NICE project (IST-2001-35293)

Natural Interactive Communication for Edutainment

NICE Deliverable D1.2a

Analysis and representation of domain information, personality information and conversation behaviour for H.C. Andersen in the first prototype

15 October 2003

Authors

Niels Ole Bernsen, Laila Dybkjær and Mykola Kolodnytsky

NISLab, Odense, Denmark

Project ref. no.	IST-2001-35293
Project acronym	NICE
Deliverable status	Restricted
Contractual date of delivery	1 September 2003
Actual date of delivery	15 October 2003
Deliverable number	D1.2a
Deliverable title	Analysis and representation of domain information, personality information and conversation behaviour for H.C. Andersen in the first prototype
Nature	Report
Status & version	Final
Number of pages	26
WP contributing to the deliverable	WP1
WP / Task responsible	NISLab
Editor	Niels Ole Bernsen
Author(s)	Niels Ole Bernsen, Laila Dybkjær and Mykola Kolodnytsky
EC Project Officer	Mats Ljungqvist
Keywords	Life-like animated conversational agents, domain modelling, personality modelling, conversation behaviour
Abstract (for dissemination)	This report, Deliverable 1.2a from the HLT project Natural Interactive Communication for Edutainment (NICE), describes the analysis and representation of domain information, personality information and conversation behaviour for life-like animated conversational agent H. C. Andersen in the first NICE prototype.

Table of Contents

1 Introduction

This NICE report D1.2a builds on Report D1.1 “Requirements and design specification for domain information, personality information and dialogue behaviour for the first prototype”. Both reports are based on work on WP1 “Domain representation and domain-oriented conversational human-machine dialogue”, and both reports serve to document the basis for the first NICE prototype (PT1). The reported work on WP1 provides input to WP4 (animated characters) and WP5 (conversational abilities). In particular, the reader is referred to D5.1a “First prototype version of conversation management and response planning for H.C. Andersen” which builds on the present report, describes the character module components which are not described in the present report, and provides an information flow description of the HCA character module. In addition, the WP1 work draws on information from the analyses of data collected in WP2 on interaction between users and incremental versions of the first prototype. As, the WP2 data analyses will only be reported later, below we provide initial information on the HCA data collections which have been made already.

Given the differences in domain information, personality, and dialogue behaviour of Hans Christian Andersen (HCA) in his study, on the one hand, and the fairy tale characters in their fairy tale world on the other, clarity of exposition suggests to divide D1.2 into two parts, one on HCA in the present report (D1.2a) and one on the fairy tale characters (D1.2b).

It is useful by way of introduction to re-visit D1.1 in order to highlight any differences between the specification presented in D1.1 and the more specific implementation strategies presented in this report.

Compared to D1.1, HCA’s domain knowledge and interests in PT1 still cover the domains of Life, Work (including fairy tales), Physical Presence, fairy tale world Gatekeeper, the present User, and the ‘Meta’ domain which is takes care of meta-communication in the conversation. Thus, PT1 still follows the ambitious multi-domain representation strategy described in D1.1 and counter-balanced by rather limited domain representations for most of the domains covered. Within these domains, and compared to D1.1, HCA’s knowledge topics have been partly reduced and partly expanded for PT1 purposes. This is partly due to further design analysis and specification, and partly due to the fact that the entire PT1 domain knowledge specification has been exposed to approx. 30 hours of Wizard of Oz simulated English conversation with young users in the HCA Museum in Odense during the summer of 2003, yielding a virtual gold mine of data which is being analysed at the time of writing. Prior to that, in the autumn of 2002, NISLab collected 6 hours of Wizard of Oz simulated English conversation with young users at schools in Odense. This earlier data served to refine the specification prior to the Wizard of Oz simulation made this past summer.

HCA’s personality, including his emotional state space, as specified in D1.1, is embodied in the PT1 Conversation Intention Planner (CIP), the six Domain Agents (DAs), one per domain, and the Knowledge Base (KB), cf. Figure 1.1.

HCA’s dialogue behaviour as specified in D1.1 has been refined into a global conversational agenda for the PT1 Conversation Intention Planner (CIP). The CIP has a fundamental role in processing the user’s input to the HCA character module, always seeking to first handle meta-communication and then to process the input by domain, topic, and sometimes semantics, in order to suggest how to continue the conversation, taking into account as well part of the emotional impact on HCA’s emotional state of the user’s input. In addition to this “no-hardwired-dialogue-structure” approach, we have invented and developed the notion of *mini-dialogues* which will allow HCA, on occasions at which he takes particular interest in the user’s input, to carry our in-depth conversation on certain topics.

In terms of HCA character module (CM) architecture, we have only made relatively minor modifications compared to the CM architecture presented in D1.1. The revised PT1 CM architecture is shown in Figure 1.1. The main changes are that the conversation history now is a common resource to be consulted by the modules in the mind state agent (MSA), and that a separate module takes care of emotional state updating (EC). The entire NICE PT1 architecture is shown in NICE Report D3.3 and, more recently, in Report D3.7.

The entire HCA character module is implemented in C++. The HCA knowledge base is an Access database.

In the following, Section 2 describes the analysis and representation of HCA's domain information. Section 3 describes the analysis and representation of personality information for HCA. Section 4 describes the analysis, design and representation of conversation behaviour for HCA.

