User Modeling and Personalization 1: Adaptive Hypermedia

Eelco Herder

L3S Research Center / Leibniz University of Hanover
Hannover, Germany

13 April 2015

Outline I

- Introduction
  - Does one size fit all?
  - Hypertext and Hypermedia
  - Foundations of adaptive hypermedia
  - Hyperlinks
  - World Wide Web
- Adaptive Hypertext
  - Definition: Adaptive Hypermedia Systems
  - The adaptation process
  - When is Adaptive Hypermedia useful?
  - Application Areas
- Adaptation techniques
  - Content-level adaptation

Outline II

- Navigation-level adaptation
- Adaptive versus Adaptable Systems

- Logic-based adaptive (educational) hypermedia systems
  - First order logic in a nutshell
  - Simple
  - Simple - Extension 1
  - Simple - Extension 2
  - Constants and Predicates Used
  - Summary

Introduction: Adaptive Hypermedia

Adaptive systems are systems that *adapt* themselves to the current, individual user.

For example:

- the system displays only a selected number of documents that are deemed relevant to the user
- the system displays only selected, relevant parts of a document
- the system provides additional links to (possibly external) content
- only a selected number of relevant links (edges) between documents (nodes) is displayed
- the system recommends certain items or products

Question: in which situations would such adaptations be useful?

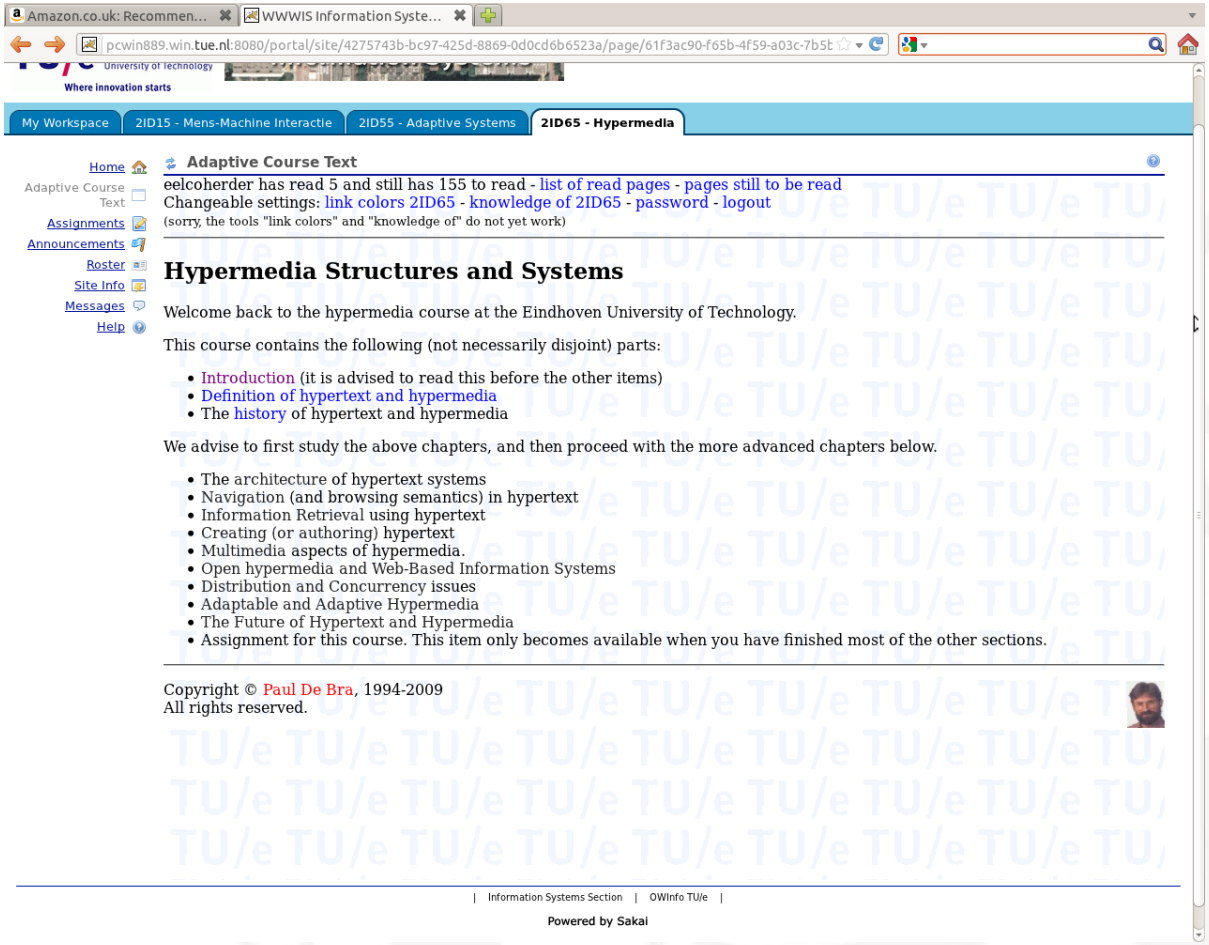


Figure: An adaptive course. Links to 'advanced' topics are disabled (made invisible) - the learner first needs to follow the introduction.

