Toward a Formalisation of Evolutionary Hypermedia Systems Based on System Theory[1]

Lina García-Cabrera[1], María José Rodríguez-Fórtiz[2], and José Parets-Llorca[2]

[1] Departamento de Informática, E. P. S., Universidad de Jaén
Avda. Madrid, 35. 23071 –JAEN (Spain)
lina@ujaen.es
[2] Departamento de L. S. I. ETS Ingeniería Informática. Universidad de Granada.
Avda. Andalucía, 38. 18071 –GRANADA (Spain)
jparets@ugr.es

Abstract. Main development and use activities of Hypermedia-Systems evolve through time. Hypermedia-systems need models that support the evolutionary nature of their building, maintenance and navigation processes. The System Theory and cognitive models offer a better perspective of web-systems and succeed in abstracting their structure, information and behaviour. We assume that an Evolutionary Hypermedia System must be modelled by means of a set of interrelated and interacting systems that allow: a) from the author's viewpoint, complete and flexible control of the development and maintenance of hyperdocuments; b) from the reader's point of view, an understandable navigation that allows easy access to and selection of information. The Model allows an explicit representation of the semantic content which allows us to structure the information-system and determines its possibilities of change, updating and evolution. In addition, the model is flexible enough in offering the necessary mechanisms to incorporate and represent the author's conceptual-domains in order to characterise the information-domains.

1 Introduction

Traditional hypermedia reference models [4], [3], [7], [2], [8] "tend to focus on abstracting the connectivity of hypermedia –links- from its underlying information -nodes- rather than abstracting structure from functionality" [12], i.e., these focus more on edition and document navigation through prefixed links than on the dynamic construction, evolution and maintenance of the document. The traditional skeleton of hypermedia models, based on a set of hierarchical levels that can be translated into a sequential and static methodology, is not the best approach in representing complex and evolving realities, where construction, maintenance and navigation are confused by their strong interrelationships. In these models there is no correspondence between structure and functionality. In our opinion, a functional systemic perspective is more

[1] This research is supported by a project –MEIGAS- by the Spanish CICYT (TIC2000-1673-C06-04) which is a subproject of the DOLMEN project (TIC2000-1673-C06).

R. Moreno-Díaz et al. (Eds.): EUROCAST 2001, LNCS 2178, pp. 270-284, 2001.

suitable and hypermedia systems must be conceived under an evolving model with the following assumptions:

1. Hypermedia systems need a functional systemic perspective [9], [11], that is, a hypermedia system can be conceived as a set of interacting systems in continuous evolution.
2. The model must help and make flexible the construction, maintenance and navigation of the hypermedia systems. These three key aspects are exposed to continuous changes and updates the model should be able to integrate.
3. An explicit semantic representation must permeate the model. The possibilities of which structuring and further changes, adaptations or evolution will depend on the level of explicitness of this semantic representation. The building process of a hypermedia system must be based on a cognitive model [5].
4. A cognitive model benefits the users –author and reader- during development and use activities: construction, maintenance and navigation.
   - The author can make an incremental design process of his/her hyperdocuments.
   - Collaboration between authors is possible.
   - Effective maintenance is possible when the process of underlying reasoning and decision-making carried out or Design Rationale [13] is represented.
   - The reader can have a contextual access that facilitates his/her knowledge and comprehension.
5. The model must offer a flexible semantic representation that allows the author a characterisation of his/her own information domains by means of his/her own ontologies.

Guided by these objectives, we present here a Semantic-Evolutionary Model (SEM-HP). Section 2 will present a general architecture of the proposed Model and an example, which will be used in further explanations. After that, in section 3, the important concepts of the model are defined and explained. Section four provides an extensive description of the Knowledge and Navigation Systems. Section 5 goes back to the architecture in order to explain a more detailed view of the functionality and evolution of the Systems. Finally, section 6 summarises the conclusions about the presented model, current research and further developments.

2 Architecture of Evolutionary Hypermedia Systems

Stemming from the previous assumptions a Semantic-Evolutionary Model is proposed in order to support HyPermedia Sytems, SEM-HP, with the following three systems – figure 1-: The Knowledge System, the Navigation System and the Learning System.

The *Knowledge System* is in charge of the storage, structuring and maintenance of the different pieces of information. It memorises the knowledge acquired about the information system that is represented. This knowledge will guide the design and structuring processes of the information system. It will determine the possibilities of transformation and change of this structure throughout its evolution. The *Navigation System* helps the reader in his/her interaction with the information system. Using the knowledge base and the reader activity through time dynamically, this system determines –firstly– the accessible information and –secondly– its interaction possibilities.