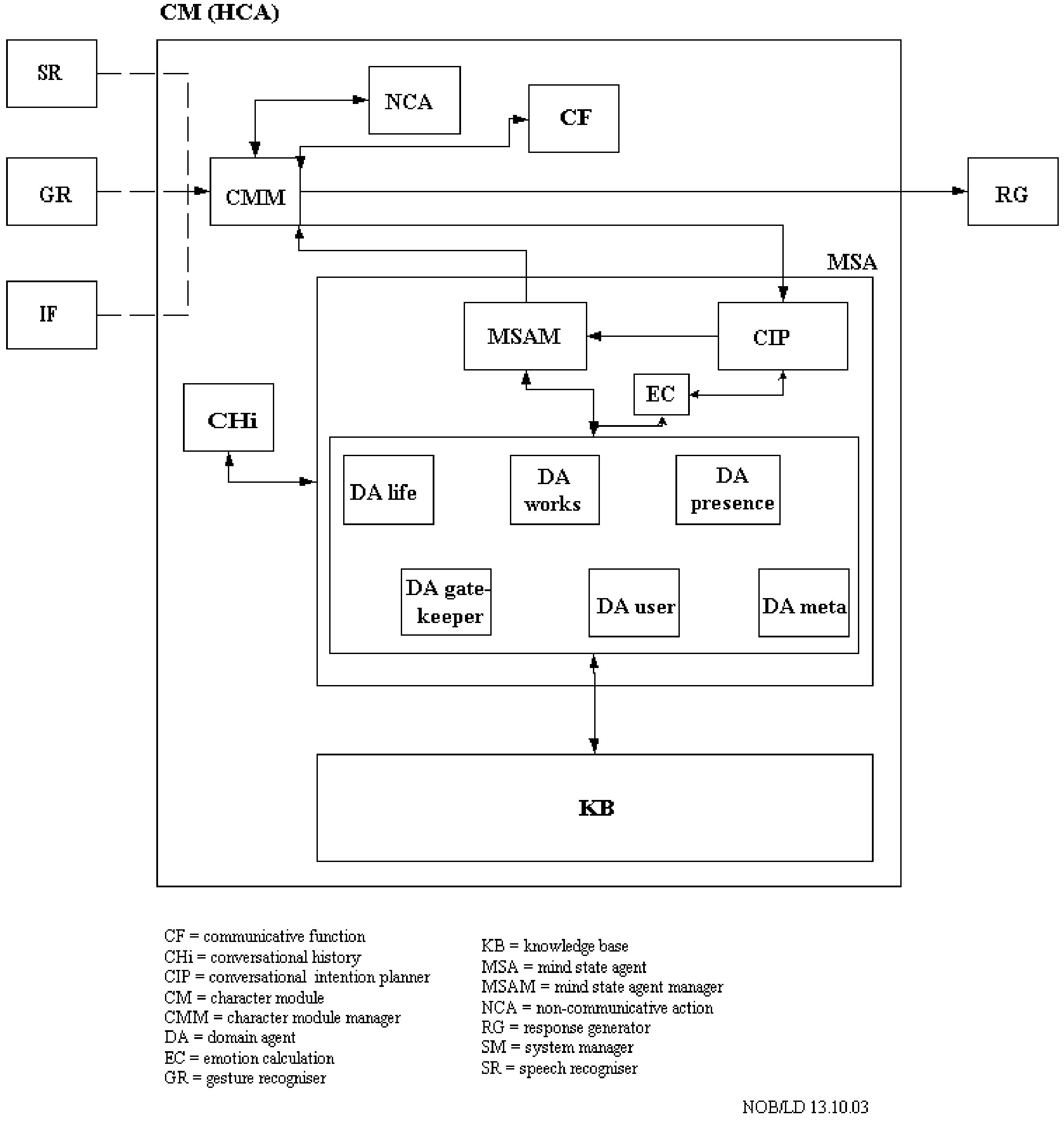

Figure 1.1. HCA character module architecture for the first NICE prototype.

2 Analysis and representation of domain information for HCA

Typically, task-oriented spoken, and possibly multimodal, dialogue systems which need more than procedurally encoded domain information, have their domain (i.e. task) information located in external information stores, such as databases, which are being queried by the system's dialogue manager. If these systems need task reasoning, this reasoning is performed in state-of-the-art modular systems by one or several task specialist modules, or task agents, see, e.g. [Charfuelàn and Bernsen 2003]. Task reasoning, in this context, means reasoning about the user's current input in order to pre-process and/or post-process the consultation of external information stores with the ultimate purpose of deciding which output to produce. Finally, most state-of-the-art task-oriented systems have a separate set of modules in the dialogue manager for producing semantic output. To ensure modularity and easy portability to new tasks, these modules are split into, on the one hand, one or more task-dependent, hard-coded dialogue structures (or models) which contain the system's semantic output per task, and a general-purpose dialogue structure processor which works on the currently relevant dialogue structure to execute procedures on the dialogue structure and to ultimately retrieve the system's response to the user's most recent input [Charfuelàn and Bernsen 2003].

By contrast with the task-oriented spoken dialogue systems described above, the NICE HCA system is a conversational, non-task-oriented system. In NICE, as described in more detail in D5.1a, this implies a number of important modifications to the familiar architecture and information flow designed to deal with task information in task-oriented systems. In particular, as the NICE character module does not make use of global hard-coded dialogue structures, one per domain, there is no longer separation between the bulk of the system's domain information and the system's store of semantic output.

The bulk of HCA's domain knowledge is located in the HCA character module's knowledge base (KB). A large fraction of this knowledge constitutes, at the same time, the CM's store of semantic output. The knowledge base is described in Section 2.1. By analogy with state of the art task-oriented dialogue managers, the character module's capabilities for reasoning about domain-specific user input are embodied in six domain agents, one per domain (cf. Section 2.2). The domain agents work in close collaboration with the knowledge base under the guidance of the conversational intention planner which is described in Section 3.1.

2.1 Domain knowledge representation

In PT1, the bulk of HCA's domain knowledge is implemented in a relational database. Based on refinements of the HCA domain analysis and domain design specification described in D1.1, HCA's hierarchically organised domain knowledge (or ontology) has been folded into the tabular organisation of the relational database representation format. The leaves of this hierarchical tree-structure are represented as a set of records in the database. These records consist of references to combined spoken and non-verbal output to be generated by the HCA response generator.

2.1.1 HCA knowledge base structure

Figure 2.1 shows a conceptual model of HCA's knowledge base drawn from the point of view of response generation.

