

A Software Engineering Model for the Development of Adaptation Rules and its Application in a Hinting Adaptive E-learning System

Pedro J. Muñoz-Merino¹, Carlos Delgado Kloos¹, Mario Muñoz-Organero¹, and Abelardo Pardo²

¹ Department of Telematics Engineering, Universidad Carlos III de Madrid,
Avda de la Universidad 30, 28914, Leganés, Madrid,
{pedmume, cdk, munozm}@it.uc3m.es

² The University of Sydney,
abelardo.pardo@sydney.edu.au

Abstract. The number of information systems using adaptation rules is increasing quickly. These systems are usually focused on implement nice and complex functionality for adaptation of contents, links or presentation, so software engineering methodologies for the description of rules are required. In addition, the distributed service oriented Internet philosophy presents the challenge of combining different rules from independent Internet sources. Moreover, easy authoring, rule reuse and collaborative design should be enabled. This paper presents the AR (Adaptation Rules) model, a new software engineering model for the description of rules for adaptation. These rules can be composed as a set of smaller atomic, reusable, parametric, interchangeable and interoperable rules, with clear restrictions in their combinations. Our model enables the distribution of rules as well as rule reuse and collaboration among rule creators. We illustrate our approach with the application of this model to a hinting adaptive e-learning system that generates exercises with hints, which can be adapted based on defined rules. Advantages of the AR model are confirmed with an evaluation that has been done with teachers and learning analytics experts for adaptive e-learning.

Keywords: software engineering, rule modeling, adaptive hypermedia, e-learning, information systems, semantic web.

1. Introduction

There is an important increase in the number of software applications that use adaptation capabilities: several applications personalize their contents based on the users' preferences, give specific product recommendations based on the users' profiles, or customize a portal depending on the source country of their visitors. One of these systems is described e.g. in [1]. Adaptive applications take advantage of the adaptation depending on the specific context or the users' needs to try to provide better services. This adaptation requires the selection of a set of resources for a specific situation (e.g. web contents, exercises, a color, a path to follow, etc.). The selection of these resources can be made with a set of rules.

Adaptation rules can be usually divided from a logical perspective into smaller parts of adaptation. It is also a fact that these smaller parts of adaptation are repeated in other composed rules in the application, as well as in other external applications. A reusable rule philosophy can also be enabled and increased with the current service oriented tendency on the Internet [2-3] that allows to obtain an adaptation rule as the result of the composition of different smaller atomic rules that can be located in different services or/and applications through Internet. An analogy between the atomic parts and the letters in an alphabet can be established. The atomic rules can be reused and combined to form different adaptations as the letters of an alphabet can be combined to form different words. In this way, there is no need to code several times the same atomic rules, which implies a reduction of costs. In addition, other advantages are the simplification for the creation of larger rules, reusability, interoperability, or easier maintainability. The adaptation rules are rules that can perform adaptation and produce adaptive systems, but this is a different term from adaptive rules which would imply that the own rules can change themselves during the time.

Therefore, a lot of advantages can be obtained if a proper software engineering model is applied for the description of these adaptation rules. The existing software and web engineering methodologies do not go into deep detail for the description of adaptation rules and they do not discuss about their atomicity, reusability or restrictions.

As a result, the benefits of traditional software engineering are not obtained for this aspect, adaptation modules are usually a single-one piece whose parts cannot be combined and reused as services with other systems, and system rules must be designed each time for each information system.

Different works outline the challenges of supporting different adaptation components to work together [4] or to make flexible and extendable adaptation systems [5-6]. Moreover, Aroyo, and Dicheva [7] described two main challenges of semantic web in education where adaptive applications are important: achieving interoperability among different educational systems, and automation of the authoring process. Specifically, Karampiperis, and Sampson [8] cited the following problems in the design of rules: inconsistency, confluency, and insufficiency.

The main contribution of this work is to present a new software engineering model, the Adaptation Rule (AR) model for the description of adaptation rules in a service oriented environment that solves the previous issues, and its application to a hinting adaptive e-learning system. This new model has a strong focus on rules and enables the typical advantages of software engineering. In addition, the model enables the definition of atomic rules and the combination and reuse of atomic rules from different sources according to restrictions.

