

# Context-Informed Adaptive Hypermedia

Alexander O'Connor, Vincent Wade, Owen Conlan

Knowledge and Data Engineering Group

Computer Science Department

Trinity College, Dublin

Ireland

+353 1 608 1335

{oconnoat, Vincent.Wade, Owen.Conlan}@tcd.ie

## ABSTRACT

Adaptive Hypermedia Systems have mechanisms for extensively modelling their domain of expertise, and reasoning over that domain to produce tailored web content. In general, these systems are supplied in advance with deep, explicit models of the relevant axes of adaptivity. Current designs in Adaptive Hypermedia therefore attempt to model the entire sphere of experience of the system deeply and without separation. Contextual input, on the other hand, is arbitrarily defined, and deep models are not separated from 'shallow interest' contextual data. It is desirable to model this data separately so as to reduce the size of AH models and to encapsulate low-level sensor data as high-level interpreted concepts. This paper presents a novel method for applying contextual information to an Adaptive Hypermedia eLearning system, providing a clear boundary between 'core' adaptive axes and contextual information. An outline of the architecture of an integrated Context Interpretation system with an Adaptive Hypermedia System is provided, along with analysis of the mechanisms by which contextual information may be applied to the specific case of eLearning, and in more general systems.

## Keywords

Context, Adaptive Hypermedia, eLearning

## INTRODUCTION

Adaptive Hypermedia [1] (AH) systems are characterised by having tailored, specific models of their user and domain of expertise. These models are correlated through the system's internal logic so as to create a tailored web experience for a specific user on a specific topic.

Of particular interest in this field is the educational application of Adaptive Hypermedia. This is due to a number of reasons, including availability of content and learner modelling techniques, combined with a high-

involvement user experience, where there is considerable scope for fine-tuning.

There are a number of projects on-going in this area; these include long-established systems such as AHA! [2] and Interbook [3], as well as more recent, distributed projects such as KnowledgeTree [4]. These systems are capable, to varying degrees, of tailoring content and guiding navigation of content in a specific domain, based on information about the user.

The process of adaptation is governed by a given rule set, which provides the basis for:

- Selecting and ordering course information.
- Presentation of the material is also standardised by the system, conforming to an overall sense of unity.

In particular, the KnowledgeTree system provides a useful example of a modern Adaptive Web-based eLearning System. KnowledgeTree consists of a network of content and activity servers, which are correlated with the information stored in a student model server at a learning portal. This architecture provides for advanced features such as pass-through interpretation of content and automatic processing of annotated content.

However, the vision of the User model is one of a central, single service, located on the service terminal. No method is provided for the learner portal to access information about the learner not directly modelled by the Service. It is argued that this will lead to significant burden on the system in the event of the course being equipped with, for example, more detailed learning histories, or sensor information acquired from a pervasive computing environment. These are examples of 'shallow' information; as opposed to the core 'deep' models currently used in AH. The difference can be characterised by determining the level of importance to the adaptation process of the particular information. In a standard eLearning application, terminal information does not affect the selection of content in a core manner. However, it may be of 'shallow' interest to select, for example, static images rather than animated examples for the purposes of conserving limited processing power on a portable terminal.

There exist a large number of example applications in the field of context-aware systems. From relatively early systems [5] on, the main consideration has been of relatively simple applications. An application of particular interest is the Location-aware messaging system, as characterised by ActiveCampus [6].

While these applications provide significant, useful information on the practicalities of Context-aware systems, it is the opinion of the authors that they have not yet addressed an important aspect of context by their relatively simple business logic.

An educational adaptive hypermedia system provides a ready example of the integration of contextual information with a system already performing significant reasoning and modelling tasks. This area therefore provides rich potential for the integration of contextual information into a system with detailed models and logic of its own.

The proposed solution is to create a separate context interpreter, which manipulates and translates contextual information, in a manner transparent to the core adaptive engine. This model has the advantage of separating ‘deep’ and ‘shallow’ models, along with advantages of re-usability and encapsulation. This paper concentrates on the methods for applying contextual information in this form to an established adaptive eLearning system, in particular the points at which information can be applied. This model is a work in progress, and this paper is concerned with the initial results of its design and implementation.