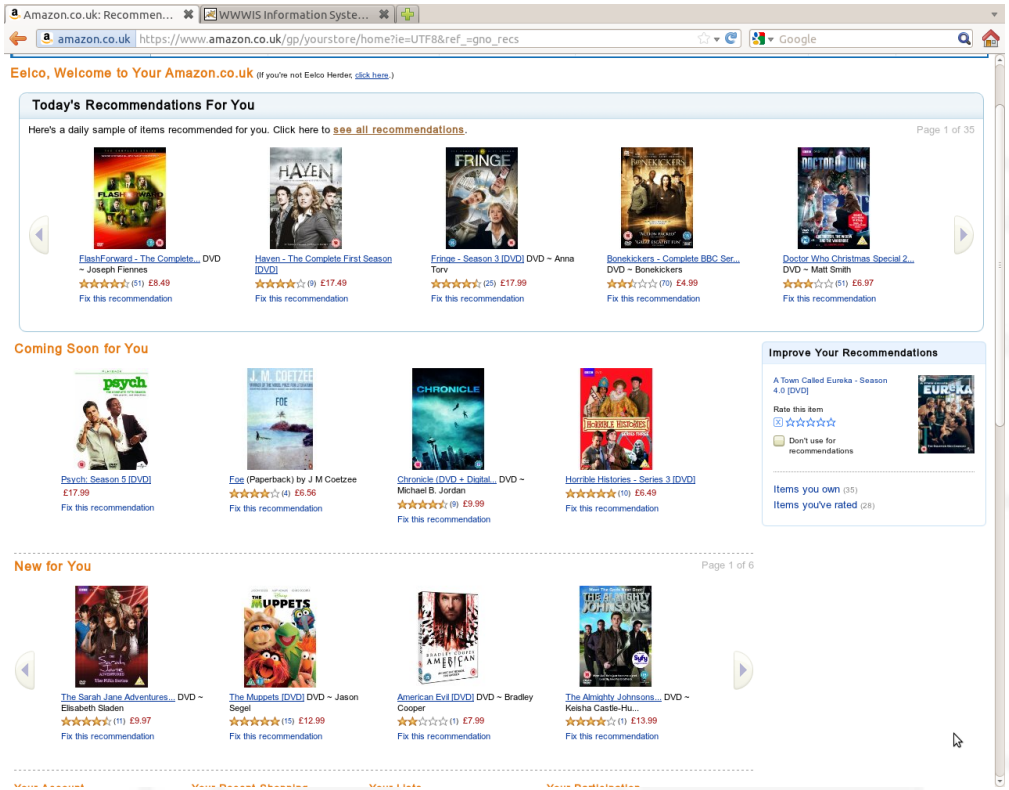


Figure: Product recommendations in Amazon. These recommendations are based on past purchases and past browsing behavior. The user can improve the recommendations by editing his or her user profile.

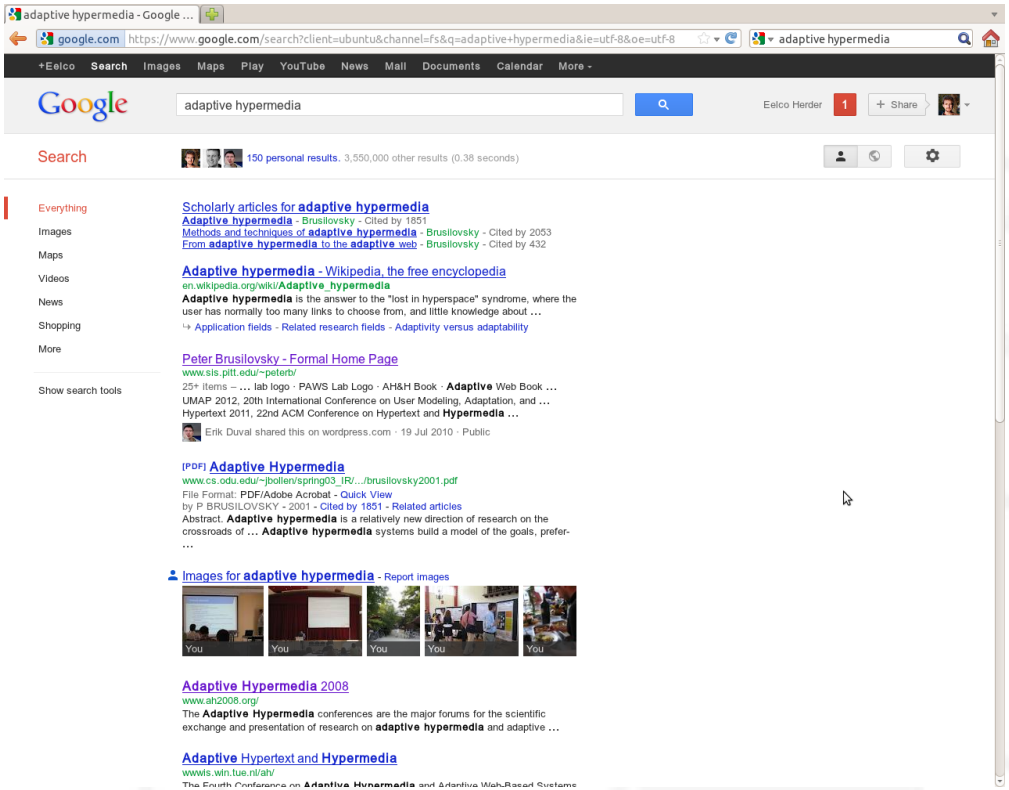


Figure: Google search results are personalized. Not only the 'personalized results', but the other results as well. Based on past searches, current location, language settings, etcetera.

Does one size fit all?

In a library, a person looks for some books on China. What will the librarian recommend?

- Is the person a *small child* who saw a TV show about China and wants to learn about this exotic country?
- Or a *high school student* working on a paper?
- Perhaps a *prospective tourist*?
- A scholar interested in *Eastern philosophy*?
- Someone who can *read Chinese*?

Elaine Rich: User Modeling via Stereotypes. Cognitive Science 3, 329-354 (1979)

Most likely the librarian will make an educated guess, based on the person's appearance:

- age, style of clothing, accent, choice of words, ...

This initial guess might be confirmed or refuted by observations.

- It is assumed that a European cannot read Chinese, unless said otherwise
- Children are generally not (yet) interested in Eastern philosophy, but there are exceptions
- ....

The educated guess, a *stereotype* can be refined with follow-up questions.

Persons expect a *personalized* advice, even though the librarian does not know them.

And the same seems to yield for Web stores.

Jeff Bezos, amazon.com

If I have 3 million customers on the Web, I should have 3 million stores on the Web

Definitions

Hypertext

A Hypertext is a *graph* with uni- or bi-directional *edges*. A *node* represents a text document. An edge between two nodes represents a *link* between two text documents.

Hypermedia

Hypermedia is a *generalization* of the concept of hypertext: apart from text documents, nodes may represent any kind of multimedia (video, audio, graphics, etc.)

Usage of the terms Hypertext and Hypermedia:

- Historically, hypertext and hypermedia refer to different concepts.
- Nowadays one can hardly find 'pure' hypertexts; virtually all documents contain some form of hypermedia.
- For this reason, both terms are currently used as synonyms.

Hypermedia

In 1945, Vanevar Bush envisioned a machine, the *memex*.
By consulting several sources consecutively, a user builds an associative *trail* of related documents, which can be labeled and annotated with notes and comments.
Over time a massive selection of pertinent trails of interests will be built that mimic the user's way of thinking.

