Cyrano goes to Hollywood: a drama-based metaphor for information presentation

Francesca Biral[**], Vincenzo Lombardo[*], Rossana Damiano[*], Antonio Pizzo[**]

[*]Dipartimento di Informatica and Centro di Scienza Cognitiva, Università di Torino
c.so Svizzera 185, Torino, Italy
{vincenzo, rossana}@di.unito.it

[**]Dipartimento di Discipline Artistiche, Musicali e dello Spettacolo, Università di Torino
via S. Ottavio 9, Torino, Italy
{antonio.pizzo, francesca.biral}@unito.it

Abstract

This paper proposes a drama-based interaction metaphor for the presentation of information on mobile devices. The metaphor relies upon an artificial character, that presents information according to a scripted drama. The system architecture is typical of BDI agents, and operates as a script organizer over a repository of elementary audiovisual units (s*cript units*), that can be combined in order to effectively react to the user's inputs. In particular, this paper focuses on the preliminary results achieved in the implementation of the audiovisual fluency, i.e. the seamless combination of script-units, that increases the overall believability of the character. This combination is achieved by adopting Hollywood-style film editing techniques, encoded in a set of parameters and functions that can implement a number of plausible transitions between the Script-units.

1. Introduction

The design of applications for nomadic computing is concerned with accounting for users' localisation and the system adaptability to users' profiles. The issues raised in the nomadic context are actually typical of all the applications dealing with presentation of information to users where the major aspect to be tackled by an application designer is to provide orientation in a potentially huge and not always relevant information space, a design issue that can be addressed by adopting suitable metaphors for human-computer interaction. Viable metaphors can be imported, after a careful craftwork on the communication languages, from the domain of visual and performing arts, domain with which most of the users are familiar (Gershon & Page 2001). In recent years, an increasing attention has been devoted to artificial characters as a communication metaphor, although not for mobile devices: character-mediated presentations are considered to be more compelling, as a *persona* has been shown to have a positive effect on the user's attitudes (Lester et al., 1997). An appropriate and well-calibrated level of emotional arousal conveyed by the dramatic component helps focusing the attention and facilitates the understanding (Picard 1997). Moreover, interactivity with the characters increases their believability and contributes to adapting the presentation to the users' needs (Cassel et al. 2000).
In this paper we describe preliminary research on a system architecture for an artificial character, Cyrano, implemented on a mobile device supported by a wireless network, by using off-the-shelf PDAs and a wi-fi LAN.
The goal of Cyrano (see also Damiano et al. 2003) is to generate interactive, drama-like presentations, acted by an artificial character who inhabits a virtual space situated on a portable device. The deliberation component of the system outputs a multi-modal script that implements the behaviour of characters. The user can interact with the system, so the system may modify its behaviour as a consequence of the interaction, by switching to a new script. The whole methodology is inspired by the *interactive drama paradigm* (Mateas 2001, 2002). The functioning of the system is modelled on the way script writers work: the developer of a drama application first encodes the character's possible behaviours into a set of Scripts, consisting in a chain of Script-units (see below); then the system generates an intentional behaviour by selecting a Script among the available ones.
The specific issue addressed in this paper is the problem of producing a seamless and appealing sequence by concatenating Script-units. This requirement is guaranteed by the adoption and the formalization of Hollywood-style editing techniques (Boldwell & Thompson, 2000) (Reisz & Millar, 1995). Editing techniques are expressed in terms of parameters and constraints that allow the selection of suitable transitions from a pool of possibilities.
The paper is organized as follows. First, we describe the system architecture, called Cyrano; then we address the problem of audiovisual fluency and the solution proposed; finally we illustrate the example application and report on the related work.