Finally, The *Learning System* optimises the knowledge acquisition process from the hypermedia system adapting navigation to the information needs and to the knowledge gained by the reader.

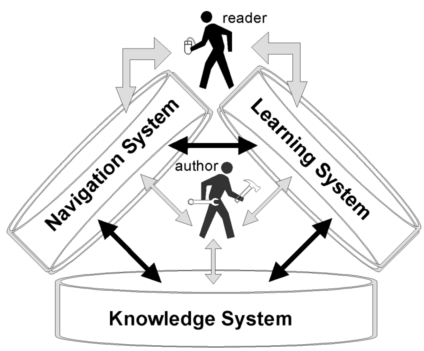

Fig. 1. Semantic-Evolutionary Model based on Systems. Different systems interact among themselves -black arrows-. The Reader interacts with the Navigation and Learning Systems, while the author interacts with the three systems -grey arrows-.

Each System redefines, to some extent, the knowledge base provided by the Knowledge System, which is stable for the reader but dynamic for the author or authoring tool. Each System is supported by itself and contributes additional information. This information will determine what pieces of information can be consulted and under what prism. The different systems interact among themselves and their interaction produces, in a dynamic way, adaptations within them. In order to clarify the explanation and show the possibilities of the approach we use a concrete conceptual and information domain example about the *Solar System* –figure 2-.

3 The Conceptual Perspective of the SEM-HP

In order to highlight the evolving aspects of the model we will start by explaining the meaning of the four most important concepts of our model: *Information Items (II)*, *Conceptual Structure (CS)*, *Restrictions (RT)* and *Evolving Actions* (AC_e). In their explanations other basic concepts will appear, and the interactions between them will be shown. We will also see how Conceptual Structure and Restrictions stress the cognitive and evolving aspects of the hypermedia system.

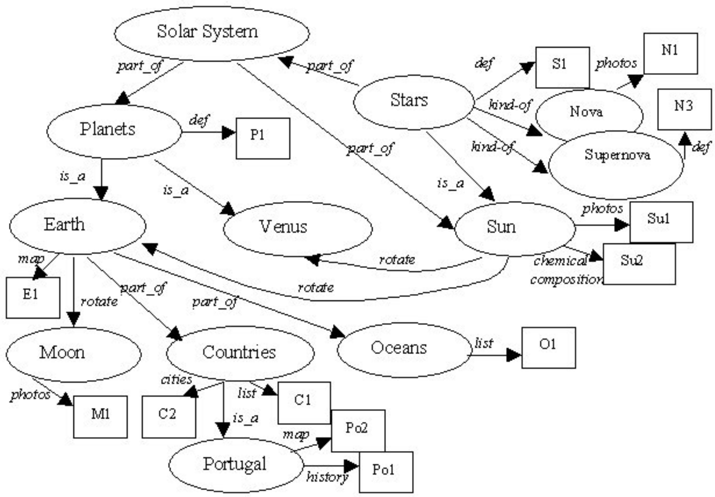

Fig. 2. Conceptual Structure CS about the Solar System

3.1 Information Items

A hypermedia system is an information system made up of different pieces of information –the *information items*[2]– that can be referred to, used and composed. These pieces can be expressed in any language and can be provided by the authoring system, a computer application or a database[3]. The author can associate *properties* to these items. These properties include the *type* of information and the *function* of the item in a context.

An information item is referenced by one or more *concepts* or labelled ideas. These concepts will be part of a *Conceptual Domain* -our example has 12 concepts represented by ellipses; i.e., *Planets*, *Stars*, etc- created by the author during the development and maintenance of the hyperdocument. The set of information items identified by the concepts in a Conceptual Domain will be called *Information Domain* -the example contains 13 items represented by a square; i.e., *Su1*, *C2*, etc-.

Many information items can refer to the same concept or set of concepts. In this case each information item will play a different function in a context. We call this function or intention of the information item its *Role*. The Role is useful for the author because it provides the function of the item, and for the reader because it guides the

---

[2] Here the term "item" is preferred to "chunk", more widely used in the literature, for two reasons: a) historical fidelity because this was the word used by Vannevar Bush in his memorable and forerunner paper "As We May Think" [1]; and b) we consider that the term "information item" represents better the idea of an own-entity piece of information that is implied by a conceptual unit.

[3] More than one semantic model in research literature simply builds documents as compositions of data represented and presented as a style template, that is to say, in the form of a database. Unfortunately, many documents cannot be adapted to this simple skeleton.

intention of a link. For instance, the *Sun* concept -one of the *Stars*- is referenced by different items, which play the roles of *photos* or *chemical-composition*. Apart from concepts and roles an information item has additional properties: *type of language of the content* -text, sound, graphics, image, animation, execution or hypermedia-, *edition aspects* -authors, date, revision date, quality, visitors,...- and *Specialisation Level* or version. This property is related to the user's degree of knowledge and allows the possibility of items with different depth levels, which can be selected by individual readers. For instance, some properties of the *Po2* item are *Portugal*, *map*, *image*, *PlanetEducation*, *26feb00*, *teenagers*.