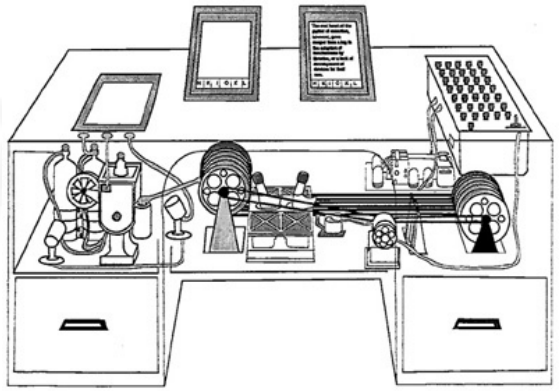
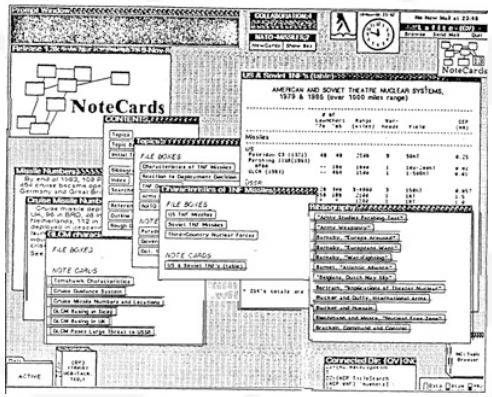


Figure: The memex.

The World Wide Web is the most common form of hypertext. But there have been (and still are) other types of hypertext.

- a hierarchical structured set of statements with cross-references (NLS, Douglas Engelbart, 1960s)
- a universe of parallel versions of documents (revisions, summaries, translations) (Xanadu, Ted Nelson, 1960s - 1988)
- Fixed-sized NoteCards connected by typed links (Halasz, 1988)
- A single scrollable document with buttons to expand hidden material and to jump to different points (Guide, Brown, 1987)



Hyperlinks

A link is not just some underlined text that refers to another document. Links can have different kinds of forms.

- *Typed* links explain the relation between two documents (e.g. glossary, annotation, next)
- *Bidirectional* links connect source and target documents in both directions (i.e. one can follow the link 'backwards')
- *Graphical* links show both source and target document
- *Trails* are links created by the users themselves that can be retraced at a later point (similar to the browser history)
- *Annotations* are links to user-generated remarks (reminders, comments, summaries) or related resources

Three categories of functions that links may serve.

- *Associative links*. Links that are typically embedded in the document (text or media) content. The structure of the hypertext graph depends on the content.
- *Hierarchical links* impose a tree structure on the hypertext, typically visualized as a menu or navigation bar.
- *Temporal links* connect pages already visited by the user. In Web browsers (some of) these links are available via history mechanisms (e.g. back button, history list, bookmarks)

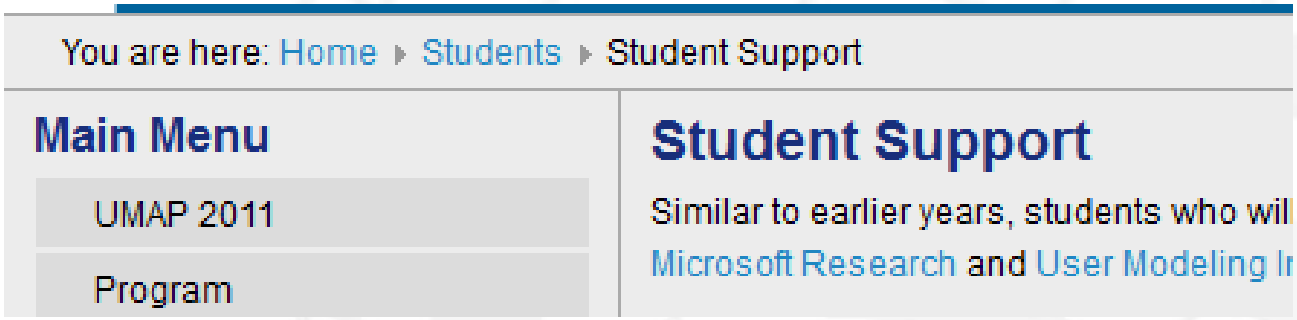


Figure: Breadcrumbs: a type of temporal links.

World Wide Web

World Wide Web

The World Wide Web (WWW) is a platform that combines the networking capacities of the Internet with the concept of hypermedia.

The roots of the Internet lie in the ARPANET, an American government-initiated research project that started late 1966. The WWW was conceptualized by Tim Berners-Lee, the first public demo was at the 1991

The original World Wide Web is a very minimalistic implementation of hypermedia

- no typed pages or typed links
- uni-directional links
- no mechanisms for maintaining link consistency and versioning

The current WWW goes far beyond the original concept of hypermedia

- from manually crafted documents to content management systems and dynamically created content
- from information and content delivery to interactive content and Web applications
- search engines, social networking, linked data

Goals of adaptive hypermedia

The general goals of adaptivity are to improve the efficiency and effectiveness of the interaction. This means that adaptive systems aim to make complex systems more usable, present the users with what they want to see, as well as speed up and simplify the interaction.
This includes:

- helping users to find information
- tailoring information presentation
- recommending products
- supporting collaboration
- taking over parts of routine tasks