2. The system architecture Cyrano

The system architecture Cyrano (see Figure 1) is structured as a cascade of modules that reside on the mobile device and the server. In general, the mobile device accounts for the detection of input events (that are transmitted in symbolic form to the server) and the chain playing of audiovisual clips selected according to the server instructions; all the other modules, that form the decision-taking part of the system, reside on the server. Each module takes into account the output representation of the previous module and the interaction already occurred in the current session (interaction history - IH), and produces a new representation while upgrading the interaction history.
The Input Manager takes as input a symbolic representation of the signals from the Event Detector (user requests expressed via text/speech, tapping, buttons, and the information about the device current location), and converts it into a logic representation. Based on this representation, the Understanding Module identifies the user's goals: for example, the user may desire to obtain more information about a certain item (*information seeking goals*), ask for clarification about a system turn (*interaction handling goals*), or simply signals the willing to attend the remaining presentation. Then, the Goal Manager generates the system goals in response to the identified user's goals by consulting the Goal-KB, a set of rules which select the interactional and dramatic goals of the system based on the interaction context and on the advancement of the drama. The Script Manager, the deliberative core of the system, exploits the knowledge encoded in the Script-KB (a repository of Scripts made of Script-units descriptions, see Figure 2), and the input from the previous modules (Interaction History, System goals) to deliberate about the next selected Script, possibly aborting the currently selected Script, by testing its preconditions. Finally, the Presentation Module inserts a number of suitable transitions (*transition-units*) between adjacent Script-units. Transition-units are retrieved from the Transitions-KB, that contains an organized pool of Transition-unit descriptions. A Script completed with Transitions is sent to the mobile device, where the Action Scheduler retrieves the appropriate audiovisual clips from Script-unit and Transition-unit pool and dispatches it to the multimedia output devices for synchronised execution.
The behaviour of the system (character) results from playing multi-modal Scripts in sequence; Scripts are selected on the basis of the interaction between the system goals and the user's reactions. A Script consists of a sequence of pre-built modules (*Script-units*), a set of pre-conditions (that determine the applicability of the Script) and a set of (non-deterministic) effects (that hold after the Script execution).
The system past behaviour is represented as a bipartite chain of vertices (composed of Script-unit instantiations and user communication acts) and directed connection edges. This structure constitutes the system memory of the interaction, and is called Interaction history (IH). In figure 2, the elements of IH are connected to the corresponding Script-units in the Script-KB by dashed edges. Diamond shapes correspond to the user's acts.

3. The fluency of Script-unit sequencing

In this section we describe the task of the Presentation module, which selects an appropriate Transition-unit that joins the previous and the current Script-units selected by the Script Manager. First, we provide a detailed description of the internal structure of units, and then we illustrate the rules that implement the fluency of Script-unit sequencing. Script units consist in a structural arrangement of multi-modal elements. This description follows a bottom-up fashion: we move from low-level elements (Script-units and their internal components, called clips) to higher-level elements (Scripts).
We call Track any item with a duration, independently from its content. A Multi-track element is a set of Tracks, where all Tracks have the same duration. The content of a Track is provided by a Clip element. We define *Clip* any multimedia item, containing audio-video information, graphics or animation. Clips can be elementary (Simple clips) or non-elementary (Complex clips). Simple clips can be Body-clips, Loopable-clips, Context-clips, and Border-clips (which can be either Start or End-clips), while Complex clips can be Character-clips or Transition-clips (see figure 3).
Script-units, the building blocks of Scripts, are Multi-Track elements composed of one or more Context-clip items, and a single Character-clip. A Script-unit is an elementary behavioral unit featuring a single character (Character clip), plus any additional audio-visual information (e.g. Context-clips provide extra information, like background or sound effects).
Character-clips, that factorize the behavior of one or more characters, convey the drama content. A Character-clip is a sequence composed of a Start-clip, a Body-clip, and an End-clip (see figure 4). The rationale for the tripartite structure of Character-clips is to express the relations of narrative coherence between subsequent Script-units as formal and content relations between End-/Start-clip pairs .