3.2 Conceptual Structure

The set of concepts of a Conceptual Domain constitutes a directed graph, in which nodes and links are labelled with semantic meanings –a semantic net [14]-. The graph represents the Conceptual Domain –concepts and associations[4] between concepts- of the information system, named *Conceptual Structure* (CS*)*. The different information items –documents- can be associated –labelled- with one or more concepts of the CS – i.e., <*Stars*, def, S1>, see figure 2-. These items are also nodes of the CS. In order to allow provisional and incomplete development, items which are not related to any concept can also be included. Therefore CS is defined as: CS = (C, II, A_c, A_i), where C is the set of concepts, II is the set of information items, A_c is the set of labelled conceptual associations and, A_i is the set of labelled associations between concepts and information items.

We distinguish between *Reference* and *Dependency* Conceptual Associations. *Reference Conceptual Associations –i.e.* <*Earth*, rotate, *Moon*>- are domain dependent and must be defined by the author for each particular conceptual domain, i.e. the author provide his own ontologies [17]. These ontologies -concepts and associations between concepts- define a dictionary of keywords which is used by the author in order to provide the structure, and by the reader in order to select material.

In addition, *Dependency Conceptual Associations*, which are domain independent and have a generic character, can be considered: *aggregation (partOf), instantiation (isA)* and s*pecialisation (aKindOf)*. The dependency *partOf* allows hierarchies between concepts –i.e. <*Solar-System*, part-of, *Planets*>-. The dependency *aKindOf* allows the composition of information items –i.e. <*Stars*, kind-of, *Nova*>-. For instance, *Nova* and *Supernova* have an association *aKindOf* with *Stars*. Then, a composed item, which is labelled with the generic concept –*Stars*-, can be constructed by grouping all items associated with the children concepts. The dependency *isA* allows the definition of a concept using more generic concepts -i.e. <*Stars*, isA, *Sun*>-.

Conceptual associations allow the definition of the *Concept Environment*, i.e. the set of concepts which are related to another concept. In the example, the environment

[4] We prefer the term “association” instead of “link” because links have a clear static meaning in current models and links are more diffuse in the research literature. The term *association* reflects the fact that this connection between information items responds to relationships between the concepts represented by them more than to circumstantial reasons -as usually occurs in links-.

of *Stars* is made up by concepts such as *Solar-System*, *Nova* or *Sun*. The notion of environment allows some interesting operations which are known in the literature as *queries based on the structure*:

- Which concepts add more information to another concept.
- Which concepts are derived from another concept.
- Which concepts produce or cause another concept.
- Which concepts are one level higher or lower in the conceptual structure.
- Which concepts are separated from another concept by a distance d.
- Which documents are related to some conceptual domain.

The previous conceptual associations allow the dynamic creation and evolution of computed documents, i.e. the authors can construct new documents by means of this explicit semantic structure. *Restrictions* about conceptual associations also guide the authors during the construction and maintenance of the Conceptual Structure because they can forbid some structures and associations -see below- in a concrete information domain.

3.3 Restrictions

Restrictions (RT) guide the development, maintenance and navigation of hypermedia systems. They are supplied by different Systems, and are always applied -as we will see later - by these systems. They limit *associations* between concepts in the CS and constrain associations of information items that can be used during navigation. Dynamically, way a set of restrictions will hold for each information item and they will limit the set of associated items. We will call this set the *Item Framework*. Two types of restrictions can be distinguished:

1. Derived from the semantic structure of the information system. Obviously, navigation will be restricted inside the *world* conceived and designed by the author. These restrictions will be applied by the *Knowledge System (KS)* and can be basic, defined as a functional part of the *KS,* or can also be defined by the author. Some examples of basic restrictions are:
   - Each association of the *CS* must connect two concepts or a concept and an item.
   - Each arc and node of the *CS* must be labelled.
   - Two nodes in a *CS* cannot have the same label.