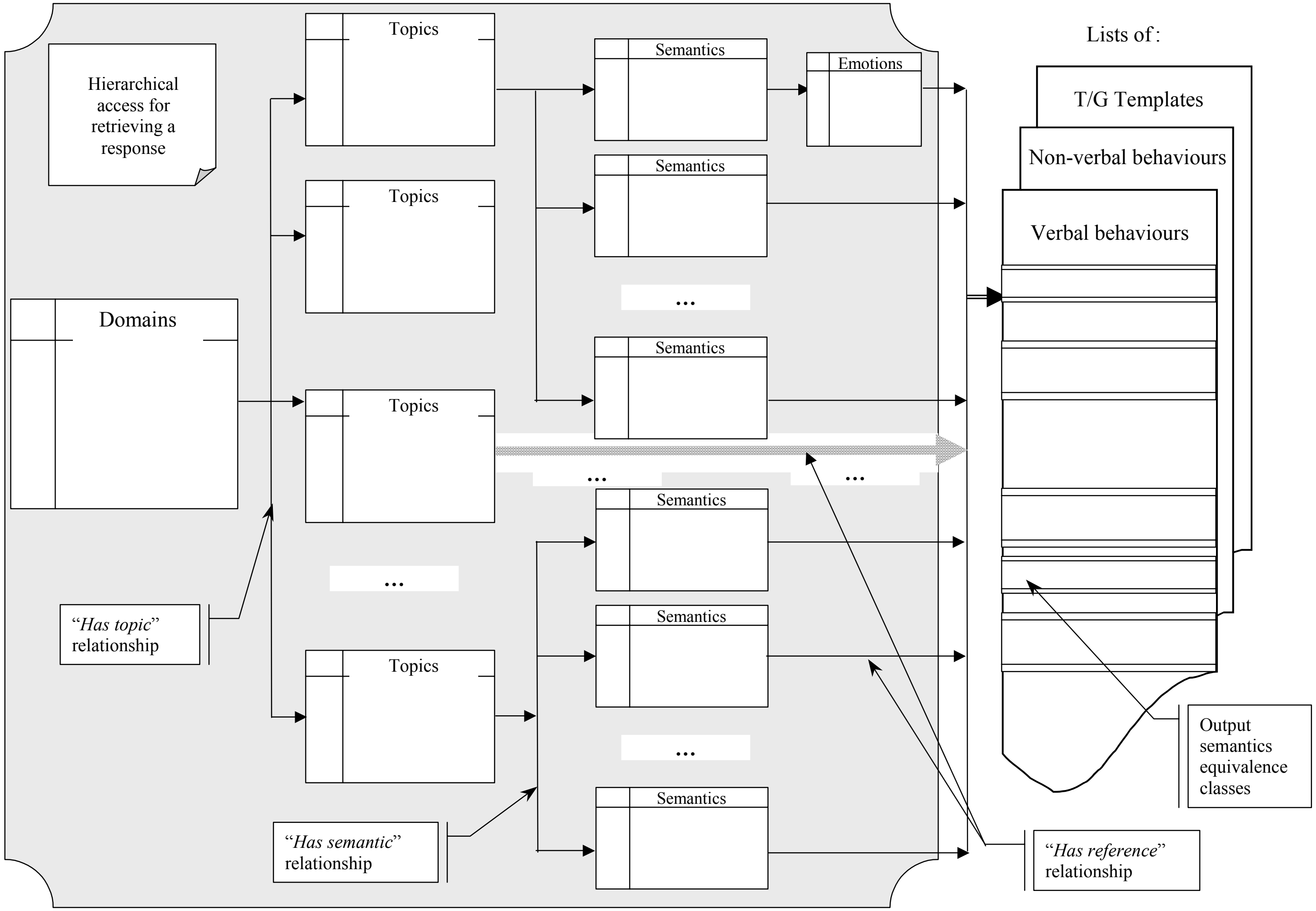

Figure 2.1. HCA knowledge base: conceptual model from the point of view of response generation.

In Figure 2.1, the character's knowledge base is shown as expanding hierarchically from left to right. The topmost level includes HCA's six knowledge domains (Table 2.2). Then follows the expansion of each domain into a set of conversational topics which, in PT1, run into about 60 topics. Each topic subsumes a set of input semantics equivalence classes. The point here is that, even if the HCA natural language understanding module is a many-to-one surface language input processor, it generates a new many-to-one multitude of semantic representations, i.e. a semantic equivalence class, relative to the records in the knowledge base. As some semantic user inputs generate emotional increments (see Section 3.2), the input query to the knowledge base which includes HCA's updated emotional state, is run through an emotions filtre which assigns the correct database record to the input query depending on HCA's current emotional state. Finally, based on the branching through the knowledge base described so far, one or more records are queried in, or retrieved from, the records lists shown at the extreme right in Figure 2.1. T/G templates are combined text-graphics output templates. Knowledge base queries may also skip the semantic contents and emotions levels as shown in the wavy line from topics to records in Figure 2.1.

The hierarchical structure represented in the knowledge base conceptual model can be viewed as an HCA ontology within which each level bears the ontological relationship to the following level indicated in Figure 2.1.

Entities in the knowledge base, i.e. domain, topic, semantics, emotion, and record, share the common structure shown in Table 2.1.

№	Property name	Description	Type (Format)
1	Id	Internal id; the database deals with it rather than a user/programmer	Internal for DB (in case of MS Access MDBS, it is so-called an auto-increment type)
2	Name	Example: domain name	String
3	ReadyPT1	Whether or not the entity is included in the 1st prototype?	Boolean
4	IsLeafNode	Does it have a child table/entity?	Boolean

Table 2.1. Knowledge base entity structure.

Table 2.2 shows the contents of the domain table at the highest level of the knowledge base hierarchy.

Id	Name
1	Life
2	Physical presence
3	Gate-keeper for the fairy tale world
4	Knowledge about the user
5	Meta
6	Works

Table 2.2. The domain table.

In addition to the entities shown in Figure 2.1, the knowledge base also includes mini-dialogue structures (see Section 2.2).

2.1.2 HCA knowledge base queries

In order to retrieve database contents, such as a particular output reference record, the database is queried by the relevant domain agent based on the user's current input, HCA's conversational agenda, HCA's emotional state, previously processed information from the conversational history, and query pre-processing reasoning done by the domain agent. Database queries include combinations of the following attribute-value structures:

- current input domain
- current input topic
- current input semantics
- current emotional state
- database record id
- mini-dialogue id

The following query types are used in PT1:

- query to retrieve an output record. An example is the search for an answer to the user's question "How many fairy tales did you write?". Another example is the search for a question or statement with a particular database id located under a particular topic. This question or statement will then be used as a continuation of the conversation, cf. Section 3.1;
- query causing database search for information from several records. An example is the look-up of an answer to the question "Did you write any scary fairy tales?". In this case, the database checks with each fairy tale represented in the database if the "scary" attribute is true or false. If true, the title of the fairy tale is retrieved and returned. Note that this functionality may be postponed to the second prototype;
- query for a mini-dialogue structure which will then be temporally stored for processing in the relevant domain agent;

The knowledge base query language is SQL.

The PT1 database records include the following data types:

- output references to spoken and/or non-verbal output behaviour;
- Boolean values used in multi-record retrieval, such as when retrieving the number of scary fairy tales in the database;
- mini-dialogue structures.

2.2 Domain reasoning

In the present context, 'domain reasoning' means reasoning about domain-specific user input, i.e. input which has already been identified by the HCA character module as pertaining to a particular domain among HCA's six knowledge domains.

Domain reasoning is being done by the six domain agents (DAs), one per domain (Figure 1.1). Basically, the six HCA DAs have the same functions, namely to pre-process database queries, query the database, and post-process database returns. The DAs are not completely identical, however. For instance, gesture input, possibly combined with spoken input, is always processed by the Physical Presence DA, except in cases of miscommunication which are all being processed by the Meta DA. Also, due to its special task of handling meta-communication, the Meta domain agent is somewhat different from the other DAs.

2.2.1 Types of domain reasoning

The following types of domain reasoning are being performed by the PT1 DAs:

- meta-communication reasoning in the Meta DA, addressing, e.g., repetitions, clarifications, input which was not understood by the pre-character module input processing modules, and insulting input;
- reasoning about gesture objects by the Physical Presence DA, making sure that the user input topic corresponds to the gesture object;
- retrieval of conversational contributions from the database;
- for each dialogue, book-keeping of which conversation continuations have been used and which ones are still available. The DAs make this information available to the conversation intention planner via the conversation history;
- mini-dialogue structure management;
- extraction of emotional state increments for mini-dialogue structures;
- extraction of values for output template filling.

2.2.2 Mini-dialogue structure processing

Mini-dialogue structures are stored in HCA's knowledge base. When the user and HCA enter into conversation based on a mini-dialogue structure, the mini-dialogue structure is retrieved from the knowledge base and processed by a general-purpose finite-state automaton processor capable of processing any mini-dialogue structure irrespective of its length (or depth) and including any action to be performed in the course of mini-dialogue structure navigation. For instance, a fairy tale mini-dialogue structure representation is shown in Tables 2.3, 2.4 and 2.5.