### ARCHITECTURE

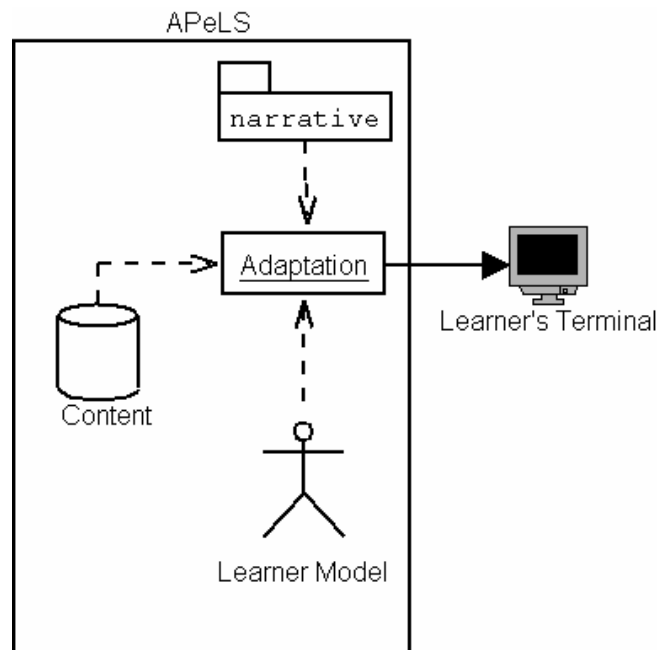
This section is concerned with a description of the APeLS adaptive eLearning system, and the method by which a Context Interpreter is connected to it.

### APeLS

The APeLS [7] system is a multi-model, metadata driven adaptive hypermedia system, comprised of three main models:

- The Learner Model contains information about the user, stored as vectors of adaptivity information. This information includes the learning goals, previous learning experience and learning vocabulary of the learner.
- The Content Model is organised into Candidate Groups composed of fine-grain ‘pagelets’ of information and associated with defined learning concepts. These are selected and composed at runtime by the Adaptive Engine based on information in the Learner Model, and subject to rules in the Narrative.
- The Narrative Model governs the rules by which selection and ordering of concepts is performed.

Schematically, the system can be viewed as:



**Figure 1 Schematic View of APeLS**

The result of the adaptation within the system is a web page with the appropriate course content, and links to the next and previous sections of the course, tailored towards the learner.

This system maintains the state of the course as part of the Learner Model, as each Learner will likely be learning different things at different times.

An important feature of this architecture is that multiple roles are envisioned in different portions of course creation. The narrative and concept domain are intended to be created by a pedagogical and domain experts, while a subject matter expert undertakes the creation and suggested organisation of pagelets. This permits the dynamic selection of narratives based on such factors as scope, learning style, objectives and curriculum.

Specifically, the deep models that this system is concerned with are the adaptivity features of the Learner, the content model as a collection of concepts to be learned, and the narrative as a learning process.

### Applying Context to APeLS

Each of the models described within the APeLS system has key importance in the process of selecting and organising information. There has been a suggestion [8] to expand the presentation aspect of the adaptation process to include terminal adaptation, where the colour, size and resolution capabilities of the Learner’s Terminal are incorporated into the selection and presentation process [9].

While it would be possible to include this information directly within the Content Model, and somehow measure the Learner’s Terminal status and include that within their adaptivity information, such a process could be considered

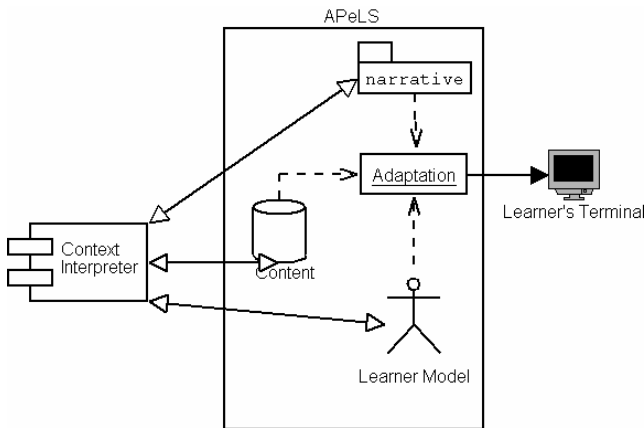
the beginning of a considerable expansion of the size of each model; the narrative would also need to be expanded, in order to account for correlating these axes.

Furthermore, such a selection does not form a key requirement of educational adaptive hypermedia. It is not a concern in the core functionality of the system, but rather a useful addition. More importantly, the process of deciding which content of a relatively small repository fits ‘best’ with any of a wide variety of systems creates the requirement for complex and expansive rules within the narrative, which will need to be altered as terminal types are added.

Instead, terminal adaptivity is categorised as being of ‘shallow’ interest, and the detail of selection is ‘handed off’ to the Context Interpreter.

The Context Interpreter is responsible for handling all shallow concerns in the use of an adaptive system. It can be considered as the ‘bridge’ between the adaptive engine’s models and the rest of the world (‘context’). Since any addition of concern to the system can be implemented in a number of ways, and can have an effect on any combination of core models, the Context Interpreter must be able to interface with the adaptive engine at any of these models.

