC-Amodeus model is able to support modifications within the user interface.

6. References

[Coutaz 93]
J. Coutaz, L. Nigay, D. Salber: The MSM framework, A Software Design Space for Multi-Sensori-Motor Systems, Amodeus 7040 Working paper, SystemModelling/WP4, Jan. 1993.

[Lunati 91]
J-M Lunatti, A. I. Rudnicky, Spoken Language interfaces: The OM system. In proceedings of CHI'91 Conference, New orleans, April 27-May 2, 1991, pp. 453,454.

[Nigay 91]
L. Nigay and J. Coutaz, Building User Interfaces: Organizing Software Agents, Conference ESPRIT'91, Bruxelles, November 1991.

[Nigay 93]
L. Nigay and J. Coutaz, A Design Space for Multimodal interfaces: concurrent processing and data fusion, InterCHI'93, May, 1993.

[Webster 89]
B. F. Webster, The NeXT Book, Addison Wesley, New York, 1989.

Annex, usability scenario description

You are a famous author traveling to different cities to promote your newest book.

a) First, you will travel from Pittsburgh to Boston for a writers convention where you will give the keynote address at 12 am . You would prefer not to sleep in Dallas. A long trip is waiting for you.

b) On the same day you will fly to Dallas. You are due in Dallas, for an autograph session at the major book store from 8am to 12am.

c) Then the following day you are off to San Francisco to appear for a local talk show. The show airs at 9pm. So you want to make sure you get there on time. The show will finish around 12am.

d) Then you will return home to Pittsburgh from San Francisco as soon as possible to be relax after this exhausting trip !

Since you are a member of the USAir frequent traveler program, you will prefer to travel with USAir.

Fast facts about the Data Base:

Cities:

Atlanta
Baltimore
Boston
Dallas/Fort Worth
Denver
Oakland Calif.
Philadelphia
Pittsburgh
San Francisco
Washington

Information:

Flight numbers
Airlines
Inflight Meals
Diretc/non direct flights
Depart/Arrival time