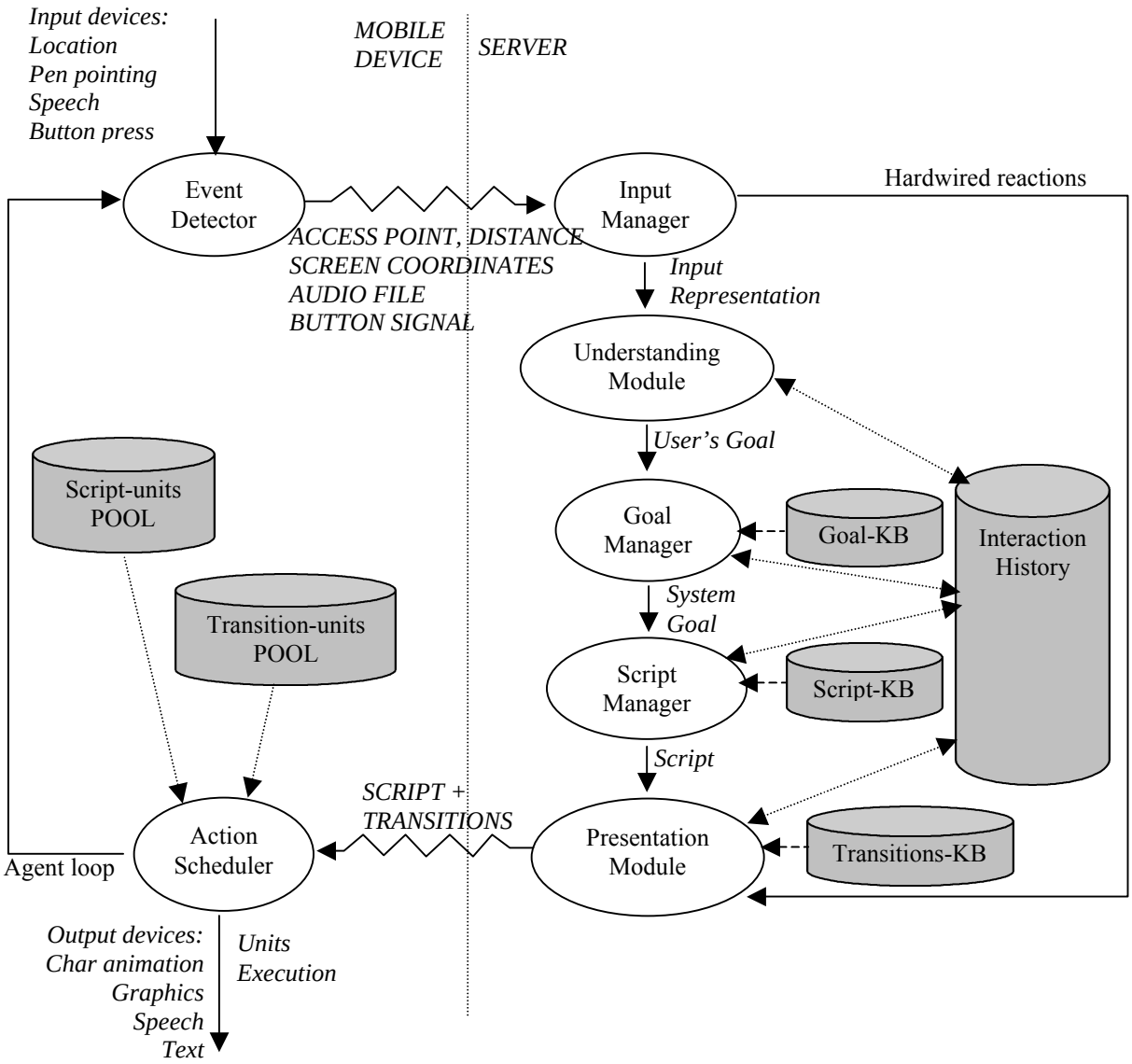

Figure 1. The system architecture.

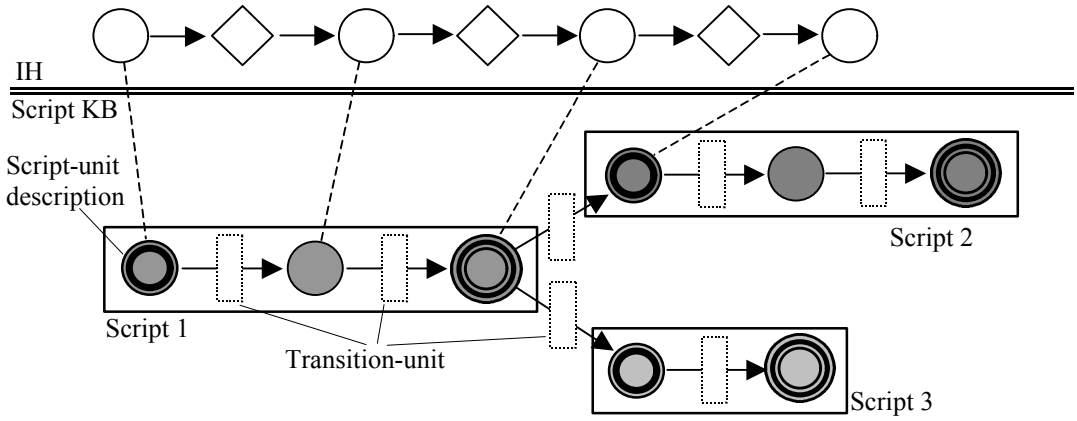

Figure 2. Scripts (linked through preconditions and effects) on bottom, and Interaction history on top. Transition-units are drawn with dashed lines because they are not actually stored in the Script-KB, but are dynamically inserted by the Presentation Module.