Adaptive Hypermedia: A definition

Adaptive Hypermedia

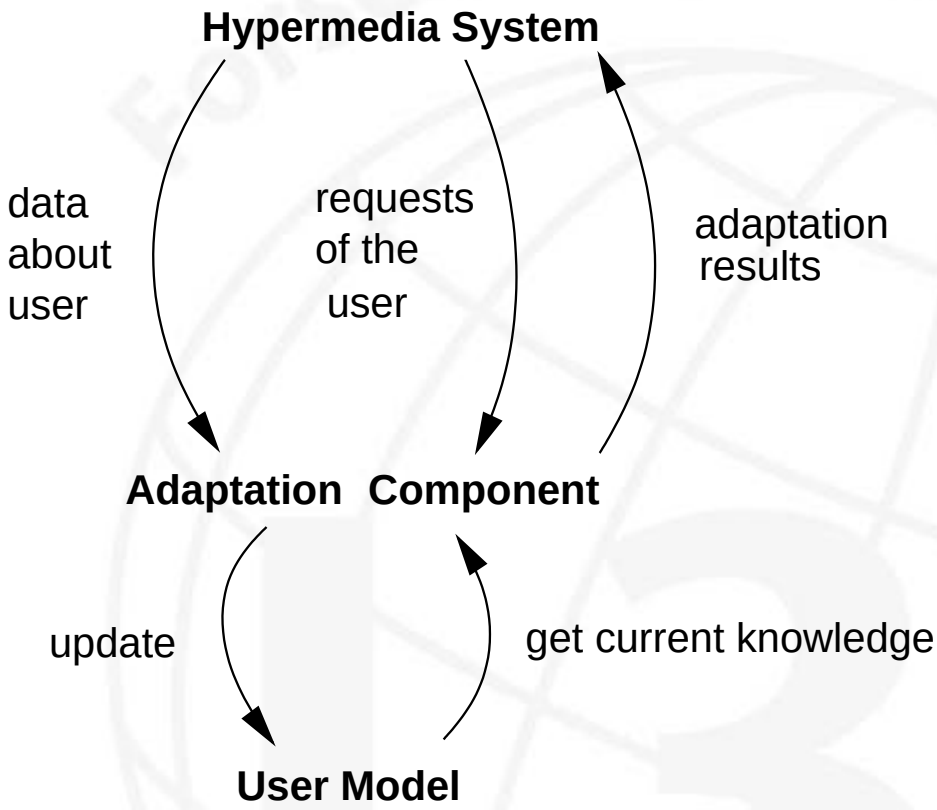
By adaptive hypermedia systems we mean all hypertext and hypermedia systems which reflect some features of the user in a user model and apply this model to adapt various visible aspects of the system to the user.

Peter Brusilovsky: Methods and Techniques of Adaptive Hypermedia. User Modeling and User-Adapted Interaction 6 (2-3), 1996

Different voices on adaptive hypermedia

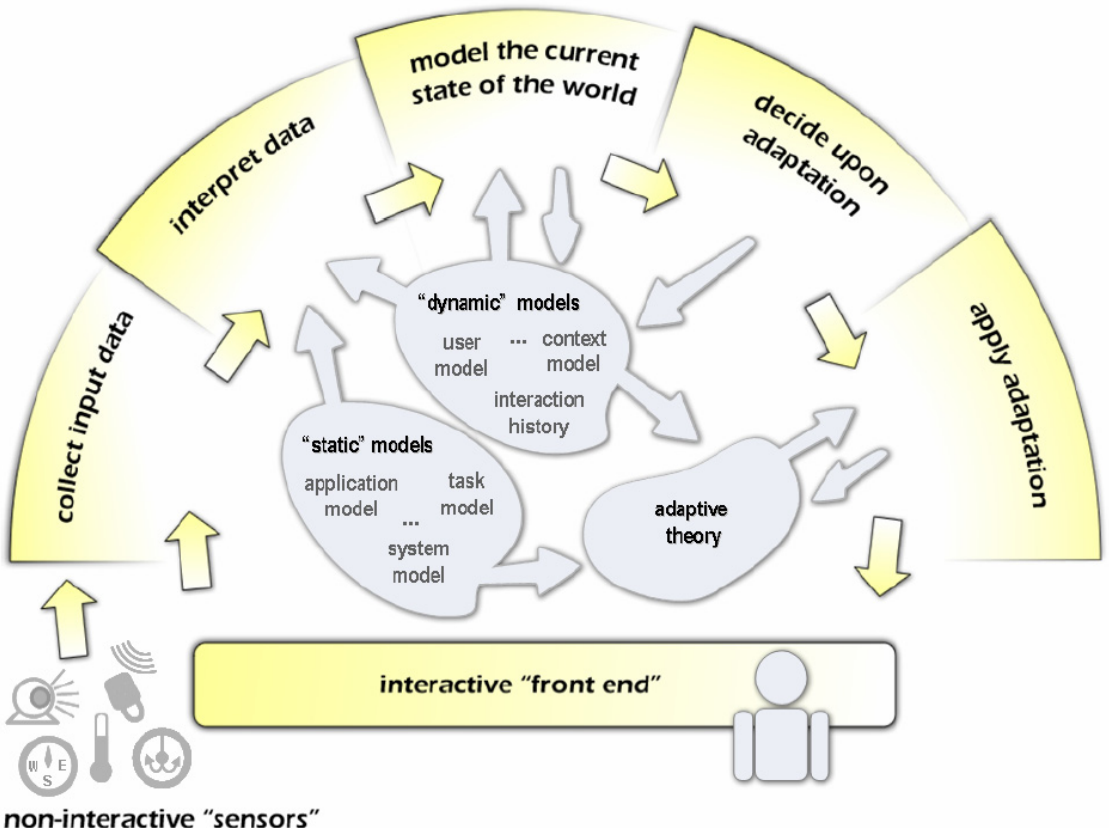
- The ambition of adaptivity is that not only 'everyone should be computer literate', but also that 'computers should be user literate' (Browne, 1990)
- Personalization is a designers' approach to achieve harmony between users, tasks, environments and the system (Benyon, 1993)
- Personalization is an overrated concept. Rather than investing time and energy on trying to predict individual users' needs it would be better to enhance the overall site design. (Nielsen, 1998)

The adaptation process: a data-oriented view



P. Brusilovsky: Methods and Techniques of Adaptive Hypermedia. User Modeling and User-Adapted Interaction 6 (2-3), 1996

The adaptation process: a process-oriented view



Alexandros Paramythis, Stephan Weibelzahl, Judith Masthoff. Layered evaluation of interactive adaptive systems: framework and formative methods. User Model. User-Adapt. Interact. 20(5): 383-453 (2010)

When are adaptations useful?

When the hypertext is *sufficiently large*

- there is a need for guidance, recommendations and information filtering

and is used by people with *different goals and backgrounds*

- there is a need to adapt to resulting differences in needs and user context

Disorientation

Adaptive hypermedia are an answer to the problem of disorientation, commonly known as "Lost in Hyperspace"
When reading a hypermedia document, users continuously have to create their own trail across the document. Moreover, the associative linking invites them to engage in side paths on related topics, to return to the main topic at a later point.