   The author can also include additional restrictions which determine what associations between concepts are possible. In order to represent these restrictions, formulas in temporal logic are used. This formalism also allows checking if the CS is valid at any moment. Some examples are:
   - The concept *Stars* can be connected to the concept *Planets* by means of the association *rotate*.
   - The association *rotate* must be acyclic.
   - A concept X can be connected with concept *Countries* if the concept *Countries* is previously reached from the concept *Earth*.
2. Derived from the navigation itself and providing different navigation 'styles' which can be performed using the same semantic structure:

- the type of navigation: a group of restrictions that constrain more or less the navigation paths of a Conceptual Structure;
- the navigation carried out by user through time or functional history. The functional history is the set of operations performed by the user during a work session, i.e. the information items selected by the reader and their order;
- considerations about security and access control: user identification, restrictions in accessing the Conceptual Structure, item roles and item versions.

The possibility of adding restrictions implies adaptations and changes in the hypermedia system. These restrictions are described formally using graph theory and in a temporal logic language –a more detailed use of these formalisms can see in [6]- which supports expressions as: “if this and that then...”, “if before ... and after ... then show...”, “take into account whether the reader knows these or those concepts ", "if the reader has made this tour.... then these and those items can be shown”. Like Stotts and Furuta [16] we consider that a hypertext is an *interactive document* which provides a *dynamic structure*. This assumption implies the need for temporal logic in expressing what link sequences can be followed during browsing. These authors propose temporal logic as formalism in checking the properties of a hypertext. In our approach we also use temporal logic as an inherent way of expressing restrictions. Consequently, this kind of rules determines, at all times, which pieces of information can be reached and which are the information items that can be searched. These rules are provided by the hypermedia author and are indirectly selected by the reader when he/she specifies a navigation type or navigates through the system. In the example, the items labelled with the concepts *Nova* or *Supernova* and the items subordinated to these concepts should be hidden to a user who does not know the *definition* of *Stars* concept.

3.4 Evolving Actions

All systems include a set of evolving actions (AC_e) that allow changes to be made and propagated in the hypermedia system. An evolving action can belong to three different types:

1. Actions that redefine some aspects the system. Obviously the basic restrictions, defined by the system, discussed below, $RT_{s,}$ cannot be changed.
2. Actions that control the propagation of these changes inside the system itself.
3. Actions that control the propagation of these changes outside the system, i.e. in the other systems of the SEM-HP.

When these actions are carried out they change the corresponding elements of the hypermedia system. Because integrity should be guaranteed in any case, these operations should be carried out following a set of meta-restrictions. The specification of these meta-restrictions implies a meta-level in the definition of the Systems.

4 The Systems of the SEM-HP

Each of the systems of the SEM-HP can be defined by: a) one or more artefacts which represent its particular vision of the conceptual and information domain, b) a set of restrictions RT which control the construction and guarantee the consistency of these artefacts and, c) a set of evolving actions AC_e that allow changes to be made and propagated in the hypermedia system. In next subsections we will define and describe the different systems and their components. For better understanding, the previous definitions are included in the appendix.

4.1 Knowledge System

The main objective of the *Knowledge System* is the storage, structuring and maintenance of the different pieces of information. It is made up by a *Memorisation Subsystem* and a *Presentation Subsystem*.

The *Memorisation Subsystem* allows the storage of selected knowledge for each Domain Information –pages or documents-. It memorises information concerning the whole *Conceptual Domain* –concepts and conceptual associations- (definition 4), which is managed in a particular information system. The elements to be managed are:

1. The *Conceptual Structure* (definition 8) which allows *information item*s (definition 1) to be catalogued. The CS is formalised by a directed graph, CS = (C, II, A_c, A_i), where C is the set of concepts, II is the set of information items, A_c is the set of labelled conceptual associations, A_i is the set of labelled associations between concepts and information items.
2. The *Information Items:* the different pieces of information that can be used to construct hyperdocuments. These information items will be expressed in one or more possible language/s -such as text, sound, graphic, image, animation, execution or hypermedia- and will have to be catalogued under one or several concepts of the domain. They will also be labelled with one or several roles into a particular context. They will have certain edition properties.

Because CS is constructed by the authors, dynamically, some evolution actions AC_{em} such as add-concept, delete-association, modify-association, add-item, etc. have to be included. The actions must verify a set of restrictions RT in order to maintain the consistency of the CS. These restrictions can be basic ones RT_s, defined as a functional part of the *MS,* or can also be defined by the author RT_a – as described in the 3.2 section-. Therefore, the *Memorisation Subsystem* is defined as *MS* = (CS, RT, AC_{em}), where CS is the previously defined, directed and labelled weakly connected graph that represents the conceptual domain of a hypermedia system, RT is the set of restrictions and AC_{em} is a set of evolving actions -see next section-.