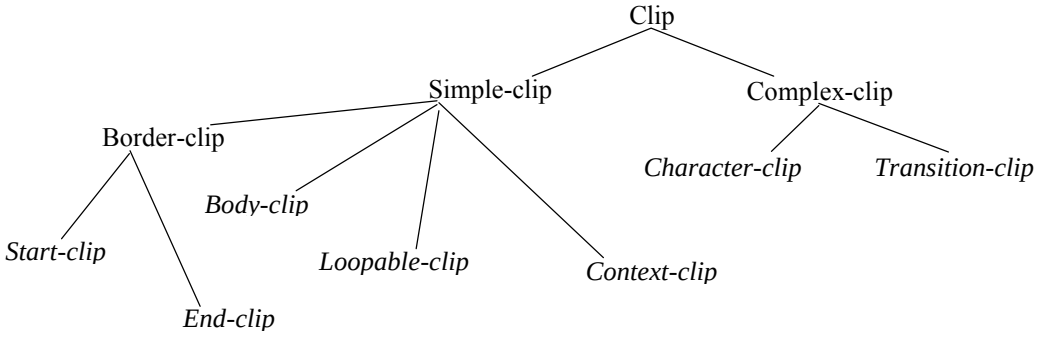

Figure 3. The taxonomy of clips.

Script units are concatenated to form sequences (Scripts in the system terminology) by interleaving Script-units and Transitions-units. The role of Transition-units is twofold: they produce the system behavior while waiting for the user input, and guarantee a smooth transition between subsequent Character-clips. In this way, the developer is not constrained to worry about graphical coherence when designing Character-clips, and Character-clips can be sequenced to form different Scripts with some degree of freedom.
A Transition-unit is formed by a Transition-clip. A Transition-clip also have a tripartite structure: a Start-transition clip (that has to be coherently put into sequence after some End-clip), one or more Transition-body clips (which are Loopable-clips, as they can be iterated if necessary), and an End-transition clip.

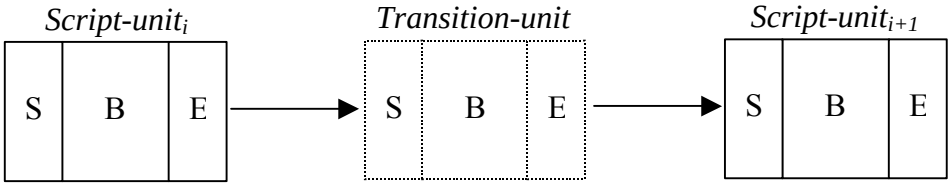

Figure 4. The internal structure of units.

With the aim of obtaining visual fluency, we connect a Script-unit and the following Transition-unit, via their Border-clips, according to a set of rules (and its set of parameters) that we have derived from the standard practice in Hollywood film editing, otherwise described as continuity editing, "a dominant editing style throughout western film history", used to "ensure narrative continuity" by means of a smooth flow from shot to shot in order to convey the illusion of spatial and temporal continuity (Bordwell & Thompson, 2000) (Reisz & Millar, 1995). A number of parameters can be used to describe Hollywood-style editing techniques: for example the graphic qualities (framing, mise-en-scene, etc.) are kept roughly continuous between shots, while the *cut-on-the-action* rule takes advantage of the fact that editing, when motivated by narrative reasons as in the case of a dynamic scene, passes unnoticed by the viewer. Spatial continuity is guaranteed by the 180° axis rule that ensures consistent screen direction, i.e. in the case of a character walking left to right, her/his path constitutes the axis of action and only shots filmed on one side of the axis are acceptable, since crossing the axis would result in the character walking right to left. The 180° system is even more important in the case of two or more characters as it ensures that their relative positions in the frame, as well as eyelines (where the characters gaze), stay consistent, as it happens in the classical shot/reverse shot sequence.
In the partial implementation reported here, we take into account two specific parameters that affect the Border-clips, PROPS and DISTANCE. The PROPS parameter (a prop is any movable articles or objects used on the set of a play or movie) accounts for the satisfaction of the *graphical* continuity between two frames, that includes, cinematographic qualities and the whole of mise-en-scene (lighting, setting, costume, behaviour of figures and presence or absence of props). As in our application Transition-units are retrieved from a large pool according to interaction needs, *graphical* continuity is a crucial aspect in the selection of viable Transition-units. For example, for a character that does smoke, adjacent clips (typically the End-clip of a Script-unit and the Start-clip of the following Transition-unit) are selected on the basis of conforming to the parameter (presence/absence of smoke). The same formalization could apply to all mise-en-scene qualities: costume (presence/absence of jacket or hat), lighting (day or night), setting (presence/absence of objects in the background).
The DISTANCE parameter is concerned with the type of shot, i.e. distance of framing depending on the camera's position. We have taken into account the six main shot types: extreme close-up, close up, medium shot, three-quarter shot, long shot and extreme long shot. Generally speaking, it is regarded as viable and even