The AR model provides a solution for several rule components working together that are flexible and extensible through the Web in line with the commented challenges previously cited in [4-5-6] and overcoming the rule design issues cited in [8].

In addition, our proposed rule model approach can also help in both challenges presented in [7]. On one hand, the model enables interoperability at the low level of rules. On the other hand, the model can help in the authoring process, filling in a set of patterns for the description of rules. Moreover, the model facilitates the collaboration among different rule designers and creators, enabling a common way for the interchange of descriptions of rules. Nevertheless, the final codification in a rule language must be done by programmers or a translator.

Section 2 is devoted to the related work. The proposed model and its advantages are explained in section 3. Section 4 presents a case study for a hinting adaptive e-learning system driven by semantic web technologies where the model is applied. An implementation in Notation 3 (N3), which is a language for describing semantic annotations and rules, of some atomic rules is shown, as well as the composition of a larger rule for the adaptation of hints. In addition, there is an evaluation of the advantages of the AR model with teachers and learning analytics experts for the e-learning area. Section 5 relates our approach with other works. Finally, section 6 summarizes our conclusions.

2. Related Work

2.1. Software and Web Engineering for Rule Models

Different software engineering methodologies exist that describe software programs in a formal way. These methodologies include adaptation algorithms such as [9] or [10], but they are not specific for rules for adaptation. Nevertheless, these methodologies can be applied for any possible algorithm, not being optimized for adaptation rules. Moreover, they do not take into account issues such as the distributed oriented Internet tendency.

The works [11] and [12] present generic models for different software adaptive applications but from a more generic point of view and without focusing on specific rules. In this way, in [11] the target is complex e-business systems that are adaptive and propose the PCBMER architecture that divides a system in six layers that can give services to higher layers and consume services from lower ones. There are components for each layer, but there is not a specific engineering of rules. In [12], the target is to propose a software engineering method for the development of Web applications in Content Management Systems (CMSs). For this purpose, two new modeling languages are proposed which are for general web applications, but there is not detail of the modeling of rules.

Some works propose solutions for helping in the development of software agents in distributed environments, such as in [13]. The work in [13] proposes a new Domain Specific Language (DSL) for semantic web technologies, but the focus is not on rules.

Moreover, Wang [14] presents an initial rule model for self-adaptive software. There are many differences with respect to the work presented here. First, the rule model in [14] is based on events and on the “if/then” construction, while we present a model based on discarding different options from the initial possibilities. Second, the rule model in [14] is intended for self-adaptive software, that is, for software that can change over time, while our model is for adapting application resources, but the code is not self-adaptive. In addition, the model in [14] is not oriented to having a couple of rule properties, which are defined in this work.

There are general approaches for modeling different web applications that fit better depending on the purpose of the web application. In this way, WebML (Web Modeling language) [15] is intended for web applications with an intensive use of data, WSDM

(Web Semantic Design Method) [16] is more user oriented, OOHDM (Object Oriented Hypermedia Design Method) [17] and UWE (UML based Web Engineering) [18] are more abstract methods which can be used in more general web applications, and OO-H (Object Oriented Hypermedia method) [19] is used for web applications with different interfaces. These methods allow the description of web applications, and specifically for adaptation using UML or extensions of UML. Nevertheless, they do not focus on the rules, their reuse, compositionality or atomicity.

2.2. Adaptation in Educational Information Systems

As our case study is about a specific adaptive educational system, this section is devoted to adaptation in educational information systems.

Adaptive hypermedia systems combine hypermedia techniques with user modeling [20]. Adaptive techniques for hypermedia systems were presented in [21], dividing the possible adaptations in two groups: adaptive presentation (the contents to present are adapted) and adaptive navigation (the different links to follow are adapted). Adaptation criteria usually follow some pedagogical guidelines, or specific empirical studies. Studies such as [22] permits to adapt based on the idea that exploratory students learn better in an open environment or [23] that suggest the positive effect of the adaptation of hinting techniques. Several adaptation systems for courseware contents have been implemented such as the described in [8] and [24].