Lost in Hyperspace

When users fail to keep track of their navigation through a document, they might arrive at a particular page and forget what was to be done there, they might neglect to return from interesting side-tracks or they might fail to find some pages that contain relevant information.

The root of disorientation is the additional *cognitive overhead* needed for users to:

- identify their current position in a hyperdocument;
- reconstruct the way that led to this position;
- distinguish among different options for moving on from this position.

Compare this with books:

- in a novel you can easily track how much you have already read and how much there is still to read
- books provide easy-to-remember cues for remembering: page number, position within the page

Everyday examples on the Web:

- Where can I find whether my train is on-time?
- What was the name of that book I found on Amazon?
- Which query did I use to find this list of hotels that I did not save?

Application Areas

- *Educational hypermedia*. Traditionally, the most popular application area. User needs are well-identified: acquiring knowledge on a certain topics. User characteristics (e.g. knowledge level, interest) are well-researched.
- *Online information systems*. Nowadays we would simply refer to them as Web sites. A popular adaptation technique is collaborative filtering and automated recommendations of products or other items
- *E-commerce*. Product recommendations and personalized offers are particularly important in the field of e-commerce

- *Web information retrieval*. The filtering, sorting, categorizing or annotating of search results based on the users' search history
- *Interaction history*. Support for personal information management and the refinding of information and resources
- *Contextualization and localization*. Adaptation of the user interface to the (mobile) device, network connection speed, current location (geocode, country, language)

Adaptation techniques

There are two main categories of adaptation techniques: adaptive presentation and adaptive navigation support.
Remember that hypertext can be modeled as a graph.

- Nodes $\longrightarrow$ **Content-level adaptation, adaptive presentation techniques**
  Which documents to show, which parts of documents to show, order of the contents
- Edges $\longrightarrow$ **Navigational-level adaption, adaptive navigation support**
  Conditional links, adaptive menus, breadcrumbs, graphical overviews, direct guidance