artistically appreciable cutting from one shot-type to a different one. However, a straight cut between a long shot and a close-up, given the abysmal magnification order, can be disturbing for the viewer and is thus to be avoided or at least it is normally avoided in classical Hollywood editing practice (although it could be adopted in other contexts, such as music videos). It is equally important to avoid cutting between shots that are too similar, a thing that is perceived as upsetting by viewers.
In the rest of this section we illustrate the definition of parameters that describe each clip, and the production rules that constrain them. Let P_1 be the binary PROPS parameter, which takes the values {0,1}, where 0 indicates the absence of an item and 1 the presence of the item, and P_2 be the six-valued DISTANCE parameter, which takes the values {E.C.U., C.U., M.S., T.Q.S, L.S., E.L.S.}, where E.C.U.= Extreme Close Up, etc. (see terms above). We represent the framing distance associating it with integer values from 1 (Extreme Close Up) to 6 (Extreme Long Shot) where: 1=E.C.U., 2=C.U., etc. .
Now we introduce a relation ***R*** between two Script-units. Given two subsequent Script-units SU_i and SU_{i+1}, we individuate a Transition-unit TU such that the Start-clip of TU is in relation ***R*** with the End-clip of SU_i, the End-clip of TU is in relation ***R*** with the Start-clip of SU_{i+1}. The axioms of the relation ***R*** can be described by the following functions, each associated with a parameter. Both functions check the corresponding parameter, take in input two Script-units and one Transition-unit, and return either an arrangement of the Transition-unit that is compliant with the two Script-units, or nothing (i.e. adopts a default transition). Check-Continuity checks the PROPS parameter; Check-Distance checks the DISTANCE parameter.

```
Check-Props (SUi, SUi+1,TU):
if  P1 is equally valued in the End-clip of SUi and in the Start-clip of TU &
    P1 is equally valued in the End-clip of TU and in the Start-clip of SUi+1
then return (TU)
else return NULL;

Check-Distance (SUi, SUi+1,TU):
if  (P2 in the End-clip of SUi) is equal to (P2 in the Start-clip of TU) +2 OR
    (P2 in the End-clip of SUi) is equal to (P2 in the Start-clip of TU) -2
then
    if (P2 in the Start-clip of SUi+1)is equal to (P2 in the End-clip of TU) +2 OR
       (P2 in the Start-clip of SUi+1)is equal to (P2 in the End-clip of TU) -2
    then return (TU)
    else return NULL
else return NULL
```

The Check-Props function selects a TU such that both Start- and End-clips are compatible (in terms of having or not having the prop) with the Script-units that surround it. The Check-Distance selects a TU where the distance of framing is different by two degrees with respect to the previous or the following Script-unit, thus avoiding contiguous shot being too similar/dissimilar. When all functions yield a NULL result, the Presentation Module selects the default Transition-unit FadeOut-FadeIn, where FadeOut means that an End-clip fades to black and FadeIn means that from black the screen fades in the following Start-clip. In case of multiple eligibility of TU's, a random selection occurs; however, it is possible to devise a set of rules that take into account some contextual factors, including the interaction history (no repetition, visual impact, …).