The *Presentation Subsystem* determines the set of possible views of a specific Conceptual and Information Domain. To some extent it establishes the possible views of the hypermedia documents which can be built with the items of the Memorisation Subsystem. The Presentation Subsystem, using as basis the CS of the Memorisation System, allows a selection of a subset of the concepts and associations included in CS. This graph, CS_p, a subgraph of CS, $CS_p = (C_p, II_p, A_{cp}, A_{ip})$, will be presented to the reader. The Conceptual substructrure chosen by the author must respect, absolutely, all the restrictions (RT) set in the Memorisation System. Each time, the author change the substructure, the system must check that the new conceptual and information Domain selected verifies the restrictions. For instance, figure 3 shows the subgraph CS_p chosen by the author, taking into account the *Earth* out associations.

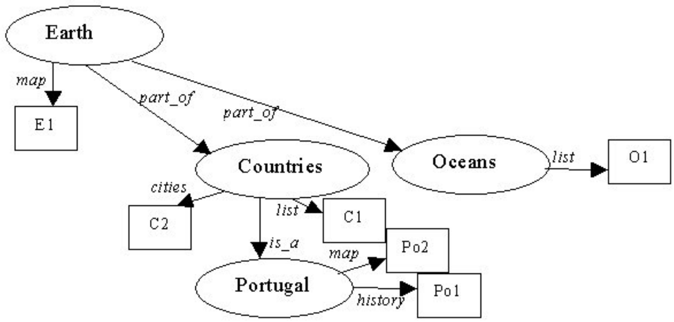

Fig. 3. CS_p selection of CS.

Therefore, the *Presentation Subsystem* is defined as *PS* = (CS_p, RT, AC_{ep}), where CS_p is a subset of the original CS, RT is the set of the same restrictions of the Memorisation Subsystem and AC_{ep} is a set of evolving actions that allows the author to limit or reduce the CS.

As a result, the Knowledge System stores the pieces of knowledge of the conceptual worlds that the author will use in his/her documents. This System permits the specification author restrictions RT_a. Using these restrictions the system can help the author in creating and maintaining –guaranteeing its consistency- their conceptual and information domains.

4.2 Navigation System

The *Navigation System* permits browsing and remembering the memorised knowledge, adapting it to the characteristics and interaction of the reader. The Navigation System permits the ordering of, in some form, the Conceptual Structure and the Information Domain associated to it, both offered by the Presentation System.

We can consider navigation as the execution of a particular presentation. The Navigation System has to take into account the following information at all times:

- First, the information item where the document reader is located at any moment,
- Second, the *conceptual environment* of the information item,
- Third, item *information framework*, i.e. the restrictions set that is true for an information item.

Therefore, the Navigation System, using as basis the CS_p of the Presentation Subsystem, can add more restrictions in order to follow more restricted paths in the subgraph. These restrictions or navigation rules RT_n are expressed formally using temporal logic. Considering the CS_p and temporal restrictions, a Petri net can be automatically constructed. As demonstrated in [10] and in [15], Petri nets give an operational semantics to temporal logic formulas allowing operational navigation. The algorithm which constructs a Petri net from temporal logic formulas is explained in [10]. Summing up, the Navigation System is defined as $NS = (CS_p, RT_n, PN, AC_{en})$, where RT_n is the set of restrictions specified by the author by means of temporal logic, PN is the Petri Net and AC_{en} is the set of evolving actions for adding, deleting or modifying navigation restrictions.

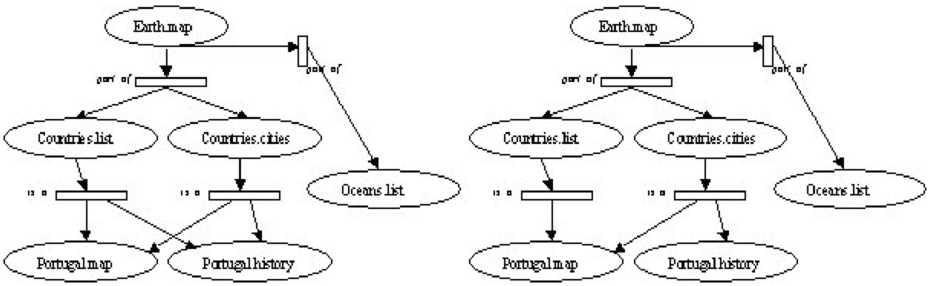

Fig. 4. a) A Petri net from the CS_p, b) Petri net from CS_p and one navigation restriction RT_n

The Navigation System models evolution using predicate temporal logic. It provides a meta-level with evolution actions which manage and change the navigation restrictions. Navigation restrictions can be added, deleted or modified, and the meta-restrictions of these operations can be established. In a similar way to the Knowledge System, the consistency must be guaranteed during the evolution of the Navigation System. In this system, changes can be produced in the navigation restrictions, RT_n, defined by the author, and therefore, in the PN obtained from them. For instance, from the conceptual substructure CS_p, the Navigation System can produce the Petri net of the figure 4a. The author can add more restrictions: “the reader can only reach the *Portugal.map* item if the has visited the *Countries.cities* item”. Then the Navigation System must generate a new Petri net –figure 4b-. Of course, all limitations are not possible, for instance, all items selected in CS_p have to be reached and all conceptual associations have to be fired –the system must verify this meta-restriction-.

