Hermes Service: Distributed Hypermedia Educational Services on Demand

I.Antoniou, Computer Engineering and Informatics Department
University of Patras, 26500 Rion, Patras, Greece, antoniou@ceid.upatras.gr

C.Bouras, Computer Engineering and Informatics Department
University of Patras, 26500 Rion, Patras, Greece and Computer Technology Institute
Kolokotroni 3, 262 21 Patras, Greece, bouras@cti.gr

V.Kapoulas, Computer Engineering and Informatics Department
University of Patras, 26500 Rion, Patras, Greece, and Computer Technology Institute
Kolokotroni 3, 262 21 Patras, Greece, kapoulas@cti.gr

D.Karageorgopoulos, Computer Engineering and Informatics Department
University of Patras, 26500 Rion, Patras Greece, karagd@ceid.upatras.gr

D.Moiras, Computer Engineering and Informatics Department
University of Patras, 26500 Rion, Patras Greece, moiras@ceid.upatras.gr

V.Ouzounis, Computer Engineering and Informatics Department
University of Patras, 26500 Rion, Patras, Greece, ouzounis@ceid.upatras.gr

P.Spirakis, Computer Engineering and Informatics Department
University of Patras, 26500 Rion, Patras, Greece and Computer Technology Institute
Kolokotroni 3, 262 21 Patras, Greece, spirakis@cti.gr

Abstract: In this paper we present a distributed HypERMEdia SERVICE (HERMES SERVICE) which supports integrated capabilities for distance education and learning purposes. Hermes service consists of hermes servers and hermes browsers. The users are connected with the service using hermes browsers and retrieve hypermedia lessons organised in thematical units. The lessons are stored in various distributed Hermes servers which retrieve the requested lesson and transmit it to the corresponding browser on on-demand basis. The requested lesson is presented to the user according to a well defined presentation scenario. The media objects are rendered in a synchronized manner according to the spatio-temporal relationships existing among them. Finally, special monitoring mechanism controls the state of the network and adjusts the presentation process so as to improve presentation quality.

1. Introduction

During this last decade we have experienced a new trend towards teletraining. The need for teletraining systems has been mentioned from the early eighties, but with the advent of this decade, the evolution of multimedia technology in conjunction with the new perspectives that the networking technology has brought, provided new potentials for the development of teletraining systems.

This new challenge will play a significant role in the near future, as it will be possible to access education, training and retraining whenever and from wherever required by the learners. This novel way of training motivated us to design and implement this distance education service. The scope of the service is to provide to the user a set of functionalities for retrieval and browsing of lessons which are stored in various distributed servers..

Our proposed service provides a number of advantages which can be summarised in the following points :

- *transparency*: the user focuses on the service and the provided lessons, while internal information regarding the distributed organisation of the service and the interconnection of the various computer systems over the network are hidden
- *ease of use*: an integrated and friendly graphical environment, implemented in various platforms, provides the user the necessary capabilities to interact with the service
- *on-demand delivery*: the requested hypermedia lessons are transmitted on on-demand basis and not on store-and-forward mode, and as a result good performance is achieved, while the needed resources are reduced [C.Bouras et al. 1996b]
- *resource management*: special monitoring mechanisms are included in the service in order to detect network bottlenecks and presentation anomalies. Using these mechanisms the needed resources are dynamically reserved [C.Bouras et al. 1996b]
- *integrated presentation environment*: the presentation of hypermedia documents is done in a well structured environment without the external invocation of presentation "players" [R.Steinmetz 1990]
- *search capabilities*: the user may search for topics of interest and retrieve the matched items avoiding the selection of a particular lesson
- *subscription-authorisation mechanisms*: the user must be member of the service in order to have access on it. Therefore, non-authorized users are not permitted to use the service
- *compatibility with the WWW and e-mail facilities*: the user may also browse html documents, send and receive e-mails. This capability is considered useful in order to communicate with the constructor of each lesson and ask additional information on themes of interest
- *flexible and modular design*: the design of the service is performed having in mind the embodiment of new functionalities and the provisioning of variety of media formats. The followed design approach is the object-oriented.