4. Example application

The approach described in Section 3 has been applied in the development of an electronic guide to an historical site. The artificial character inhabits the virtual space provided by a PDA, from where s/he illustrates the contents of the site, with reference to the current location of the visitor.
The interaction implemented in our project is that of drama, played by the user and an artificial character in the PDA. In the rest of this section, we provide an example of a drama-based presentation of a room in the king's residence, the Billiards Room. So, in the system execution the current location of the user is "the Billiards Room"; the current system goal is "the presentation of the Billiards Room"; this presentation can be conveyed throughout several Scripts.
Now we describe the Script-units of a specific Script, called "PlayBilliards", where the king character Vittorio Emanuele II exhibits a playful and competitive attitude towards the user (L.S., M.S. and C.U. stand for Long Shot, Medium Shot and Close-Up, respectively.). The description of the first Script-unit and the following Transition-unit are in figures 5 and 6 (see the pictures from our Macromedia Flash™ implementation).
We can note that the End-clip of Script-unit 1 and the Start-clip of Transition-unit 1 satisfy both the two parameters P_1 (Continuity: in both clips, presence of the billiards cue) and P_2 (Distance: the type of shot in the two clips is two-degree different), the set of overlapping rules R makes the Action Scheduler prompt the system to play: I) the Script-unit to its end and the Transition-unit 1 from its start and II) both clips one after the other. The Body-clip of Transition-unit1 is a Loopable-clip, that takes into account the user's interaction time (in this

case the system awaits for a positive or negative answer) and the eventual inertia of the system in calculating the best dramatically suited Script-unit to follow in response of the user's selection. In case the user gives a positive answer (he does indeed play billiards), the continuation is given by the Script-unit 2 (figure 7).

5. Related Work

This paper concerns three major aspects of the development of intelligent systems for information presentation in a mobile device context: the use of a drama-based metaphor for HCI, the implementation of animated characters on a PDA platform, the formalization of film editing rules in order to achieve a flow of coherent video sequences.

SCRIPT UNIT 1

CONTEXT CLIP	**CHARACTER CLIP**		
	THE KING ENTERS FROM LEFT AND STOPS IN THE MIDDLE OF THE STAGE		**START**
A WALZER MUSIC.			
BACKGROUND: BILLIARD ROOM TAPESTRY	M.S. OF THE KING WITH BILLIARDS CUE IN HIS HANDS	THE KING DO YOU PLAY BILLIARDS**?**	**BODY**
	THE KING RAISES HIS EYEBROW QUIZZICALLY (M.S.)		**END**

Figure 5 – Start-clip, Body-clip and End-clip for Script-unit 1.

TRANSITION UNIT 1: LOOPABLE CLIP "CHALK"

CONTEXT CLIP	**CHARACTER CLIP**		
	THE KING HAS HIS EYEBROW RAISED QUIZZICALLY (C.U.)		**START**
BACKGROUND: BILLIARD ROOM'S TAPESTRY		THE KING PUTS CHALK ON THE TOP OF HIS BILLIARD CUE	**BODY**
	"EMPTY CLIP"		**END**

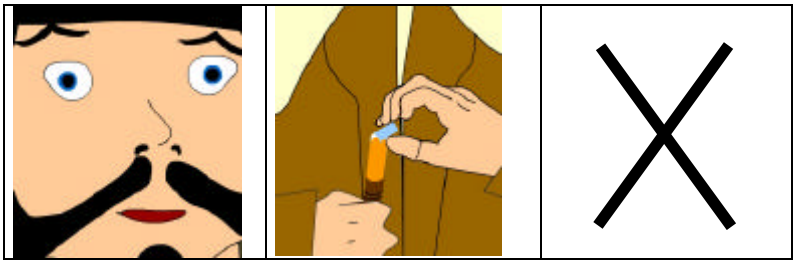

Figure 6 – Start-clip and Body-clip for Transition-unit 1

SCRIPT UNIT 2: POSITIVE REPLY

CONTEXT CLIP	CHARACTER CLIP	
	KING'S PROFILE LEFT TURNS TO FRONT (L.S.)	**START**
BACKGROUND: BILLIARD ROOM TAPESTRY	STOPS AND TALKS STAYING IN PROFILE (M.S.) THE KING DO YOU RATHER PLAY ANY PARTICULAR GAME?	**BODY**
	RESUMES PROFILE LEFT (L.S.)	**END**