Diagrammatically, the addition of the Context Interpreter can be viewed as:



**Figure 2 Influence of the Context Interpreter**

The Context interpreter is therefore free to make decisions at a variety of places within the system.

This method of applying context has a number of advantages:

- Changes can be made to the attributes and range of choice of contextual models without change to the core system.
- Technical concerns such as terminal adaptation can be left to technical experts, and place no additional burden on likely non-technical content authors such as the Pedagogue and Domain Expert.

- There is no requirement to specify what additional contextual information might be useful in the course. Instead, the Context Interpreter is given authority to make such decisions at runtime.

This separated model provides the full capabilities of the contextual infrastructure within a system while placing minimal requirements on the authors of the system to predefine such capabilities.

Of particular interest is the method by which the Context Interpreter and the adaptive engine reach semantic agreement on the terms they manipulate. Currently, this is modelled as a simple shared vocabulary. However, future work is intended to examine the use of ontologies as a method for finding such agreement.

**SCENARIOS FOR USE**

Three primary categories of scenario have been outlined, in order to guide the experimental testing of the Context Interpreter as part of the adaptive engine:

- The ‘high-context’ scenarios include those where the Context Interpreter provides information which considerably improves the quality of the adaptation. For example, in the case where a new Learner begins a course with the system, the Context Interpreter may provide details of previous information learned, or of preferred learning style, enriching the largely-blank Learner Model with contextual information. This permits the adaptive engine to alter what course information is presented and perhaps start at a different point in the course.
- ‘Mid-context’ scenarios are concerned with assisting the adaptive engine where a multiplicity of equal choices exists. For example, from an educational perspective the choice of C++ or Java for explaining Object Orientation may be neutral from the perspective of the adaptive engine, but the Context Interpreter may be aware of the Learner’s preference for one language or another. This can be viewed as a defined decision point within the process of adaptation where the adaptive engine could make a random choice and still produce a useful adapted course.
- ‘Low-context’ scenarios are related to fine tuning adaptive choices and technical decisions. These might include the Terminal Adaptation example discussed above, or the preference of one set of content over another, because the learner at home (and on a slower connection) than in college. These are choices which the adaptive engine is totally unaware of and uninterested in.

In all these cases, it can be seen that contextual information is the residual or supplemental portion of data, available optionally to the system. In the event of it not being available, there is no critical loss to the system.

**IMPLEMENTATION AND INITIAL EXPERIENCE**

Currently, the investigation of this system is at the stage of experimental testing. The mechanisms for applying contextual information are under examination in the scenarios as presented. Currently, the Context Interpreter itself has not been implemented as an advanced system. Once the implications of applying contextual information are examined, it is proposed that the Context Interpreter be expanded. Currently, the three mechanisms for applying contextual information are implemented as a web service over SOAP (Simple Object Access Protocol). Three mechanisms are provided, the first is to permit alteration of the Learner Model, the second is to create decision points in the Narrative, and the third is to provide for content group manipulation.

APeLS is composed of a set of dynamic JSPs (Java Server Pages) which present data from the Content Model as determined by the Narrative, which is formed by a JESS [10] (Java Expert Shell System) interpreter, to provide rule-based correlation of models.

Contextual mechanisms are implemented as user functions in JESS, which can be called at times chosen by the author of the Narrative. This maximises the control of the author, creating a transparent interface for decision points and system interaction.

Initial experience [11] with the system reinforced the need to accommodate *user empowerment* in adaptive systems. It is vital in order to maintain user commitment and concentration that feedback be available for decisions made by the system. In addition, feedback mechanisms provide a facility to poll the user directly for choices for which the Context Interpreter cannot find sufficient information.

Context-informed adaptive routines demonstrate promising advantages for supporting candidacy[12] in the AH system. The three contextual mechanisms, as implemented, permit a dialogue between the adaptive engine and exterior concerns, which allows the system to employ other resources transparently. The dialogue for the purposes of Candidate Group Selection can be provided either at Content or Concept level, depending on the architecture of the specific course. This permits selection and reuse to be made based on reasoning over a considerable number of factors, without significant growth in the complexity of the AH model set.

The other useful capacity suggested would be the maintenance of a contextual history. This might be used for a great many applications, not least to cache decisions, but also to provide a corpus of data for more advanced reasoning, such as semantic translation to deduce the relevance of topics in other courses.

#### **RELATED WORK**

The Context Awareness Subsystem (CAS) of the MOBIlearn project [13] has some similarities with this work. However, given that the system is concerned with *mobile* learning, location cannot be said to be a shallow

concern. The CAS gathers all model information as 'context', and does not provide a separate layer for context interpretation. Instead, it is a fully integrated system. The question over this integration is how robust it will be in the face of ever-increasingly numerous concerns.