4.3 Learning System

The last element of our model is the *Learning System*, which modifies navigation by taking into account the type of information that the reader wants to achieve -the *goals* of the reader- and/or knowledge that he wants to acquire or learn *–achievements-*. Now, we are beginning to work in the development of this System as we wanted first to have a prototype of the model.

5 The SEM-HP Systems Functionality and Evolution

Up to now we have described the pieces of information that our model can use in building a document, the properties that characterise them, their content, the conceptual structure and the restrictions that control the structuring and navigation process. In addition, the set of *Evolving Actions* and their preconditions, the *Restrictions*, provide conscious support for every one of previous components. Different formalisms –the most suitable for each system- are used in order to specify the evolving actions and their meta-restrictions.

The author, an expert in a domain, represents his complex domain/s of knowledge using the Memorisation Subsystem. He/she creates his concepts, C, and associations between concepts, A_c. This knowledge will be used in characterising the different information items II; the author associated these items with concept/s A_i.

As a result, he/she builds the Conceptual Structure CS. In addition, the author defines its restrictions in order to guide the constructions of the CS. During this process, the *Memorisation Subsystem* must always guarantee its consistency. Two aspects of this system can change, the CS –the graph- and the restrictions defined by the author. Graph Theory is used to represent the evolving actions of the graph and their associated meta-restrictions. Changes in restrictions defined by the author, $RT_{a,}$ must be defined by means of meta-restrictions.

When the author changes the CS –add, delete or modify a concept, item or association- the system must check:

1. The CS verifies the restrictions defined by the system and the associations satisfy the set of restrictions defined by the author. The RT acts as a set of restrictions for the actions, only if the action matches these restrictions, will it be carried out -internal propagation of changes-.
2. The subgraph used by the *Presentation Subsystem*, $CS_{p,}$ is consistent with changes in the CS. If a concept or association has been deleted in the CS, the *PS* must also delete this concept or association in the CS_p -external propagation of changes-.

When the author redefines –add, delete or modify- one associative restriction RT_a, the system must check:

1. The set of axioms about associations is valid, by means of predicate temporal logic.
2. The CS verifies the new set of restrictions, using graph theory. The system must detect the associations that do not satisfy one or more restrictions and delete them -internal propagation of changes-.
3. The CS_p –the subgraph selected by the *PS*- verifies the new set of restrictions by means of graph theory. The system must detect the associations that do not satisfy these restrictions and delete them -external propagation of changes-.

In addition, the author can select a particular subgraph CS_p from one Conceptual Structure CS using the Presentation Subsystem. In a similar way to the Memorisation Subsystem, the consistency must be guaranteed during the evolution of the Presentation Subsystem. In this system, changes can be produced in the subgraph selected CS_p. When the CS_p is changed –the author select another set of concepts and associations- the subsystem must check:

1. The CS_p verifies the restrictions defined by the system and the associations satisfy the set of restrictions defined by the author.
2. A new view or presentation is defined. In this case, the author must define again the navigation restrictions. This change is not a real evolution, the author is designing a new view of the information and, therefore, new navigation possibilities, but if these possibilities are defined in an incremental way, the system can aid the author in the design process -external propagation of changes-.

Finally, the author defines their navigation restrictions RT_n and the Navigation System must guarantee the consistency again. When the author redefines –add, delete or modify- a navigation restriction, RT_n, the system must check:

1. The set of restrictions that establish the order of navigation is consistent. Predicate temporal logic is used to specify the evolution operations over the restrictions, and their associated meta-restrictions.
2. The navigation restrictions have changed. Changes in a restriction can imply the modification of other restrictions. The PN based on the navigation restrictions must evolve, generating it again -internal propagation of changes-.

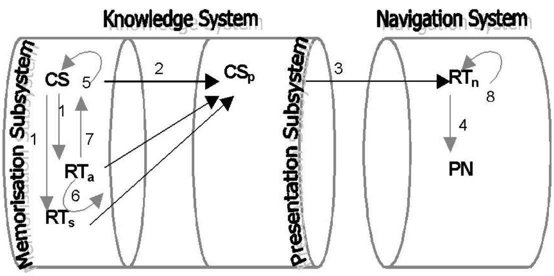

Fig. 5. Evolution: the evolving actions and their propagation.