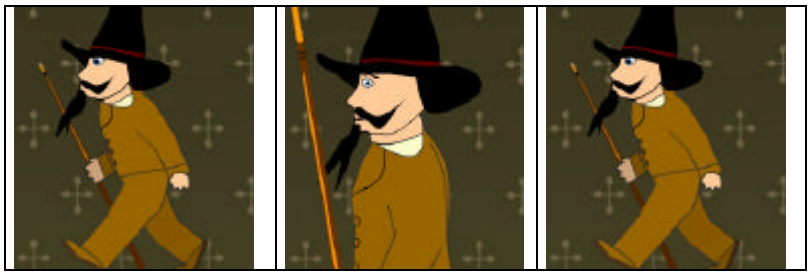

Figure 7 – Start-clip and Body-clip for Script-unit 2

The drama-based metaphor has seen a growing interest in the HCI community in the last decade Animated agents are present in a wide range of different application areas (including education, training, entertainment, e-commerce, …). Andrè & Rist (2001) review a number of systems they have developed for different presentation scenarios. Their analysis points out a range of possible approaches from total scripting (or hard coding) to self-emerging behaviour, even if this dichotomy is actually blurred in the implemented systems, where hard coding of character's behaviour remains unavoidable for practical reasons. Our approach is script-based. Although this choice may introduce a certain amount of rigidity, pre-built material allows an increased believability of characters, in contrast with the poor results of the self-behaviour approach reported by some researchers (Mateas 2001).

PDA applications have received much attention in the museum context (for a general review, see Amirian 2001, Schwarzer 2001). A recent survey of such applications (Biral & Lombardo 2003) has pointed out the general lack of a clear communication metaphor. Prototypes, dating as far back as 1993, have so far proved challenging and confusing for the user, even when familiar metaphors were borrowed and modified (see Cheverst et al. 2000a, 2000b and Fleck et al. 2002 for an evaluation of the browser metaphor). There are, however, a few exceptions to this common trend: for example, *SottoVoce* (Grinter et al. 2002) adopted an innovative perspective by giving the electronic guidebook a conversationalist role: the prototype permitted eavesdropping on the audio content of a companion's guidebook when wanted, and this enhanced socialization. More recently, the PEACH project (Rocchi and Zancanaro 2003), which follows previous research on adaptive hypermedia (Petrelli et al. 1999), features a cinematic metaphor: sequences of camera movements (pans, tilts, etc.) are applied to e repository of semantically annotated still images and synchronized to an audio commentary. In one of the latest versions the commentary is interpreted by a "cyberella"-type agent explicitly aimed at attracting a younger audience and used as a "mediator" to direct the user's attention in the case of "multi-device" presentation performed on both PDAs and large displays in the museum space (Kruppa and Krüger 2003). In this prototype the agent has an ancillary role in the presentation strategy, rather than, as in our work, being the core of it.

For the Hollywood-style fluency in video editing we have derived inspiration on automated editing techniques from work done in systems that support interactive narration, like SoftCinema (http://www.manovich.net) and Korsakow (http://www.korsakow.com). Both approaches attribute video segments a few keywords that mostly describe their *pictorial* qualities; then editing rules rely upon these qualities. In our case, editing meets the necessity of portraying a well-paced and entertaining dramatic action, thus achieving a *fine cut* approach as opposed to the *rough cut* implemented in (Nack & Parks 1995).

5. Conclusions

This paper has presented some preliminary results on a drama-based metaphor for information presentation on mobile devices. The presentation of information is scripted in terms of elementary units (called Script-units) that are assembled in larger units called Scripts. The system displays information by selecting the Script that better contributes in achieving the system goals. This paper has shown a solution for guaranteeing the audiovisual fluency of the sequencing of Script-unit in the display of the presentation. The method encodes the Hollywood-

style film editing techniques in parameters and functions that operate on Script-units and the Transitions pool. This system has been applied in the design of an electronic guide application for an historical site environment, where a theatre metaphor is widely recognized as a useful mean to overcome the squalor generally met in cultural heritage visits (Maure, 1995) (Waterfield, 1998).

Acknowledgements
The authors thank Paolo Olocco for his collaboration in the implementation of the prototype.

References

(Amirian 2001) S. Amirian, Hand-held Mobile Computing in Museums. A Background Paper, 2001 Web link: http://www.cimi.org/whitesite/AmirianBJM.htm.

(Andrè & Rist, 2001) E. Andrè, Thomas Rist, Controlling the behaviour of animated presentation agents in the interface: scriopting versus Instructing, in AI magazine, AAAI press, vol. 22, no. 4, pp.53-66, 2001.

(Biral & Lombardo 2003) F. Biral, V. Lombardo, Ten years of hand-held devices and wireless applications in museums: a review, submitted to the International Journal of Human-Computer Studies, Special Issue on Human-Computer Interaction Issues in Mobile Computing, July 2003.