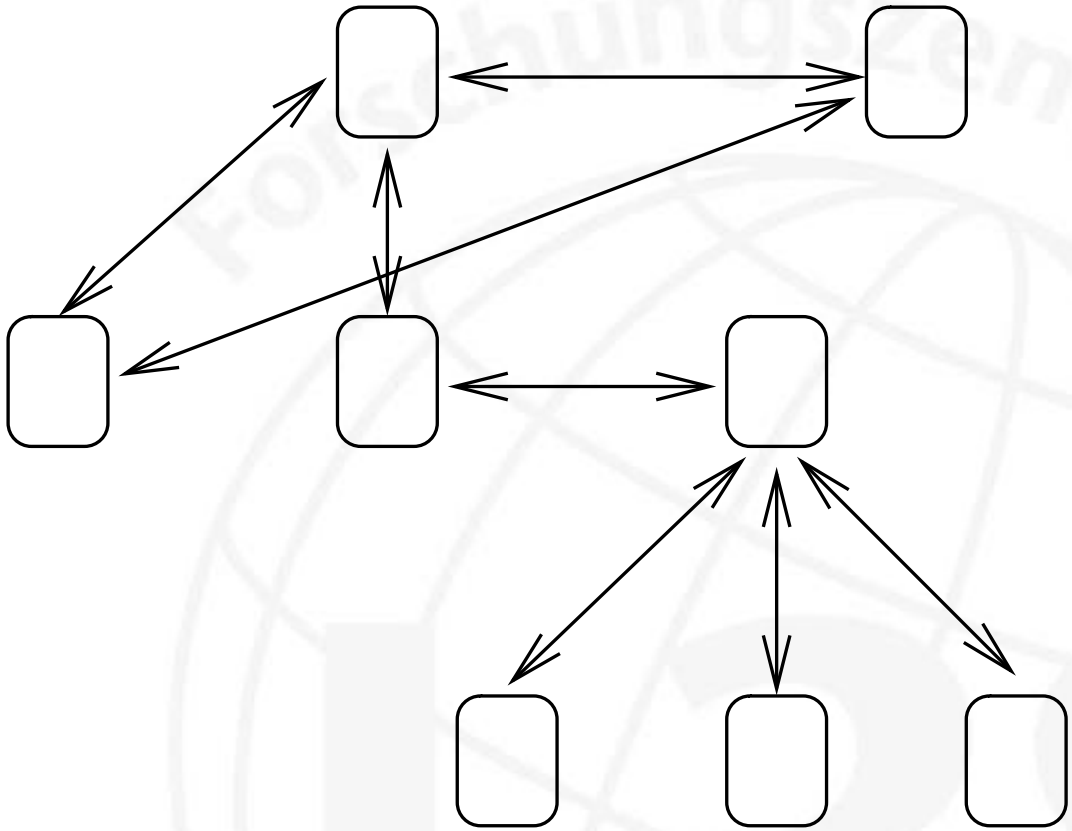


Figure: Hypertext

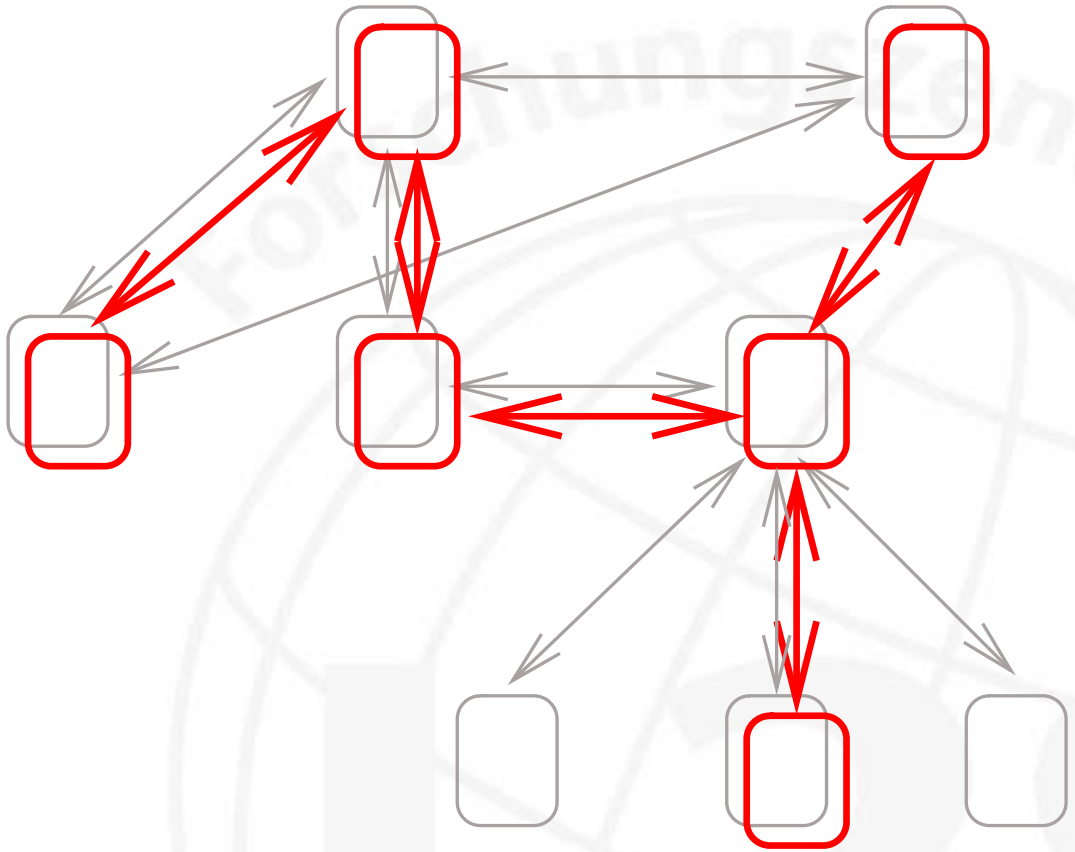


Figure: Relevant contents and links for an individual user at a specific point in time

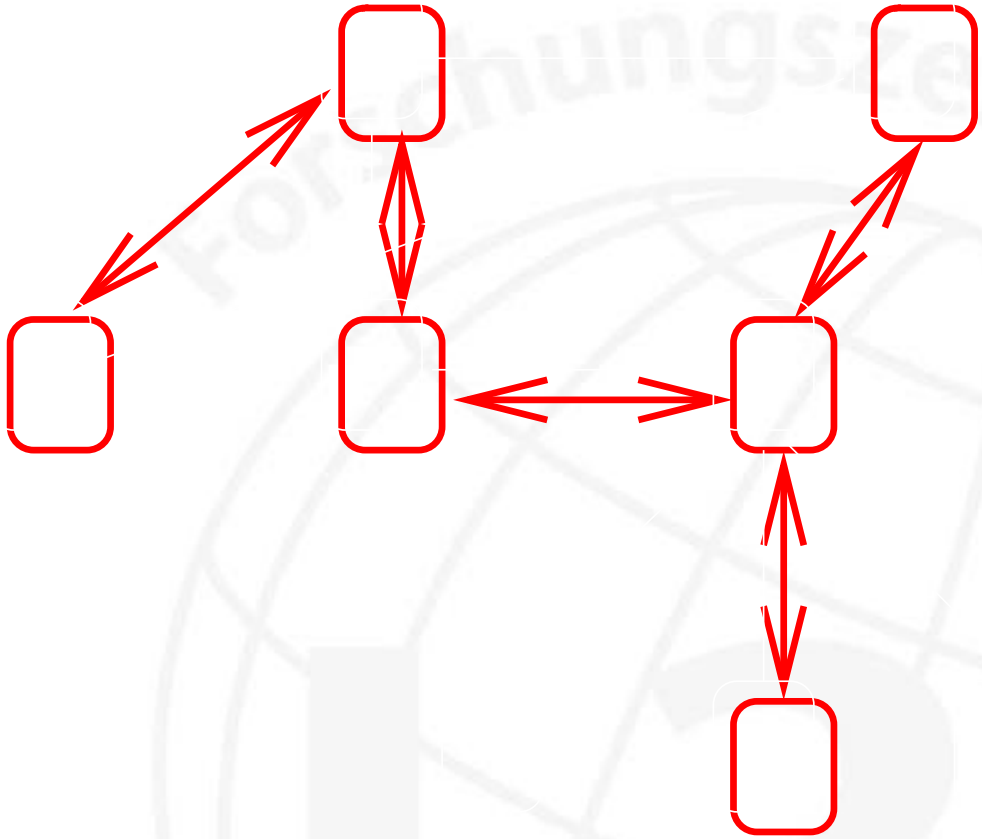


Figure: Adaptive hypertext - for an individual user at a specific point in time t)

Content-Level Adaptation: what can be adapted

Adaptive presentation techniques work on the content level. Items that may be adapted include text, layout, graphics or any other form of multimedia. A significant amount of research has been carried out on *text adaptation*. The general goal of text adaptation is to hide some parts of information that are deemed not to be relevant for the user

Example text adaptation techniques

- Additional text or material: explanations, introductions, definitions
- Summary: only the relevant or important parts are displayed
- Comparison: alternative (versions of) documents are displayed side to side
- Alternative presentation: based on user preferences, device characteristics
- Combinations of the above points

Other content-level adaptation techniques:

- Adaptation of modality: text to voice, mobile devices, reduction of images,
- Personalized presentation: style sheet, contents of choice, background image, alternative designs, skins

Content Level Adaptation: how can it be adapted?

The most popular form of text adaptation is *canned text adaptation*: conditioning the view on certain ready-made text fragments.

- *Conditional Text* – text that is only displayed when certain conditions are met
- *Stretch text* – text that is initially hidden or collapsed, but that can be extended by the users themselves
- *Page fragments / Page variants* – Different versions of the content (e.g. easy, intermediate, advanced) that can be chosen from

Navigational Level Adaption: what can be adapted?

Disabling, removing or annotating associative links can help users to find relevant items more easily.
Disorientation or cognitive overload is addressed with adaptation of the hierarchical (or contextual) navigation aids (menus, navigation bars, graphical overviews):

- *Direct guidance*: next page, next step, menu outline, trails
- *Adaptive sorting*: based on similarity, user background knowledge, relevance, ...
- *Adaptive hiding*
- Continued on next slide

- *Adaptive annotation*: color coding — traffic light metaphor: red (not recommended), yellow (suitable), green (priority)
- *Adaptive overview maps*: site maps, history visualizations
- Combinations of the above

Navigational Level Adaption: how can it be adapted?