2. Architecture Overview

The architecture of Hermes service is based on the client-server model. The basic components that comprise it are the following ones:

- *Hermes server* where the lessons are stored. For every lesson a presentation scenario is associated. The presentation scenario of a lesson actually describes the spatio-temporal relationships among various media objects. The inline media objects are stored in the corresponding media servers. When a request for a specific lesson is triggered, the specific server where that lesson resides on, is contacted. The server searches the database of provided lessons and locates the requested lesson. When the presentation scenario is found, Hermes server activates the involved media servers in order to start the transmission of the corresponding media objects which are referred to the presentation scenario on on-demand basis.
- *Media servers* in which media objects are stored. For every media object (e.g., text, image, audio, video, etc) a media server is associated with each Hermes server. These media servers may be located in the same host with the Hermes server and each one is responsible for transmitting a certain media type through a parallel connection which is established between the browser and the corresponding media server. The involved media objects are transmitted from the media servers towards the browser according to the presentation scenario and the presentation constraints. The transmission process of each media object is adjusted according to the feedback reports which are sent by the Hermes browsers and describe the status of the network and the presentation process.
- *Hermes Browser* is the integrated graphical environment through which the user interacts with the service by retrieving and browsing lessons. The browsers may run on a multimedia PC and facilitate all the provided functionalities of the service. The browsers also manage the synchronized presentation of the requested hypermedia lesson. Internal mechanisms monitor the status of the network and report the network and synchronization anomalies to the corresponding Hermes server and media servers in order to adapt the transmission rates. In that way, the presentation quality is significantly improved while the needed resources are dynamically reserved.

3. Functional Description

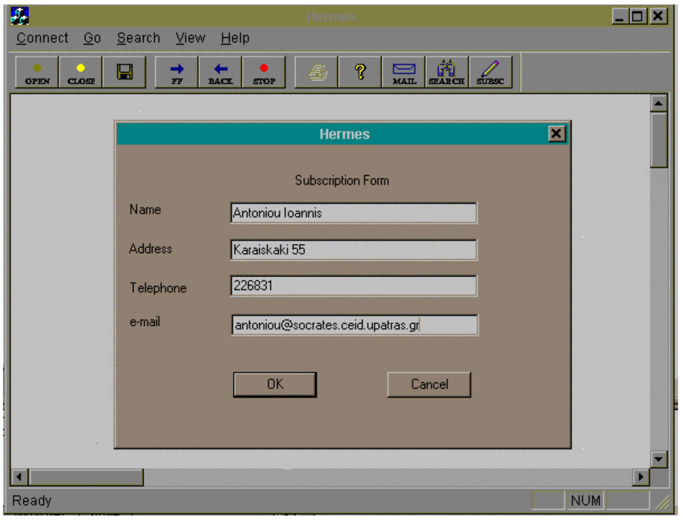
Since Hermes is a service designed to be used from users, we focused on *simplicity* and *functionality*.

In the sequel, the basic features that make Hermes Service a useful and attractive service for educational purposes are introduced.

- Connection-Subscription with the service
- Search for related topics
- Viewing a lesson
- Off-line interaction between the tutor and the users

3.1 Connection-Subscription with a Hermes Server

The connection primitive is the first action that the user must follow in order to access the service. Initially, the user must specify the Hermes server that desires to connect to. For that reason, a list of available Hermes servers is provided. For every Hermes server, a small description concerning the kind of lessons that are stored in it, is presented. Every Hermes server contains lessons concerning specific and well known thematical units. Therefore, the user may easily access the server of his preference.
When the user specifies the Hermes server that desires to connect to, an authentication process takes place in order to verify whether the specific user is authorised to use or not the service. If the user is not an authorized one, he is requested to subscribe to the service filling a subscription form as it is depicted in Figure 1. The subscription form contains various information, such as the real name, the address, the postal and e-mail address and telephone number of the user. A coherent, centralized database of authorized users is updated, and from now on the user may easily access the service. Whenever a user is connected with the service, the database of the authorized users is updated and specific information about the exact time logged in the service, as well as, the lessons that are retrieved are captured.