Table 2.3 shows the input/output references (I/O) at each dialogue structure node as well as the processor actions which are associated with all or certain nodes in the mini-dialogue structure, such as the action of checking the preconditions for entering a new state, or the action of retrieving the emotion increment for a certain HCA reaction to the user's input.

Table 2.4 shows a concordance between semantic input and mini-dialogue structure ids, including informal examples of the user input involved and their corresponding semantic representation.

Table 2.5 shows a concordance between HCA text-to-speech output and mini-dialogue structure ids. Non-verbal output and emotion increments are not shown.

HCA	User	HCA	User	HCA
O-0	I-1	O-1/A1		
	I-2	O-2/A2	I-1	O-3/A8
			I-2	O-8/A9
			I-3	O-4/A10
			I-4	O-9/A11
			I-5	O-6/A12
			I-7	O-3/A13
	I-3	O-3/A3		
	I-4	O-4/A4		
	I-5	O-5/A5		
	I-6	O-6/A6		
	I-7	O-3/A7		

Table 2.3. A fairy tale mini-dialogue structure representation.

Id	User input (informally)	nlu semantics
I-1	No	<User_Opinion:Negative>
I-2	Yes	<User_Opinion:Positive>
I-3	I don't know	<User_Opinion:Negative> <verb:know>
I-4	Yes, ugly duckling	<User_Opinion:Positive> <fairytale:ugly duckling>
I-5	Yes, little mermaid and ugly duckling	<User_Opinion:Positive> <fairytale:ugly duckling> <fairytale:little_mermaid>
I-6	Yes harry potter	<User_Opinion_Positive> <fairytale:non_hca_fairytale>
I-7	Other	Any other semantics

Table 2.4. User input for a fairy tale mini-dialogue structure.

Id	HCA Output
O-0	Did you ever read any of my fairy tales?
O-1	Do you know [The Little Mermaid / The Ugly Duckling / The Princess and the Pea]?
O-2	Do you remember the titles of any of these fairy tales?
O-3	I wrote around 180 fairy tales. That is actually quite a lot. And many of them are known all over the world. Some of the most well-known are The Little Mermaid, The Ugly Duckling and also The Princess and the Pea. You don't know any of them?
O-4	Oh yes, that's the one about [summary]. Do you like it?
O-5	These are quite popular. [fairy tale] is about [summary]. What do you think about it?
O-6	Oh, that is not one of my fairy tales. Tell me more about this, maybe it could give me some inspiration.
O-7	I wrote around 180 fairy tales. That is actually quite a lot. And many of them are known all over the world. Some of the most well-known are The Little Mermaid, The Ugly Duckling and also The Princess and the Pea. You don't know any of them?
O-8	Would you be able to tell me the titles of any of these fairy tales?
O-9	Oh yes, they are quite popular. [fairy tale] is about [summary]. Do you like it?

Table 2.5. HCA output for a fairy tale mini-dialogue structure.

3 Analysis and representation of personality information for HCA

HCA's personality as described in NICE deliverable D1.1 is primarily represented in the HCA character module in two ways, i.e. in:

- HCA's conversational agenda represented in the communication intention planner (CIP);
- HCA's dynamic emotional state calculated in the emotion calculation module (EC).

In addition, the conversational history (CHi) stores HCA's conversational memory (see Figure 1.1). The PT1 conversational history is described in NICE deliverable D5.1a.

3.1 Conversational agenda

The text in this section does not systematically describe the ordering principles for communication intention planner (CIP) processing, such as look at input domain first, then topics, then ... etc. There is no systematic and general description of the CIP processing steps, except incidentally as when we mention that miscommunication has top priority. A systematic description of the CIP processing steps is provided in report D5.1a.

The communication intention planner (CIP) is the core of the HCA mind state agent (MSA), see Figure 1.1. The CIP represents HCA's conversational agenda. Basically, the agenda goals are to:

1. respond to the user's input, and
2. decide on a continuation of the conversation.

A *continuation* is output which serves to continue the conversation rather than leaving all or most initiative with the user. A continuation may be either a question or a statement.

During conversation HCA will, as far as possible, try to adhere to the current topic or domain addressed by the user in order to maintain conversational coherence. Exceptions to this general strategy reflect HCA's more specific conversation agenda goals. The exceptions are the following:

1. reflecting human conversational strategy, HCA assigns top priority to resolving miscommunication problems through meta-communication before proceeding with the conversation unless he is too absorbed in what he is talking about himself, cf. point 6 below. When miscommunication is detected, e.g. due to a low confidence score or need for clarification, and should be handled, the Meta domain takes priority over all other domains. HCA will not try to respond to the subject brought up by the user but continue the conversation through the meta-communication output;
2. HCA wants to talk about the User domain early on in the conversation in order to gather information about the present user before proceeding with the conversation;
3. HCA has a personal preference for talking about the Works domain, in particular his fairy tales. Thus, if his last output was about this domain, he may return to it even if the user tries to bring up a new domain in the most recent input;
4. HCA wants to have talked about all of the domains of User, Life, Works, and Physical Presence before addressing the Gatekeeper domain. The Gatekeeper domain concerns how the user can leave HCA in his study and enter the fairy tale world. Therefore, if the Gatekeeper domain is introduced by the user before the other domains have been sufficiently covered in the conversation, HCA will try to escape that domain by selecting a topic from one of the other domains;

5. there are a few topics which HCA does not want to talk about, e.g. his age. If the user brings up such a topic, HCA will select a different topic or maybe even a different domain for continuation and will not respond to the user's input;
6. if a mini dialogue is ongoing, the user's input is used to determine which branch to take in the mini-dialogue structure. Most of the time, HCA's response will correspond with the domain and topic addressed by the user. However, the implication also is that out-of-mini-dialogue-topic input will not detract HCA from continuing the mini-dialogue. The output at the end of the selected branch serves as the continuation of the conversation;

An important design goal is that HCA should not appear as a senile old man who does not remember what he has said already, so we have a strategy for making sure that he does not use the same continuation twice. Moreover, to ensure conversational variety and liveliness, HCA will only sometimes use a continuation. Thus,

1. if a mini-dialogue is ongoing, the next output specified in the mini-dialogue structure may be viewed as the continuation. No other separate continuation is used;
2. if meta-communication is needed, the meta-communication output may be seen as a continuation. No other separate continuation is used;
3. if HCA is not able to reply to the user's input, a continuation will always be used;
4. if a reply to the user's input can be given, HCA will in about 50% of the cases just give the reply and, in the remaining cases, let a continuation follow the reply. This is done to vary the conversation and make it less predictable.

Finally, a more general strategy has been implemented in order to increase the conversational variety and liveliness of HCA's contributions. For a range of conversational input topics which we expect will be addressed with relatively high frequency in conversations with users, HCA has several different response variations available. Some obvious cases are HCA's opening and farewell greetings (Physical Presence domain), and the mini-dialogue about the User domain, but the strategy has been implemented for other domains as well.