Our proposed model is applied as a case study for an adaptive hypermedia system that generates hints in e-learning using Semantic Web techniques. The provision of hints is a successful strategy (e.g. [25], [26]) in which a tutor helps the student to solve an exercise. Some examples of such hinting systems are described in [27] or [28]. Specifically, several systems provide personalized hints adapted to users using techniques different from the semantic web such as Bayesian Networks student models [29], [30], or the Item Response Theory [31].

Systems for adaptation can be built using semantic web technologies using this reasoning capability that is based on the definition of rules. The semantic web technologies can also take advantage of the distributed service oriented tendency of applications that allows taking data from different sources on the Internet.

Educational ontologies have been defined to describe course topics [32], or ontology mappings [33]. There is a project with a repository of educational ontologies [34]. In addition, some ontologies are taken from e-learning standards for the description of educational resources, such as Dublin Core [35], IEEE-LOM [36] (Learning Object Metadata) or IEEE PAPI (Public and Private Information) [37]. Relationships among other e-learning standards and the semantic web have been studied, such as for SCORM (Sharable Content Object Reference Model) [38], or IMS-LD (Learning Design) [39], [40]. Both of these specifications cover different issues but not hints in assessments.

A review of semantic web in education and specific adaptation applications can be seen in [41]. Mizoguchi, and Bourdeau [42] exposed how to use ontological engineering in order to address Artificial Intelligence in Education. Some works show the creation of resources and rules for adaptive learning, using semantic web techniques such as Henze, Dolog, and Nejdil [43] which explains adaptation for different issues of contents,

Cheniti-Belcadhi, Henze, and Braham [44] for assessments, Vassileva, and Bontchev [6] for selecting a student learning path or Jovanovic et al. [45] for generating feedback.

There are also other educational systems that use rules for adaptation purposes but do not use semantic web techniques. APELS (Adaptive and Personalized E-learning System) [46] calculates the best learning path based on a quality of service measure based on the idea of the learning curve of students, [47] for the adaptation of contents or [5] for the adaptation of different HTML tags for web pages based on rules.

3. A Model for the Description of Adaptation Rules

The proposed model establishes that an adaptive software application has to choose one or more specific resources in a specific moment (e.g. a complete Web page to present to a user). The selection might depend on many different factors such as the user preferences or her context. A resource is anything that can be selected by the software application. Each resource is represented as R_k .

Let be S the space that represents all the possible resources which can be selected for the adaptation software in a moment (e.g. all the possibilities of Web pages that can be generated). Each $Res_k \subset S$, can be modeled as the union of different aspects which can be adapted (e.g. a web page can be the union of its content, presentation, possible user interactions and links to Web pages). These adaptation aspects can have other different sub-aspects (e.g. the links can be external or internal, or the presentation can be divided into color, font, etc.) and so on in a hierarchical relationship, forming a tree. This tree is a graphical representation of all the adaptation aspects which influence the process.

Let's consider that each adaptation aspect is a node. There will be some leaf nodes of the tree that cannot be further divided into other aspects. These indivisible final leaf nodes are specific adaptation variables. Let be x_i one of these variables. An adaptation variable can be of the following different types:

- Numerical. It might be integer, float, etc. A specific example can be the amount of penalties on the scoring for a specific exercise for a student (e.g from 0 to 10).
- String. An example can be some adaptive text to show to each student as feedback.
- Categorical. An example is a set of web page contents which can be selected.
- Ordinal. An example is 3 different grades for students (bad, average, good).
- Vector. An example is a set combination of whatever of the previous variable types.

Each R_k is represented as the combination of all its values for its adaptation variables x_i . The value of each x_i can also be a special one to denote that does not exist for a resource. The following vector represents whichever $R_k \subset S$:

$$R_k = (x_0 \ x_1 \ x_2 \ .. \ x_i \ ... \ x_{MAX-1}) \quad (1)$$

The different aspects for the same hierarchical level must be disjunctive, so an adaptation aspect must not include anything of another one. But there can be one type of dependency among one aspect and other aspects (e.g. it is necessary to know the number of required exercises to show to the student before selecting these exercises). This dependency will be also traversed to the specific aspects of each branch until the leaf