To sum up, restrictions defined by the system, RT_s, or by the author, RT_a, are associated to the conceptual structure CS (1). Evolution can be carried out in the conceptual structure, CS (5), in RT_a by means of predicate logic (6) and in RT_n using predicate temporal logic (8). When RT_a is modified CS could also change (7). PN evolves being reconstructed from RT_n (4). The evolution in the *Memorisation Subsystem* is also propagated to the *Presentation Subsystem* (2) and, later, to the *Navigation system* (3).

6 Conclusions and Further Work

Traditional hypermedia reference models shows that they are not able to represent the development, maintenance and navigation processes of an information system in continuous evolution. We have proposed a SEM-HP model composed of some interrelated and interacting systems: *Knowledge –made of Memorisation and Presentation Subsystems-*, *Navigation* and *Learning*, where an explicit representation of the provided knowledge is carried out.

Each System redefines or restricts the knowledge base -which is stable for the reader but dynamic for the author or authoring tool- provided by the *Memorisation Subsystem* by means of a set of *Restrictions*. In addition, the SEM-HP model supports different formalisms –graph theory and temporal logic-, which allow the specification of the evolving actions and the propagation of the changes in order to maintain the integrity of the systems. The Learning System is optional but, if present, it offers an optimisation of the knowledge acquisition process, which is very useful in educational systems. The explicit representation of the semantic structure drives the development, maintenance and navigation processes of information systems. Consequently each system, basing on semantics, evolves –restructures the knowledge base- and makes the rest of the systems evolve.

Using the SEM-HP model and its specification formal, we are working in the construction of a prototype in Java and XML. In the near future we will improve the model specifying and formalising the Learning System.

References

1. Bush, V.: As We May Think, The Atlantic Monthly, July (1945) 176, 101-108. [On-line] http://www.isg.sfu.ca/~duchier/misc/vbush/.
2. Campbell, B., Goodman, J.M.: HAM: A general purpose Hypertext Abstract Machine, CACM, Vol. 31, n.7, July (1988) 856-861.
3. De Bra, P., Houben, G.J., Kornatzky, Y.: An Extensible Data Model for Hyperdocuments, 4th ACM Conference on Hypertext, Milan, December (1992) 222-231.
4. Furuta R., Stotts, P.D.: The Trellis Hypertext Reference Model, NIST Hypertext Standardization Workshop, February (1990) 83-93.
5. García-Cabrera, L., Parets-Llorca, J.: A Cognitive Model for Adaptive Hypermedia Systems. The 1st International Conference on Web Information Systems Engineering, Workshop on WWW Semantics. IEEE Press. Hong-Kong, China, June (2000) 29-33.
6. García-Cabrera, L., Rodríguez-Fórtiz, M. José, Parets-Llorca, J.: Formal Foundations for the Evolution of Hypermedia Systems. 5th European Conference on Software Maintenance and Reengineering, Workshop on FFSE. IEEE Press. Lisbon, Portugal, March (2001) 5-12.
7. Halasz, F., Schwartz, M.: The Dexter Hypertext Reference Model, NIST Hypertext Standardization Workshop, February (1990) 95-133.
8. Lange, D.B.: A Formal Model of Hypertext, NIST Hypertext Standardization Workshop, February (1990) 145-166.
9. Le Moigne, J-L.; La théorie du système général. Théorie de la modélisation. Paris. Presse Universitaires de France (1990).
10. Lin, C.; Chaudhury, A.; Whinston, A. B.; Marinescu, D. C.: Logical Inference of Horn Clauses in Petri Net Models. IEEE Transactions on Knowledge and Data Engineering, vol 5,3. (1993) 416-425.
11. Parets, J.; Torres, J.C. 1996. Software Maintenance versus Software Evolution: an Approach to Software Systems Evolution. IEEE Symposium and Workshop on Engineering of Computer Based-Systems (ECB'96). Friedrichshafen, Germany, March (1996) 134-141.
12. Schnase, J. L., Leggett, J.J, Hicks, D.L., Szabo, R. L.: Semantic Data Modelling of Hypermedia Associations. ACM Transactions on Information Systems, Vol. 11, n.1, January (1993) 27-50.
13. Shipman III, F.M., McCall, R. J.: Integrating Different Perspectives on Design Rationale: Supporting the Emergence of Design Rationale form Design Communication. Artificial Intelligence in Engineering Design, Analysis, and Manufacturing (AIEDAM), Vol. 11, n.2, April (1997) 141-154.
14. Sowa, J. F., De: Principles of Semantic Networks: Explorations in the Representation of Knowledge. The Morgan Kaufmann Series in Representation and Reasoning. San Mateo, California, Morgan Kaufmann (1991).
15. Rodriguez-Fortiz, M.J, Parets Llorca, J.: Using Predicate Temporal Logic and Coloured Petri Nets to specifying integrity restrictions in the structural evolution of temporal active systems. ISPSE 2000. International Symposium on Principles of Software Evolution. Kanazawa, Japan, November (2000) 81-85.