In addition to these cases there will, of course, be cases in which HCA is not able to respond to the user's input. Although the user's input may have been recognised and parsed correctly, HCA obviously does not have a to-the-point reply for each possible input. This follows from the *systematicity* property of the natural language understanding and gesture interpretation modules. For instance, if the linguistic input parser can generate semantics for "I like your fairy tales" (which it can) and HCA's 'mother' is in the lexicon, then the parser will also generate semantics for "I like your mother". However, HCA does not have any to-the-point response to the latter input utterance.

If HCA cannot respond directly and relevantly to the user's input, it is very important that he has available a continuation, as explained above, so that the conversation does not just stop. One of the many issues for investigation during evaluation of the first NICE prototype is the extent to which out-of-domain (OOD) output in the sense just described causes problems for HCA.

3.2 Emotional state

HCA has four main emotions in PT1: happiness, friendliness (default), sadness, and anger. Like humans, HCA can be more or less happy or angry, etc., depending on the user's input.

At any time, HCA's *emotional state* is a vector in a space whose *limiting states* are full anger, full sadness, and full happiness. Friendliness, HCA's default emotional state, is located at the "centre" of that space (Figure 3.1).

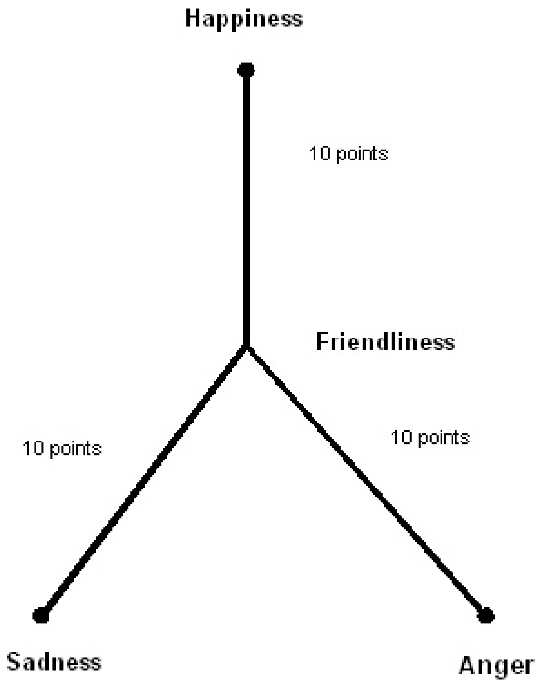

Figure 3.1. HCA's emotional state vector space.

Two processes in the HCA character module affect HCA's emotional state: both the CIP and the mini-dialogues can produce modifications to HCA's current emotional state. These modifications are represented as emotional state *increments* EI: [h: , s: , a:]. Increment values are positive integers from 1 to 10, usually 1 or 2. An EI modification expression either includes a happiness increment, a sadness increment, or an anger increment but never two different increments at the same time.

Each of the limiting states are 10 increments (points) away from the default state of friendliness.

The implication is that HCA's emotional state is either (i) at, or between, friendliness and happiness, (ii) some degree of pure sadness, (iii) some degree of pure anger, or (iv) a frustrated mixture of anger and sadness (Figure 3.1). The emotional state description is ES: [h: , s: , a:] where *either* happiness has a value between 0 and 10, in which case both sadness and anger have the value 0, *or* sadness and/or anger have values between 1 and 10, in which case happiness has the value 0. Examples are ES: [h: 5, s: 0, a: 0] and ES: [h: 0, s: 5, a: 9]. Friendliness is represented as ES: [h: 0, s: 0, a: 0], full happiness as ES: [h: 10, s: 0, a: 0], full pure sadness as ES: [h: 0, s: 10, a: 0], and full pure anger as ES: [h: 0, s: 0, a: 10].

HCA's emotional state is updated prior to each conversational output turn. The key considerations are that HCA's emotional state should change as a function of two factors: the contents of the user's input and the general principle that humans do not remain in their non-default emotional states forever. Rather, non-default emotional states *decay* in a gradual fashion if nothing happens to maintain or increase them, until the human has reached the baseline emotional state (or default mood) characteristic of that particular person.

HCA's emotional state initialises to friendliness at system start-up and after the end of each dialogue.

HCA's emotional state is updated twice in each input-output turn cycle.

Firstly, following HCA's conversational agenda through a series of input processing steps (see Section 3.1 and NICE Report D5.1a), the CIP identifies any emotional increments produced. These are added up to an emotional sum, e.g., EI_sum: [h: 0, s: 2, a: 2]. This sum is then used to update HCA's current emotional state. Factors which may contribute to emotional increments include multiple occurrences of meta-communication, multiple attempts by the user to talk about the Gatekeeper domain before the other domains have been sufficiently covered, insulting input from the user, and input concerning topics which HCA likes or dislikes. In other words, HCA's emotional state is a function not only of the current user input but also of observations on the conversation history (or discourse context).

Secondly, as user input semantics in mini-dialogues may produce emotional increments as well, and as mini-dialogues are being processed after the end of CIP processing, a second emotional state update is performed after the processing of mini-dialogues by the DAs and the KB. It is at this stage that emotional decay is taken into account. If neither the CIP agenda nor the mini-dialogues have produced any emotional increments, HCA's emotional state is incrementally nudged towards the friendly default state, i.e. ES: [h: 0, s: 0, a: 0].

4 Analysis, design and representation of conversation behaviour for HCA

In Sections 2 and 3 we have described some of the core mechanisms which serve to generate HCA's dialogue behaviour, i.e. the domain knowledge representation in the knowledge base (Section 2.1), the domain reasoning in the domain agents (Section 2.2), the conversation agenda in the communication intention planner (Section 3.1), and the emotional state dynamics (Section 3.2). In the present chapter, we describe HCA's PT1 conversation behaviour, following the WP1 description, in terms of:

> *Dialogue behaviour includes the dialogue which can be conducted between the user and the character and its consequences for language understanding, dialogue management, response planning, and domain information and reasoning (WP1).*

Section 4.1 describes the design analysis underlying HCA's conversation behaviour, adding information to the description provided in D1.1. Section 4.2 describes the consequences of the design for language understanding, dialogue management, response planning, and domain information and reasoning. The actual conversation management in the HCA character module is described in NICE deliverable D5.1a.

4.1 Conversation between the user and HCA

4.1.1 Two conversation requirements and their implications

The two main requirements which determine HCA's conversation behaviour are perhaps the following:

1. HCA conducts conversational, domain-oriented dialogue rather than task-oriented dialogue with the user; and
2. the NICE HCA system is not an information system which has the purpose of enabling the user to efficiently obtain large amounts of correct information about HCA. Rather, the system is an edutainment system which, in addition to providing some amount of correct information to users, has the goal of entertaining users through conversation with HCA.

These two requirements have profound implications for the conversation between users and HCA. The implications are outlined in the following.