nodes denoted by variables x_i . Therefore, a variable x_i can depend on others for obtaining its value. For each x_i a dependency function must be defined as:

$$x_i = f(x_0, x_1, x_2, \dots, x_j \dots x_{T-1}) \tag{2}$$

The value of x_i can only be obtained when the variables that depend on it have received some value. Therefore, this model requires that there is a safe sequence of variables so that the adaptation engines can be executed in this order so that each variable can obtain a value because its dependencies have been previously solved (similar concept as safe state for processes in operating systems for deadlock problems in inter process communication). These dependencies will be reflected later in the model for the description of each rule associated to a x_i , as “required inputs” as well as the specific dependency description.

For each indivisible adaptation aspect represented by the variable x_i , there are a set of specific associated rules (e.g. for determining the next web page content to show to a student, there can be a rule for filtering web page contents that were already visited by this student, another one for filtering web page contents that cover concepts the student already master, or another one for taking web page contents depending on if the student is exploratory or non-exploratory).

Let be R_m a rule of our model related to a x_i variable. A rule can be defined such as a processing block that receives (e.g. from a file) the information with the possible candidate options (PO_i) to select for that variable x_i (in general a subset of all the possibilities for x_i) and other dependent adaptation variables (x_j) or/and parameter values (p_j) (e.g. student preferences or probabilities); and returns the possible values (SO_i) to select for x_i and other output parameters (gp_i) that can be used in other rules. A rule is represented in figure 1.

A rule can be expressed as a function in the following way:

$$(SO_i, gp_i) = R_m(PO_i, x_0 \dots x_j, p_0 \dots p_j) \tag{3}$$

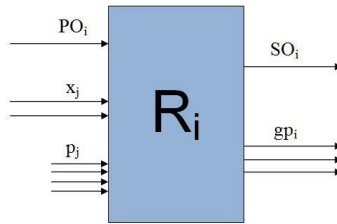


Fig. 1. A graphical representation of a rule of the model

PO_i and SO_i can be given in different formats which also depend on the type of variable. For example, as candidate or discarded options: expressing that are equal or different to a set of values, in specific intervals, greater or less than some values.

Let be I the set of all possible input parameters of a rule without taking into account PO_i . A rule of the model must follow the following property:

$$(\exists PO_i \subset S) \text{ and } (\exists c_k \subset I) / R(PO_i, c_k) = SO_i \text{ and } (SO_i \neq PO_i) \tag{4}$$

This property sets that a rule must have at least some input combination for which the application of the rule makes to change some of the selected options with respect to the initial possible options for x_i . Therefore, the concept of rule in our model is associated to have some effect on the possible selected values of a specific aspect to adapt.

3.1. Types of Combination of Rules

The rules of the model can be combined to form new composed rules with the following types of combinations (fig. 2): sequence, union, parallel and condition.

Sequence Combination. In the sequence combination, the possible options to select are firstly filtered by the first rule and next by the second rule, among the candidates that were the output of the first rule (fig. 2, a). An example of sequence combination for the selection of a hint exercise (categorical variable) is the following:

- Rule 1: The hint exercises that are selected are such exercises that cover at least some concept which is also covered by the root exercise.
- Rule 2: The hint exercises that are selected are the ones which estimated difficulty level by the teacher is closer to the student knowledge level in such concept.

A sequence combination can be represented as:

$$\text{If } R_i(\text{inputs}_i) = (\text{SO}_i, \text{gp}_i) \text{ then SEQUENCE } (R_1 \dots R_N) = \tag{5}$$

$$(\text{SO}_1 \cap \text{SO}_2 \dots \cap \text{SO}_N, \text{gp}_1 \cup \text{gp}_2 \dots \cup \text{gp}_N)$$