Subscription Form

Name: Antoniou Ioannis

Address: Karaiskaki 55

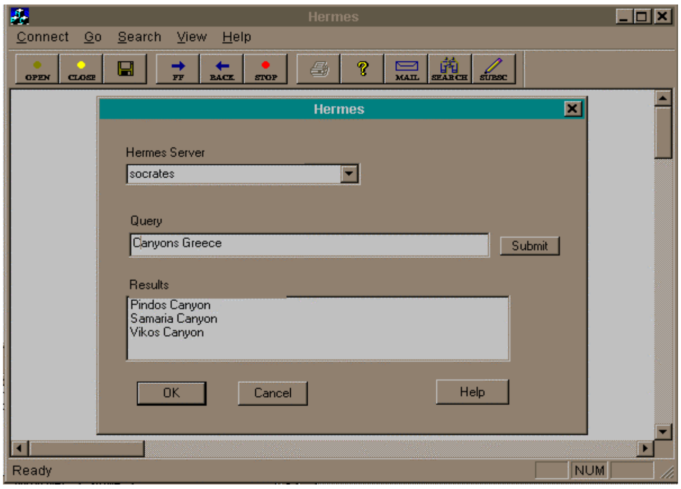
Telephone: 226831

e-mail: antoniou@socrates.ceid.upatras.gr

Figure 1: The Subscription Dialog Box

3.2 Search for Related Topics

In case the user is interested in a certain topic, he is able to search the contents of the *whole* service by invoking the search primitive. In particular, the user specifies the search token which best describes the topic of interest, and selects the server that is likely to contain lessons on the topic. The user selections are transmitted to the specified server and in the sequel, all the text documents stored in that server are scanned. Additionally, this particular server sends the query to all other Hermes servers for the same reason. In every server the same procedure is performed. The results of the query on every server are forward to the initial server and then directly to the user. It must be noticed that only the lessons which contain the item of interest and the server location are transmitted and presented to the user. In that way, the user may search the contents of the service and retrieve only specific lessons on particular topics, avoiding the navigation through all the provided lessons. In addition to that, the user using a friendly dialog box, as shown in the following Figure 2, can specify the search options .



Hermes Server: socrates

Query: Canyons Greece

Results:
Pindos Canyon
Samaria Canyon
Vikos Canyon

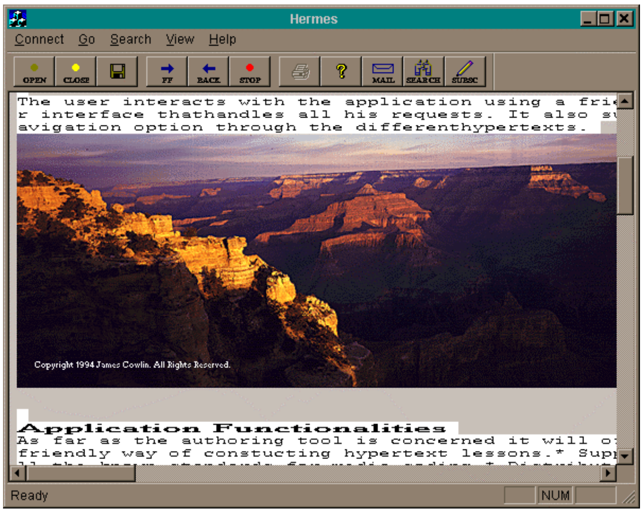
Figure 2: The Search Dialog Box

3.3 Viewing a Lesson

After the connection has been established, a list of available lessons which are stored on that server are presented to the user. The user may select a specific lesson from the list by "clicking" on the corresponding hyperlink. In that case, the request is forwarded to the server. The server upon request retrieves the presentation scenario of the lesson and determines the inline media objects as well as the spatio-temporal relationships among them. For every media object referred in the presentation scenario, the server determines the media server at which it is stored, and specifies the transmission requirements for that object. Consequently, all the associated media servers are contacted and the transmission requirements for every media object are explicitly determined. Every media server, builds a parallel connection with the browser, retrieves the media object, transmits the data on on-demand basis based on the transmission requirements. During the transmission of the media objects feedback reports are sent to the involved media servers and the Hermes server in order to adjust the transmission rates or the enconding format.

At the browser site, the presentation scenario of the requested lesson, which is sent by the Hermes server, is scanned and the needed resources are reserved. Every media object is temporarily stored inside the buffers, and then is presented to the user according to the presentation scenario [Chun-Chuan Yang and Jay-Hsiung Huang 1996], [Herng-Yow Chen and Ja-Ling Wu 1996]. The browser during the transmission of the hypermedia lesson, collects information about the network status and the presentation quality, i.e. the presentation anomalies that may occur.

From the application point of view, the user may follow sequential or explorational links. The sequential links specify a coherent view of the lesson, i.e. the tutor's way, while the explorational links are relative to theme hypermedia documents with additional information on specific topics. In both cases, the browser manages the user selections in the same way. If the selected lesson or topic is stored on another server, a suspend connection primitive is invoked, and a request for a new connection with the new server is triggered. The suspended connection remains active for a period of time in case that the user requests to view a previously selected document. When this interval is passed, the connection closes and the attached client is informed about the event. At that time the client has an active connection with the new server. At any time the user may disconnect from the service by "clicking" on the appropriate button.



The user interacts with the application using a frie
r interface thathandles all his requests. It also su
avigation option through the differenthypertexts.

Application Functionalities

As far as the authoring tool is concerned it will o
friendly way of constucting hypertext lessons.* Supp

Figure 3: The Browsing of a Lesson

Among the several facilities that can be supported by the browser are the following ones:

- Moving *back* and *forward* in the list of already viewed lessons. This can be achieved with the use of menu buttons.
- *Scrolling* of the hypertext document.
- *On-line help* provision.
- *Interactive* operations can be triggered during the presentation of the lesson. The user can pause the presentation of the lesson and resume later from the point that it was paused via control buttons.

In

Figure 3, an example of the browsing process is given.