16. Stotts, P.D., Furuta, R., Ruiz-Cabarrus, C.: Hyperdocuments as Automata: Verification of Trace-Based Browsing Properties by Model Checking. ACM TOIS, Vol. 16, n. 1, January (1998) 1-30.
17. Uschold, M.: Ontologies: Principles, Methods and Applications. Knowledge Engineering Review, vol. 11, n. 2, June (1996).

Appendix: Glossary

1. An *Information Item* is any piece of identified information, which represents a conceptual unit in the information system. Each information item has a set of properties describing the type and functionality of the information it contains.
2. A *Property* of an information item is an associated attribute which describes the type, function and behaviour of the information that the information item contains.
3. A *Concept* is an idea, thought or abstraction which can be labelled by the author in order to make explicit his knowledge and understanding, i.e. a concept is a labelled idea.
4. A *Conceptual Domain* is the set of concepts to which the different information items in a hypermedia system may refer.
5. An *Information Domain* is the set of information items identified by concepts belonging to a certain Conceptual Domain.
6. An information item may play different *Role*s in the context of an information system. From the author's point of view, an item may play a certain role in the context, but for the reader, it follows a link with the aim of reaching a certain type of information about a specific concept.
7. The *Specialisation Level* or version is a property of an information item that determines the level of specialisation of the information contained in the item.
8. A *Conceptual Structure CS* of a Conceptual Domain is a graph of labelled concepts which maintains information about a) associations between concepts, and b) associations between concepts and information items.
9. A *Reference Conceptual Association* is a labelled association between two concepts, members of a Conceptual Domain.
10. A D*ependency Conceptual Association* is a labelled association that is independent of the considered Conceptual Domain {partOf, kindOf and isA}.
11. A *Concept Environment* is the set of concepts that are related to a specific concept.
12. *Restrictions* are the set of conditions or rules that constrain the conceptual and information associations. They guarantee the consistency of the different artefacts of the Systems and carry out the function of preconditions to evolving actions.
13. An *information item framework* is the set of restrictions that holds when an item is achieved. It limits or constrains the set of information items that can be further associated with it.
14. An *evolving action* is the set of operations that can change the artefacts and restrictions of the different Systems. An evolving action must verify a set of restrictions and, in this way, guarantee the consistency of the System and carry out propagation of the change inside and outside the System.

15. The *Memorisation Subsystem* is a subsystem of the Knowledge System. It establishes the raw material used in building the hypermedia system. It includes two main components: information items and conceptual structure. Other basics components of this subsystem are the dictionaries of concepts, reference associations between concepts, and roles.
16. The *Presentation Subsystem* is one of the subsystems of the Knowledge System of a hypermedia system. It allows the selection of subsets of a Conceptual Structure in order to determine the hypermedia document which will be shown by means of the Navigation System.
17. The *Knowledge System* is one of the Systems that models a hypermedia system. It provides information items, their categorisation and the basic rules to establish their possible associations. It is made up by the Memorisation and Presentation subsystems.
18. The *Navigation System* is one of the Systems that models a hypermedia system. It constrains or filters the set of possible presentations with the aim of choosing a subset of them. It uses the restrictions provided by the Presentation Subsystem and the selfsame Navigation System.
19. The *Learning System* is one of the Systems that models a hypermedia system. It allows the evaluation and modification of the Navigation System, taking into account the goals and achievements proposed by the reader at each moment.
20. A *Goal* is a set of information items that the reader wants to achieve.
21. An *Achievement* is a set of pieces of knowledge that the reader wants to acquire or learn. They can be defined using the Conceptual Structure.
22. A *Hypermedia System* under a SEM-HP model based on systems is a set of interrelated and interacting systems called a Knowledge System -composed by the Memorisation and Presentation Subsystems-, a Navigation System and a Learning System, which allows: a) easy and flexible development and maintenance of hypermedia documents, b) representation of the conceptual structure and dependencies between them, c) more than one representation of the information system –of a set of possible representations - and, d) dynamic navigation where multitarget, multiproposal navigation with structural contextualisation is possible.
23. A *Hypermedia Document* or Hyperdocument built with a SEM-HP model is a subset of information items and possible associations between them determined by the Navigation System according to a set of restrictions which it verifies at each moment.