Many different technologies that may (and do) differ from system to system

Adaptive versus Adaptable Systems

- *Adaptable* means that users can adapt system behavior themselves (e.g. iGoogle personalized start page)
- *Adaptive* means that the system adapts its own behavior on the users behalf (e.g. Amazon recommendations)

There are in-between solutions: the system suggest the user to execute a particular adaptation (remove a menu item, add a widget to the portal page, register to a particular news feed, ...)

Think of Web sites that you visit on a regular basis. What adaptive elements and what adaptable elements do they have? Which elements are useful and which elements are annoying?

A logic-based definition of adaptive (educational) hypermedia systems

We now introduce a simple adaptive educational hypermedia system, which makes use of logic-based rules for adaptation. E-learning has been an important application area of adaptive hypermedia research.

- clearly defined user goals (learning outcomes)
- established ways to test whether the goals have been met (e.g. tests)
- known individual differences between learners (e.g. knowledge, learning style, learning goals)

Components of a AHS

- 1: *Document space with relations* - A AHS is a hypermedia system: a graph with nodes (content, documents) and edges (links)
- 2: *User model* - Represents properties (e.g. preferences, knowledge, demographics, goals) of a user
- 3: *Observations* - The user model is built from *inferences* based on user interactions with the system
- 4: *Rules for adaptations* - Rules that specify which adaptations (e.g. text hiding, link removal, recommendations) should be applied to the document space, given the properties of the user model

First order logic in a nutshell

First-order logic is a formal system used in mathematics, philosophy, linguistics, and computer science. It is also known as first-order predicate calculus, the lower predicate calculus, quantification theory, and predicate logic.

Terms, constants and function symbols

Objects are represented by terms:

- *Constants:* e.g. D_j, U_i, John
- *Function symbols:* e.g. father-of, successor

Function symbols can be used for mapping terms to another term

- e.g. successor(D_1) = D_2

Terms are simply names for objects.

Predicates

Propositions are represented by a predicate applied to a tuple of terms.

A predicate represents a property of or relation between terms that can be true or false:

- Brother(John, Fred)
- Left-of(Square1, Square2)
- Prerequisite(D_1,D_2)
- Visited(D_4,U_1)
- obs(D_j, U_i, Visited)

In a given interpretation, an n-ary predicate can defined as a function from tuples of n terms to {True, False}.

Sentences and quantifiers

The standard propositional connectives ($\vee \neq \wedge \Longrightarrow \Longleftrightarrow$) can be used to construct complex sentences:

Quantifiers allow statements about entire collections of objects rather than having to enumerate the objects by name:

- the *universal* quantifier $\forall$ D_x asserts that a sentence is true for all values x of constant D_x.
- the *existential* quantifier $\exists$ U_y asserts that a sentence is true for at least one value y of a constant U_y

"Simple": A Simple Adaptive Educational Hypermedia System

This adaptive educational hypermedia system can annotate the links of the hypertext according to the actual knowledge state of the user.
For the annotation we use the so-called *traffic light metaphor*:

- Relevant links are marked with a green ball
- Non-relevant links are marked with a red ball

Simple: Document Space

A set of n constants (n corresponds to the number of documents in the document space) which represent the documents:

$D_1, D_2, \ldots, D_n$.

A finite set of predicates stating the documents that need to be studied before a document can be learned, e.g. D_j is a prerequisite for D_i:

$preq(D_i, D_j)$ *for certain* $D_i \neq D_j$.

Simple: User Model

A set of m axioms, one for each individual user:

$U_1, U_2, \ldots, U_m$.

Simple: Observations

One constant for the observation whether a document has been visited:

Visited.

And a set of predicates

obs(D_j, U_i, Visited) for certain D_j, U_i.

Simple: Adaptation Component

One constant for describing the values of the adaptive functionality "learning_state":

Recommended_for_reading,

and two constants representing values of the adaptive functionality:

Green_Icon, Red_Icon.

Rules for describing the learning state of a document

$\forall\ U_i\ \forall\ D_j$
$\forall\ D_k$ *preq*(D_j, D_k) $\wedge$ *obs*(D_k, U_i, *Visited*)
$\Longrightarrow$ *learning_state*(D_j,U_i, *Recommended_for_reading*).

And rules for describing the adaptive link annotation with traffic lights:

$\forall\ U_i\ \forall\ D_j$
learning_state(D_j, U_i, *Recommended_for_reading*)
$\Longrightarrow$ *document_annotation*(D_j, U_i, *Green_Icon*),
$\forall\ U_i\ \forall\ D_j$
$\neg$ *learning_state*(D_j, U_i, *Recommended_for_reading*)
$\Longrightarrow$ *document_annotation*(D_j, U_i, *Red_Icon*).

Simple 1

We extend our AEHS Simple by an additional rule in the user model UM. The visible adaptive functionality of this system, which we call *Simple 1*, will remain the same as in Simple.
However Simple 1 deduces more information from the user observations as Simple: we assume that a document has been *learned* by a user:

- if it has been visited by the user (as in Simple)
- or if another document for which this document is a prerequisite has been visited (i.e. learned)

Simple 1: Document Space

Same as the document space in Simple.

Simple 1: User Model

As the user model in Simple, plus a rule for inferring that whenever a document has been learned by a user, all the documents that are prerequisites for this document are learned, too.

Simple 1 uses an additional constant for describing user characteristics:

Learned.

A document D_j is assumed to be learned by a user, if it has been visited,

$$\forall\ U_i\ \forall\ D_j$$
$$obs(D_j,\ U_i,\ Visited) \Longrightarrow p\_obs(D_j,\ U_i,\ Learned).$$

or if a document D_k, for which D_j is a prerequisite, has been visited:

$$\forall\ U_i\ \forall\ D_j$$
$$(\ \exists\ D_k\ preq(D_k,\ D_j) \wedge obs(D_k,\ U_i,\ Visited)\ )$$
$$\Longrightarrow p\_obs(D_j,\ U_i,\ Learned).$$

These inference rules *process* an observation, they are therefore abbreviated by p_obs for process observation.

Simple 1: Observations

Same as Simple.