It follows from (1) above that the user and HCA have no shared task goal(s) to pursue in the dialogue. Rather, the user can in principle say anything of relevance to HCA and expect to get a conversationally appropriate response from the old fairy tale author. Obviously, the notion of “relevance to HCA” is, and must remain, a rather vague one and very different from the notion of task relevance familiar from task-oriented systems. Faced with input of this broad nature, HCA's on-line conversation strategy, just like the conversation strategies of humans, must be based on who he is, i.e. on what he knows and does not know, what he believes, what he wants and desires, on his basic moods, and on how he feels about the conversation situation and its topics. This is what he can conduct conversation about. This is a large and varied lot of information, and, again, a far cry from the succinct and carefully circumscribed information inherent to information systems. Moreover, HCA has no chance of informing his partners in conversation about it to any degree of detail. For instance, it does not make sense for HCA to provide the user with a, spoken, of course, list of the contents of his knowledge base by reference to which users could then organise their conversational contributions.

Requirement (1) thus implies serious uncertainty about the subjects about which the user will engage HCA in conversation.

Another implication of requirement (1) thus is the following. For HCA conversation design, the domain-oriented nature of the conversation means that it becomes extremely important to investigate how the target users, i.e. children and youngsters 10-18 years old, prefer to address HCA and what they like to talk to him about. Only in this empirical way is it possible to design HCA's conversation capabilities in anticipation of the users' dialogue behaviour in conversation. For instance, if it turns out that all or most young users have little interest in discussing even the most well-known HCA fairy tales in depth, there is little reason to include the capability for doing just that in the HCA character module. To investigate the target users' conversation interests and behaviours, NISLab has collected, in two Wizard of Oz (WoZ) simulation cycles, 6 hours (WoZ1) + 30 hours (WoZ2) of spoken conversation data with target users. The results from the second WoZ iteration in July and August 2003 were collected based on the PT1 specification. The PT1 specification, including more than 90% of the output designed for HCA in PT1, was used by the wizards to identify what to say during conversation. The WoZ2 data is being analysed at the time of writing. Most results of the analysis will come in too late for having any profound impact on the PT1 software. These results will be used in the design of PT2.

It follows from education part of requirement (2) above, we submit, that HCA should provide accurate factual information during conversation, including non-verbal, graphical information about how he looked and behaved as well as how his study looked at the time. We have found, however, that the entertainment part of requirement (2) implies that the key to designing HCA's dialogue behaviour lies elsewhere. The key lies in making HCA's dialogue behaviour conform to:

- the entertainment goal(s) of the system; and
- the nature of conversation as opposed to, e.g., task-oriented dialogue.

Our *entertainment goal* for the HCA NICE system is that the target users should leave HCA with a profound impression of having spoken to a very interesting character from another age who, despite all differences due to historical epoch, personal age and experience, is far more like they are themselves than they might have expected.

4.1.2 On the nature of conversation

By its nature, task-oriented dialogue is shared-goal dialogue [Bernsen et al. 1998]. It is dialogue to get the task done. The task-oriented dialogue serves as means for the user and the system to achieve some common generic goal, such as making airline ticket reservations or being trained in operating some piece of equipment. Thus, task-oriented dialogue design aims to specify a dialogue which cooperatively helps users to accomplish their particular task goals all of which are subsumed by the generic task goal. In conversation, whether domain-oriented or completely unrestricted, on the other hand, there are no task goals. Moreover, during conversation, the participants may not pursue any particular goals at all, they may pursue temporary goals of any kind and drop these at any time, or they may have conflicting goals, including goals which conflict with the goals of their partner(s) in conversation. Instead, *successful prototypical human-human conversation* may be characterised by principles, such as the following:

- initially, the interlocutors *search for common ground,* such as shared interests, shared knowledge, and similarity of character and personality, to be pursued in the conversation;
- the conversation is successful because the interlocutors *find enough common ground* to continue the conversation;

- the interlocutors provide, by and large, *symmetrical contributions* to the conversation, for instance by taking turns in acting as domain experts, so that one partner does not end up in the role of passive hearer/spectator, for instance like the novice who is being educated by the other(s); and
- to a significant extent, the conversation is characterised by the participants taking turns in *telling stories,* such as anecdotes, descriptions of items within their domains of expertise, jokes, etc.;
- conversation is *rhapsodic,* i.e. highly tolerant to digression, the introduction of new topics before the current topic has been exhausted, etc.; and
- conversation, when successful, leaves the partners with a sense that it has been *worthwhile.*

For the NICE HCA system, the principles of conversation imply that:

(1) HCA should be prepared to address domains and topics which constitute common ground with his young interlocutors. These will seek common ground when talking to HCA and, if they fail, the conversation will fail as well. To a large extent, what constitutes common ground must be determined empirically (cf. Section 4.1.1). So far, our data suggest that common ground may be found in, e.g., conversation about HCA's fairy tales to the extent, at least, that the users have prior knowledge about them, games played in HCA's days as well as today, technical inventions made at his time and today, flippant humour, etc.

Topics such as games and inventions have the additional advantage that:

(2) they enable the establishment of strong symmetry in the conversation because, just as HCA can delight in explaining his fascination with photography, the young users can establish themselves as symmetrical partners in conversation by explaining topics such as football and computer games to a highly interested HCA.

Moreover, and in complete contrast to task-oriented systems which must follow efficiency principles in dialogue, such as those of being precise, unambiguous, and always to the point:

(3) HCA should often express himself by way of telling stories rather than providing to-the-point information. Finally,

(4) the principles of prototypical successful conversation clearly encourage HCA to digress and to suddenly change topic and domain. This implication is, in fact, a much-welcomed one because it allows the character module to dynamically change topic and domain when HCA is faced with what is to him incomprehensible input. The reasons why this is important to HCA conversation design are many, including: (a) the speech recogniser is likely to have a significant word error rate; (b) the natural language understanding module will make errors; (c) the gesture input chain and the input fusion module will make errors; and (d) the character module will be unable to provide to-the-point conversational contributions to an unknown fraction of well-recognised and well-parsed speech input and well-recognised and well-interpreted gesture input.

4.1.3 NICE HCA conversation design

The NICE HCA PT1 conversation design has been based on the above requirements and principles and their implications. We have to some extent adhered to the educational sub-goal of the system. This has involved making sure that the provided factual information is largely historically correct, including HCA's appearance, the appearance of his study, and the appearance of the specific objects present in his study, cf. Figures 4.1 through 4.5. The educational design sub-goal has also involved the inclusion, in HCA's knowledge base, of various kinds of factual information about his life and fairy tales, even if we, at present, have insufficient knowledge about the extent to which this information will contribute to the common ground principle of conversation. This, of course, has been inevitable in PT1. The

PT1 user testing data will no doubt provide important information on the extent to which we have been right or wrong in our common ground assumptions, story-telling strategy, rhapsodic strategies, as well as in the compromise struck between the entertainment and the educational sub-goal of the system.