#### **CONCLUSIONS**

This paper has presented the architecture of a separated Context Interpreter, as applied to a multi-model, Adaptive eLearning System.

The separation of systems, as presented, is the key principle of this method. This separation supports the consistency of the deep models, by permitting those concerns which are extraneous to be placed in an external location. It furthermore supports the design of models, and the performance of the adaptive engine, by relocating the expansion of functionality, and aggregating technical concerns away from non-technical authors.

The three mechanisms outlined reflect the fact that contextual concerns apply within models, and may perhaps apply to several models at once.

The architecture is 'double-blind', the Context Interpreter and adaptive engine are not permitted to attempt to 'second-guess' each other, instead they communicate only through the defined interfaces. Feedback and decision support are provided directly by the Learner.

Future work on this system will be concerned with the measurement of contextual attributes, and translating these attributes into alterations to the model.

It is thought that this model of a high level, remote contextual reasoning system has general application. The use of such a 'wrapper' system has clear advantages, providing component-like pluggability to almost any system. Further work may also include applying this architecture to other systems, for example by generalising the adaptive engine to become a web service aggregator and broker.

#### **REFERENCES**

1. Brusilovsky, P.: Methods and techniques of adaptive hypermedia. In P. Brusilovsky and J. Vassileva (eds.), Spec. Iss. on Adaptive Hypertext and Hypermedia, User Modeling and User-Adapted Interaction 6 (2-3), 87-129
2. De Bra, P., Aerts, A., Berden, B., De Lange, B., Rousseau, B., Santic, T., Smits, D., Stash, N., AHA! The Adaptive Hypermedia Architecture. Proceedings of the ACM Hypertext Conference, Nottingham, UK, August 2003, pp. 81-84
3. Brusilovsky, P., Eklund, J., and Schwarz, E. (1998) Web-based education for all: A tool for developing adaptive courseware. Computer Networks and ISDN Systems (Proceedings of Seventh International World Wide Web Conference, 14-18 April 1998, 30 (1-7), 291-300.

4. Brusilovsky, P. and Nijhawan, H. (2002) A Framework for Adaptive E-Learning Based on Distributed Re-usable Learning Activities. In: M. Driscoll and T. C. Reeves (eds.) Proceedings of World Conference on E-Learning, E-Learn 2002, Montreal, Canada, October 15-19, 2002, AACE, pp. 154-161.
5. Bill N. Schilit, Norman I. Adams, and Roy Want.(1994) Context-Aware Computing Applications. *Proceedings of the Workshop on Mobile Computing Systems and Applications*, Santa Cruz, CA, December 1994. Pages 85-90. IEEE Computer Society
6. W. G. Griswold, R. Boyer, S. W. Brown, and T. M. Truong, "A Component Architecture for an Extensible, Highly Integrated Context-Aware Computing Infrastructure", 2003 International Conference on Software Engineering (ICSE 2003), May 2003.
7. Conlan, O.; Wade, V.; Bruen, C.; Gargan, M. Multi-Model, Metadata Driven Approach to Adaptive Hypermedia Services for Personalized eLearning. Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Malaga, Spain, May 2002,
8. Dagger, D.; Wade, V.; Conlan, O. Towards "anytime, anywhere" Learning: The Role and Realization of Dynamic Terminal Personalization in Adaptive eLearning. Ed-Media 2003, World Conference on Educational Multimedia, Hypermedia and Telecommunications, Hawaii
9. Brady, A., Conlan, O., Wade, V. Dynamic Composition and Personalization of PDA-based eLearning Personalized mLearning. In proceedings of the World Conference on E-Learning in Corporate, Government,Healthcare and Higher Education (eLearn 2004; in Press).
10. JESS, the rule engine for the Java Platform <http://herzberg.ca.sandia.gov/jess/>
11. Conlan, O. and Wade, V. Evaluation of APeLS - An Adaptive eLearning Service based on the Multi-model, Metadata-driven Approach" In proceedings of the third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2004; in Press).
12. Dagger, D., Conlan, O., Wade, V.. An Architecture for Candidacy in Adaptive eLearning Systems to Facilitate the Reuse of Learning Resources. In proceedings of the World Conference on E-Learning in Corporate, Government, Healthcare and Higher Education (eLearn 2003) pp112-116.
13. Lonsdale, P., Baber, C., Sharples, M. (2003). A Context Awareness Architecture for Facilitating Mobile Learning. Proceedings of MLEARN 2003: Learning with Mobile Devices, London. Learning and Skills Development Agency, pp. 36-37.