(Bordwell & Thompson, 2000) D. Bordwell, K. Thompson, Film Art, McGraw Hill, 2000.

(Cassell et al., 2000) J. Cassell, J. Sullivan, S. Prevost, E. Churchill, Embodied Conversational Agents, The MIT Press, Cambridge (MA - USA), 2000.

(Cheverst et al. 2000a) K. Cheverst, N. Davies, K. Mitchell, A. Friday and C. Efstratiou, Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences, Proc. of CHI 2000, Netherlands, April 2000, pp 17-24.

(Cheverst et al. 2000b) K. Cheverst, N. Davies, K. Mitchell and C. Efstratiou, Using Context as a Crystal Ball: Rewards and Pitfalls, Proc. of Workshop on 'Situated Interaction in Ubiquitous Computing' CHI 2000, April 2000.

(Damiano et al. 2003) R. Damiano, V. Lombardo, F. Biral, A. Pizzo, Cyrano: a character-centered architecture for interactive presentations, to appear in the Proc. of HCI-Italy 2003.

(Fleck et al. 2002) M. Fleck, M. Frid, T. Kindberg, R. Rajani, E. O'Brien-Strain, M. Spasojevic, From Informing to Remembering: Deploying a Ubiquitous System in an Interactive Science Museum, Technical Report HPL2002-54, HP Labs Palo Alto, 2002.

(Grinter et al. 2002) R. E. Grinter, P. M. Aoki, A. Hurst, M. H. Szymanski, J. D. Thornton and A. Woodruff, Revisiting the Visit: Understanding How Technology Can Shape the Museum Visit, in Proc. ACM Conf. on Computer Supported Cooperative Work, New Orleans, LA, Nov. 2002, 146-155.

(Gershon & Page, 2001) N. Gershon, W. Page, What Storytelling Can Do for Information Visualization, Communications of the ACM, p. 31-37, vol. 44(8), 2001.

(Kelso et al. 1992) M. Kelso, P. Weyhrauch, J. Bates, Dramatic Presence, in PRESENCE: The Journal of Teleoperators and Virtual Environments, Vol.1 (1), 1992, pp. 133-8.

(Kruppa and Krüger 2003) M. Kruppa and A. Krüger, Concepts for a combined use of Personal Digital Assistants and large remote displays , Proceedings of SimVis 2003, SCS Verlag, 2003.

(Lester et al., 1997) Lester, J., Converse, S., Kahler, S., Barlow, S., Stone, B., Bhoga, R.: The Persona effect: Affective Impact of Animated Pedagogical Agents, in CHI'97 Electronic Publications, 1997

(Maure 1995) M. Maure, The Exhibition as Theatre, On the Staging of Museum Objects, Nordinska Museologi, 1995, pag. 4.

(Mateas 2001) M. Mateas, M. Leonardo, Expressive AI: A hybrid art and science practice in Journal of the International Society for Arts, Sciences, and Technology, 34 (2), 2001. 147-153.

(Mateas 2002) M. Mateas, Ph.D. Thesis, Interactive Drama, Art, and Artificial Intelligence, 2002.

(Nack & Parks 1995) F. Nack, A. Parks, AUTEUR: the creation of humorous since using automated video editing, in IJCAI95, Workshop on AI and Entertainment and AI/Alife, 1995.

(Petrelli et al. 1999) D. Petrelli, E. Not, M. Sarini, O. Stock, C. Strapparava and M. Zancanaro, HyperAudio: Location-Awareness + Adaptivity, in Proceedings of CHI'99, Conference on Human Factors in Computing Systems 1999, pp. 21-22.

(Picard 1997) R. Picard, Affective Computing, The MIT Press, 1997.

(Reisz & Millar, 1995) K. Reisz, G. Millar, Technique of Film Editing, Focal Press, 1995.

(C. Rocchi, M. Zancanaro 2003) C. Rocchi, M. Zancanaro. Generation of Video Documentaries from Discourse Structures. In Proceedings of Ninth European workshop on Natural Language Generation. Budapest, April 2003.

(Schwarzer 2001) M. Schwarzer, Art & Gadgetry. The Future of the Museum Visit, Museum News, July/August 2001.

(Waterfield 1998) G. Waterfield, The Modern Visitor and the Historical Palace: is understanding possible? in "Abitare la storia" Conference papers, ed. Umberto Allemandi, Turin, 1998.