Figure 4.1. Photograph of Hans Christian Andersen.

Figure 4.2. NICE Hans Christian Andersen in his study.

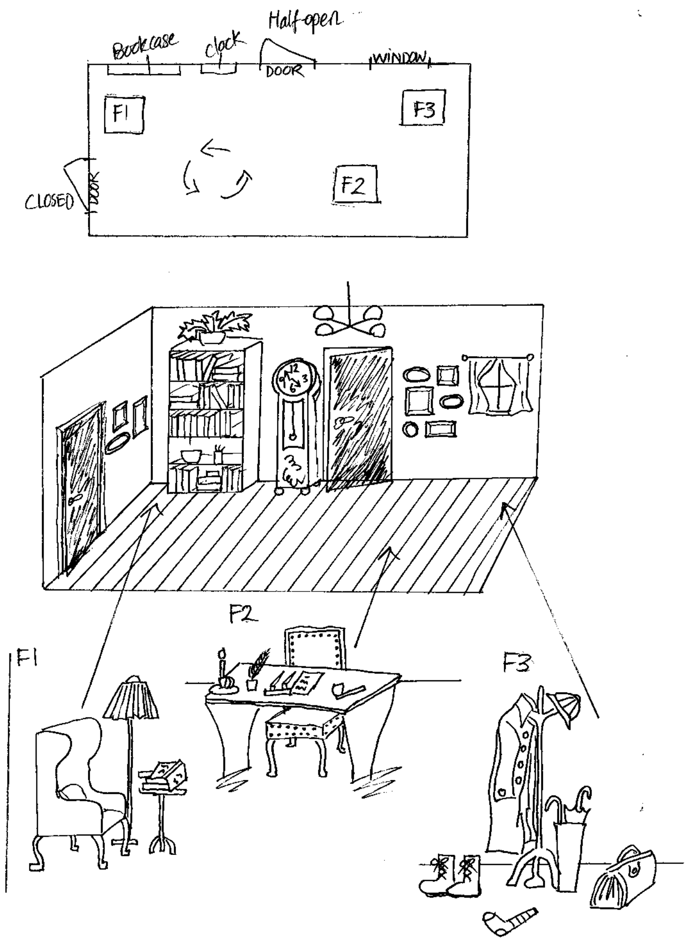

Figure 4.3. Early design sketch of Hans Christian Andersen’s NICE study.

Figure 4.4. Hans Christian Andersen’s study in Copenhagen.

Figure 4.5. Hans Christian Andersen's NICE study.

Finally, it should be emphasised that, since the NICE HCA PT1 is a multimodal input system which accepts both spoken and 2D gesture input, the above design principles for edutainment systems have been followed for multimodal input as well, although to a much more limited extent so far. Thus, during conversation, users can indicate, using gesture, objects in HCA's study which HCA is happy to tell stories about. However, since we have not yet been able to include gesture in our WoZ studies, we have precious little information at this point about the extent to which the gesture part of conversation meets the principles of edutaining conversation outlined above. The pointable objects are historically correct, of course, but are they entertaining? Do they contribute to building common ground? We do not know yet.

4.2 Consequences for language understanding, dialogue management, response planning, and domain information and reasoning

The change of design objectives from task-oriented spoken dialogue to domain-oriented conversation as outlined in Section 4.1 has profound implications for the design of the NICE HCA system's language understanding, conversation management, response planning, and domain information and reasoning functionalities. These implications are described below.

4.2.1 Language understanding

The NICE HCA system cannot expect to receive lexically and grammatically quasi-limited, quasi-circumscribed, and mainly brief spoken input. Conversational contributions may go all over the place and they are not necessarily brief, due to the story-telling aspect of conversation (Section 4.1.2). In fact, the only limitation on input length is imposed by the Scansoft speech recogniser. The only limitations on lexicon and grammar are provided by HCA's domains and by the natural language understanding module's restricted coverage of those domains which, moreover, new users know nothing about at the start of the conversation and the details on which the system cannot communicate to users in any detailed fashion anyway (Section 4.1.1).

The main implication for natural language understanding design is that the parser should be quite shallow lest it will become bogged down in complex syntactic analysis often leading nowhere but most of the time endangering or damaging the system's real-time performance.

To this end, NISLab is developing a new parser based on ATNs and limited syntactic post-processing, able to handle the N-best lists (PT1) and word hypothesis graphs (WHGs, PT2) received from the recogniser. PT1 will not include a separate stochastic topic spotter able to spot the topics in the user's input at a higher level of abstraction than the semantic representations generated by the parser. Rather, in PT1, input topics and, at the highest level of abstraction of all, domains, will be generated from the top-ranked parse. Meanwhile, we are topic-tagging the entire WoZ2 corpus (Section 4.1.1) in order to be able to train the topic spotter for use in PT2.

It may be added that, according to the NICE deliverables plan, the NISLab natural language understanding module will be described in detail in NICE Deliverable D3.5-2 due in Year 3 of the project.

4.2.2 Conversation management

To comply with its conversational design goal, NICE HCA conversation management must be designed for unconstrained mixed-initiative dialogue. Conversation just is like that. There is no way of constraining conversational initiative in order to favour either the dominance of initiative in conversation by HCA or by the user.

Given the quasi-open-ended diversity of domain-oriented conversational input as parsed by the natural language understanding module, the NICE HCA character module (acting as dialogue manager) cannot include global hard-coded dialogue structures like the ones we are used to in task-oriented dialogue managers. Even in the latter, writing realistic, task-complete hard-coded dialogue structures (or models) for complex tasks has become quite hard already, yielding enormous dialogue trees with an increasing number of procedures attached. For domain-oriented conversation, this approach is no longer viable. This conclusion reveals one of the major challenges facing the NICE HCA system, i.e. that of managing domain-oriented conversation.

Our approach in PT1 is to abandon the notion of a hard-coded dialogue structure and replace it with the following functionalities:

1. a HCA character module conversational agenda or planner of the next conversational move;
2. pattern-spotting in the conversation history;
3. constraint-based search in the HCA knowledge base;
4. mini-dialogue structures;
5. intelligent domain agents.

(1), *the HCA character module conversational agenda or planning functionality,* is performed by the conversation intention planner (CIP). The CIP's task is to analyse the user's input domain, topic(s), and, to some extent, semantics, in order to plan how to continue the conversation through HCA's next output. The CIP is described in more detail in Section 3.1 above. See also NICE Deliverable D5.1a.

(2), *pattern-spotting in the conversation history,* is performed by the CIP in order to detect in the conversation history patterns, such as repeated meta communication or repeated efforts from the user to address the Gatekeeper domain in order to enter the fairytale world.

(3), *constraint-based search in the HCA knowledge base,* is performed by the domain agents (DAs) based on the CIP's plan for how to continue the conversation. The constraints are imposed by the user's input domain(s), topic(s), semantics, meta-communication problems, if any, conversation history patterns, CIP priorities, HCA's current emotional state, and any pre-processing done by the domain agent(s) concerned. The knowledge base is described in more detail in Section 2.1 above.