3.4 Off-lineInteraction Between the Tutor and the Users

In case the user wishes to make comments or submit questions concerning a lesson, he is able to use the e-mail service offered by Hermes. The user can send e-mail to the tutor asking for additional information about a theme. In that case, the tutor can send replies to the user prompting him/her to retrieve specific lessons from the service. Using this off-line mechanism an immediate connection between the user and the tutor is achieved and as a consequence, the educational activity is improved. Therefore, the user may avoid the useless navigation through relative but not specific lessons of interest.

4. Implementation Issues

The Hermes server has been developed under the Unix operating system using the facilities offered by OSF/Motif, Xtoolkit and Xlib. On the other hand, the Hermes browser has been implemented both under the Windows 95 operating system using Visual C++ version 4.0 and the Unix using OSF/Motif. The implementation of the network support primitives that are needed for the realisation of the on-line operation is based on the TCP\IP protocol suite. The hypermedia presentation scenario is transmitted through TCP connections, the non time sensitive media objects (e.g., text, images, graphics) are sent under TCP connections, and the time sensitive media objects (e.g. audio, video) under UDP connections. We utilize the UDP connections for the time sensitive data in order to avoid the additional delays introduced by re-transmissions. Although the UDP is not a reliable protocol, it seems that adequately meet the requirements for real-time sensitive data. In addition to that, the network connections for the control data are transmitted through TCP connections. The underline network infrastructure is an 100 Mbit/s FDDI ring which is able to support on-demand service. Despite the fact that our service does not based on quality of service guaranteed networks, the adaptive nature of the service in conjunction with the network monitoring facilities may significantly improve the presentation quality.

Furthermore, we used the Real-Time Protocol -RTP- [H.Schulzrinne 1995] as a transport vehicle suitable for transmitting time sensitive data. The RTP\RTCP resides between the transport and the application layer and are designed to be independent of the underlying transport and network layer. We used the RTCP protocol for the transmission of the control data, and the RTP protocol for the transmission of the presentation scenario and the inline media objects. The RTP protocol provides end-to-end delivery services which include payload type identification, sequence numbering, timestamping and delivery monitoring. The RTP itself does not provide any mechanism to ensure timely delivery or other quality-of-service guarantees, but relies on lower-layer services to do so. The RTP control protocol (RTCP) is used to transfer feedback monitoring reports concerning the network status and the presentation anomalies.

For the description of hypermedia lessons we implement a hypermedia, html-like markup language with simple synchronization primitives. Using that language the tutor can explicitly specify the layout of the hypermedia lesson and the spatio relations between media objects. In particular, special keywords for the text formation, the figure and graphics placement are provided. In addition to that, the constructor of the lessons may also specify the temporal relations among the inline media objects, like annotated text.

The system has been installed and successfully tested on a variety of SUN SPARCstations and PCs with different memory and processing power characteristics. The system can efficiently operate over LAN and WAN networks.

5. Conclusions and Future Work

Hermes service can support the distance education process bridging the gap that is created by the geographical distances that separate users from tutors. Its real-time characteristics along with the attractive user-interface make it a useful service for realising the purpose of distance learning.

Hermes service provides an integrated graphical environment for the synchronised presentation of hypermedia lessons which are stored in distributed Hermes servers. Additionally, the included searching capabilities and e-mail facilities constitute our service an integrated educational environment which helps the user to acquire the topics of interest with a friendly and transparent way. Finally, Hermes browsers can present html documents, and as a result compatibility with the WWW is achieved.

In the future, we intend to incorporate special mechanisms for efficient information retrieval, to support the presentation of MHEG hypermedia-like documents, and to implement network interfaces for quality of service guaranteed networks like the ATM.

6. References

[C.Bouras et al. 1996a]. An Interactive Cooperative Teleworking Environment: Telemathea, ED-Telecomm 96, 1996, Boston,Mass., USA; June 17-22, 1996.

[C.Bouras et al. 1996b], An On-Demand Multimedia\Hypermedia Service over Broadband Networks, 1996, 5th IEEE International Conference on High Performance Distributed Computing, Syracuse, N.Y., USA.

[R.Steinmetz 1990] Synchronization properties in a multimedia system, IEEE Selected Areas In Communications, 1990, vol8 ,no3, 401-412.

[Chun-Chuan Yang and Jay-Hsiung Huang 1996], A Multimedia Synchronisation Model and its implementation in Transport protocols, 1996, IEEE J. Select. Areas on Communications, vol14 ,no1, 212-225.

[Herng-Yow Chen and Ja-Ling Wu 1996], Multisync : A synchronization model for multimedia systems, 1996, IEEE Journal on Selected Areas in Communications, vol 14, No.1.

[H.Schulzrinne 1995], RTP: A Transport Protocol for Real-Time Applications, 1995, Internet Draft Standard, Internet Engineering Task Force.

Acknowledgements

This research work is partially funded by the Telematics Application Programme under the project TRENDS-BT/4024 and the Greek Ministry of Education