Simple 1: Adaptation Component

The rule describing the learning state of a document is updated as follows:

$\forall\ U_i\ \forall\ D_j$
$\forall\ D_k\ (preq(D_j, D_k) \wedge$
$(\ obs(D_k, U_i, Visited) \vee p\_obs(D_k, U_i, Learned)\ )$
$\Longrightarrow learning\_state(D_j, U_i, Recommended\_for\_reading).$

The rules for adaptive link annotation remain unchanged with respect to *Simple*.

Question: did you spot the unnecessary part of the update rule as displayed above?

Simple 2

We extend our AEHS Simple 1 with tests T for each document D that the learner can optionally perform, to show that he already mastered the prerequisites for learning the contents of document D.

If the learner passes the test, the document will receive a 'yellow' traffic light, indicating that the content *might* be understandable; if not, the document receives an 'orange' light, indicating that the content will become understandable after mastering the prerequisites.

These lights are shown along with the green or red lights.

Simple 2: Document Space

Same as the document space in Simple.
Plus:
A set of n constants (n corresponds to the number of tests in the document space) which represent the prerequisite tests corresponding to the document:

$T_1, T_2, \ldots, T_n$.

A finite set of predicates stating that a test T_j verifies the prerequisites for a document D_i:

testpreq(D_i, T_j).

Simple 2: User Model

As the user model in Simple 1, plus a rule for inferring that whenever a test has been passed or failed by a user, the result is propagated to the corresponding document(s).

Simple 2 uses an additional constant for describing user characteristics:

PassedTest

The prerequisites for a document D_j are assumed to be fulfilled if the user successfully completed at least one of the corresponding tests.

$$\forall\ U_i\ \forall\ D_j$$
$$(\exists\ T_k\ \mathit{testpreq}(D_j,\ T_k) \wedge \mathit{obs}(T_k,\ U_i,\ \mathit{Passed})\ )$$
$$\Longrightarrow \mathit{p\_obs}(D_j,\ U_i,\ \mathit{PassedTest}).$$

Similar as in Simple 1, these inference rules process an observation, they are therefore abbreviated by p_obs for process observation.

Simple 2: Observations

Same as Simple and Simple 1. Plus one constant for the observation whether a test has been passed:

Passed

And a set of predicates

obs(T_i, U_j, Passed) for certain T_i, U_j.

Simple 2: Adaptation Component

The rule describing the comprehension state of a document is updated as follows:

$\forall\ U_i\ \forall\ D_j$ *p_obs*(D_j, U_i, *PassedTest*)
$\Longrightarrow$ *comprehension_state*(D_j, U_i, *Understandable*).

And additional rules for describing the adaptive link annotation with traffic lights:

$\forall\ U_i\ \forall\ D_j$
comprehension_state(D_j, U_i, *Understandable*)
$\Longrightarrow$ *document_annotation*(D_j, U_i, *Yellow_Icon*),
$\forall\ U_i\ \forall\ D_j$
$\neg$ *comprehension_state*(D_j, U_i, *Understandable*)
$\Longrightarrow$ *document_annotation*(D_j, U_i, *Orange_Icon*).

Constants used

System	DOCS	UM	OBS
Simple	$D_1, D_2, \ldots, D_n$.	$U_1, U_2, \ldots U_m$.	Visited.
Simple 1	$D_1, D_2, \ldots, D_n$.	$U_1, U_2, \ldots U_m$, Learned.	Visited.
Simple 2	$D_1, D_2, \ldots, D_n, T_1, T_2, \ldots T_s$.	$U_1, U_2, \ldots, U_m$. Learned, PassedTest	Visited, Passed.

System	AC–Learning State	AC–Adaptive Link Annotation
Simple	Recommended_for_reading.	Green_Icon. Red_Icon.
Simple 1	Recommended_for_reading.	Green_Icon. Red_Icon.
Simple 2	Recommended_for_reading. Understandable.	Green_Icon. Red_Icon. Orange_Icon. Yellow_Icon.

Table: Constants used in *Simple*, *Simple 1* and *Simple 2*.

Predicates used

System	DOCS	UM	OBS
Simple	preq(D_i, D_j).	–	obs(D_k, U_j, Visited).
Simple 1	preq(D_i, D_j).	p_obs(D_j, U_i, Learned)	obs(D_k, U_j, Visited).
Simple 2	preq(D_i, D_j). testpreq(D_i, T_j).	p_obs(D_j, U_i, Learned) p_obs(D_j, U_i, PassedTest)	obs(D_k, U_j, Visited). obs(T_k, U_j, Passed)

System	Adaptation Component
Simple	learning_state, document_annotation
Simple 1	learning_state, document_annotation
Simple 2	learning_state, comprehension_state, document_annotation

Table: Predicates used in *Simple*, *Simple 1* and *Simple 2*.

Summary: Adaptive (Educational) Hypermedia Systems

In our formalism, we used first order logic to describe:

- DOCS: Document Space
  A finite set of first order logic (FOL) sentences with atoms for describing documents (and knowledge topics) and predicates for defining relations between these atoms.
- UM: User Model
  A finite set of FOL sentences with atoms for describing individual users (user groups), and user characteristics, as well as predicates and rules for expressing whether a characteristic applies to a user.

- OBS: Observations
  A finite set of FOL sentences with atoms for describing observations and predicates for relating users, documents / topics, and observations.
- AC: Adaptation Component
  A finite set of FOL sentences with rules for describing adaptive functionality.

Real World Examples

First Generation Adaptive Hypermedia System

Interbook

Peter Brusilovsky, John Eklund, Elmar Schwarz: "Web-based Educations for All: A Tool for Development Adaptive Courseware". Proceedings of the Sevenths International World Wide Web Conference, 1998.

Second Generation Adaptive Hypermedia Systems

NetCoach

Gerhard Weber, Hans-Christian Kuhl and Stephan Weibelzahl: "Developing Adaptive Internet Based Courses with the Authoring System NetCoach". Proceedings of the Third Workshop on Adaptive Hypermedia, 2001.

KBS Hyperbook

Nicola Henze and Wolfgang Nejd: "Adaptation in Open Corpus Hypermedia". IJAIED Special Issue on Adaptive and Intelligent Web-Based Systems, vol. 12, 2001

Third Generation Adaptive Hypermedia System

AHA! 2.0

P. De Bra, A. Aerts, D. Smits and N. Stash: AHA! Version 2.0: More Adaptation Flexibility for Authors.