(4), *mini-dialogue structures,* seem to be a new invention which is being used to capture input of particular interest to HCA and enable HCA to conduct in-depth conversation about that input. To some extent, mini-dialogue structure processing works as conventional processing based on hard-coded dialogue structures.

(5), *intelligent domain agents,* are familiar from task-oriented systems in the form of task-specific agents which analyse the user's input per task, pre-process the input prior to consulting external information stores, post-process query results, and report the processed query results to the dialogue structure manager which, using one of the task-specific dialogue structure, identifies the semantic output references to be sent to the response generator [Charfuelàn and Bernsen 2003]. In the NICE HCA system, domain agents have similar pre- and post-processing roles. The main difference is that the NICE HCA DAs simultaneously consult the system's information store *and* its repository of semantic output references. The result is that the DAs themselves send the semantic output references to the response generator via the managing modules for the mind state agent (the mind state agent manager, MSAM), and the character module (the character module manager, CMM), cf. Figure 1.1. The NICE domain agents are described in more detail in Section 2.2 above.

4.2.3 Response generation

Response planning having been described in Section 4.2.2, we now briefly describe the design rationale for response generation in the NICE HCA PT1 system. For more detail, see NICE Deliverable D3.7.

Given the story-telling nature of much of HCA's spoken output, the NICE HCA system cannot adopt an advanced concept-to-speech approach to response generation. The state of the art in natural language response generation does not allow the generation of output at this high level of complexity. Moreover, adoption of an on-line concept-to-speech approach to response generation would demand solution to an additional high-complexity problem, i.e. that of *concurrent* on-line generation of non-verbal behaviour corresponding to the generated, rather complex spoken output. This is far beyond the present state of the art.

To solve the problem of NICE HCA response generation for PT1, NISLab has extended its template-based response generator (RG) to enable handling of combined spoken and non-verbal output and to express, in the process, HCA's current emotional state. The RG receives from the character module some or all of the following (Figure 1.1):

- references to "canned text" spoken and non-verbal output in the RG's Prolog-based response data store;
- references to template-based spoken output, referring to canned text-cum-attribute-variables in the RG's response data store;
- values for the attribute variables in the templates. The RG replaces the variable(s) with the values and ensures grammatical correctness of the surface output language through natural language syntactic processing;
- HCA's current emotional state (ES). The RG uses the ES to fine-tune HCA's non-verbal output using various parameters, cf. NICE Report D3.7.

4.2.4 Domain information and reasoning

HCA character modules domain information and reasoning is described in Sections 2.1 and 2.2 above. At this point, we want to add the following in order to make more explicit the consequences of HCA's conversation behaviour for the NICE HCA PT1 system's domain information and reasoning.

HCA's domain information has a number of properties which are uncommon in standard task-oriented information. HCA's domain information is:

- *personal,* i.e. it expresses HCA's personal view of the world rather than objective information. Obviously, HCA must express his own views during conversation rather than acting as an information automaton;
- extremely *complex,* given the facts that his domains include his life and his fairy tales;
- potentially, *cross-linked* in open-ended ways. We must assume that, in HCA's brain, his life and his fairy tales are interrelated in virtually endless ways.

In our view, it is, if not unlikely, then at least a matter of further research, whether classical AI ontology approaches can be used as foundations for representing HCA's domain knowledge and for making the system able to reason about this knowledge. For instance, a life viewed from the perspective of the person living that life, bears little resemblance to classical micro-world ontology topics, such as lawn-movers or coffee machines. It also bears only modest resemblance to large-scale, well-structured ontologies, such as those of physics, botany, or zoology.

Given this assumption, our PT1 approach is to represent HCA's domain knowledge as an ontology involving several two or three-level hierarchies of the type "has parts or aspects". This is the way in which we represent HCA's domains of Life, Physical Presence, User, and Gate-keeper for the fairy tale world. In addition, for representing his fairy tales which form part of his Works domain, we use reduced versions of a general ontological model of fairy tales in which each fairy tale known to the system is characterised in terms of a set of common attributes, such as: has_author_[name], was_written_in_[year], has_summary_ [summary], relates_to_HCA's_life_[relationship], has_main_character_[name], main_ character_[name]_has_main_property_[property], has_main_object_[object], has_morale_ [morale], is_scary, is_funny, is_sad, etc. We need user trials with PT1 in order to evaluate the appropriateness of these ontologies and analyse ways in which to improve or revise them in order to increase HCA's domain reasoning capabilities.

5 References

5.1 Publications

Bernsen, N. O., Dybkjær, H. and Dybkjær, L.: *Designing Interactive Speech Systems. From First Ideas to User Testing.* Springer Verlag 1998.

Charfuelán, C. and Bernsen, N. O.: A task and dialogue model independent dialogue manager. In Angelova, G., Bontcheva, K., Mitkov, R., Nicolov, N., and Nikolov, N. (Eds.): *Proceedings of the International Conference on Recent Advances in Natural Language Processing* (RANLP-2003), Borovets, Bulgaria, September 2003. INCOMA, Shoumen, Bulgaria 2003, 91-97.

5.2 NICE deliverables

D1.1: Bernsen, N. O. (Ed.), Boye, J., Kiilerich, S. and Lindahl, U.: Requirements and design specification for domain information, personality information and dialogue behaviour for the first NICE prototype. HLT project NICE Report D1.1. NISLab, Denmark, 2002.

D1.2b: Boye, J., Gustafson, J. and Wirén, M.: Formal representation of domain information, personality information and dialogue behaviour for the fairy-tale domain in the first prototype. HLT project NICE Report D1.2b. TeliaSonera, Sweden, October 2003.

D3.3: Martin, J.-C., Bernsen, N.O., Blasig, R., Boye, J., Buisine, S., Dybkjær, L., Fredriksson, M., Généreux, M., Gustafson, J., Lindahl, U., and Mehta, M.: Analysis and specification of cooperation between input modalities and cooperation between output modalities. HLT project NICE Report D3.3. LIMSI, France, 2003.

D3.5-2: English natural language understanding module. HLT project NICE Report (due Month 32).

D3.6: Martin, J.-C., et al.: Multimodal input understanding module for the first prototype. HLT project NICE Report D3.6. LIMSI, France, October 2003.

D3.7: Corradini, A., Gustafson, J., Martin, J.-C., Mehta, M., Bernsen, N.O. and Fredriksson, M.: Multimodal output generation module for the first prototype. HLT project NICE Report D3.7. LIMSI, France, October 2003.

D4.2: Fredriksson, M., Königsmann, J. and Johannesson, L.: First version of the system prototype, inhabited by characters from H. C. Andersen's world. HLT project NICE Report D4.2. Liquid Media, Sweden, October 2003.

D5.1a: Bernsen, N. O. and Dybkjær, L.: First prototype version of conversation management and response planning for H.C. Andersen. HLT project NICE Report D5.1a. NISLab, Denmark, October 2003.