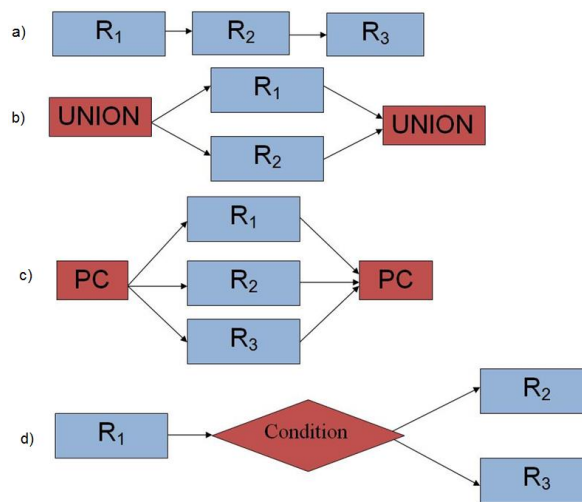


Fig. 2. Sequence (a), parallel (b), union (c), and conditional (d) rule combinations

Let be $inputs_i$ the inputs of the correspondent rule (previously calculated adaptation variables required plus necessary parameters). The sequence combination can be

established between rules for the same adaptation variable x_i but it can also be applied between rules that decide on different adaptation variables. In the later case, each rule filters on a different final aspect.

Union Combination. A union among N rules denotes that the selected options of the obtained rule are the union of the selected options by each rule independently (fig. 2, b). An example of a union with two rules for the selection of the next web page is:

- Rule 1: The next web page contents that are selected are the ones that a user has not already seen.
- Rule 2: The next web page contents that are selected are the ones for which the user has a strong probability to buy some product.

The resulting rule will give web page contents which have not been selected plus the web pages with high probability of the user to buy something.

A union combination from R_1 to R_N can be represented as:

$$\text{If } R_i(\text{inputs}_i) = (\text{SO}_i, \text{gp}_i) \text{ then UNION } (R_1 \dots R_N) = \quad (6)$$

$$(\text{SO}_1 \cup \text{SO}_2 \dots \cup \text{SO}_N, \text{gp}_1 \cup \text{gp}_2 \dots \cup \text{gp}_N)$$

From fig. 2, b, there are two union boxes. Rules in a union combination can be for the same adaptation variable or a different one. The rules R_1 and R_2 have both the same input possible options PO_i but they might have different input parameters p_j . In case that rules might have also different input possible options PO_i , then the initial union box should be removed and both rules would come from different branches.

Parallel Combination. In the parallel combination (PC) the options of an adaptation variable x_i are selected taking into account the different rules that are in parallel, as a tradeoff among all the rules for the decision (fig. 2, c). An example of this combination can be for deciding the background color of a web page, and the following rules can be used for the decision:

- Rule 1: According to the user color preferences.
- Rule 2: According to the device a user is accessing to.
- Rule 3: According to the company preferences.

There are many different types of parallel combinations. Indeed, the previous union can be seen as a subtype of it, but for its importance was presented independently. In any case, the semantics of each PC should be provided. An example is the linear combination for obtaining a number or an ordinal variable. N rules might select a number each one (r_t), and generate the strength of the decision ($strenght_t$) as a parameter (a number from 0 to 1). Some coefficients (w_t) can denote the importance of each rule criterion for the decision. The number is obtained in the PC with the following formula.

$$PC = \sum r_t w_t strenght_t \text{ for } t \text{ from } 1 \text{ to } N \quad (7)$$

A specific example of the linear combination can be for calculating the amount of penalties for each incorrect attempt for a student in an exercise. This can take into

account the teachers' difficulty criterion for that question (rule 1), the users' preferences (rule 2) and the previous students' interactions with the system (rule 3).

Finally, the conditional combination provides several paths to choose. The decision about which path to choose will be based on the output selected values of the associated rules and their generated parameters and/or other external parameters. Fig. 2, d shows an example of a conditional combination. An example can be that depending on a rule that decides if a user is going to see a web page content or a survey, the next rule paths will be different, e.g. for the survey, the different question resources should be selected.

Conditional Combination. This combination can be seen as an if/else structure that can check a lot of variable values in order to select the next path of rules to execute.

Atomic Rules. An important concept of the model is atomic rules. A rule R_i is said to be the same of another R_j if and only if:

$$(SO_i, gp_i) = R_m(PO_i, x_0 .. x_j, p_0 ... p_j) \quad (8)$$

A rule is said to be atomic if that rule cannot be formed as the combination of two or more rules using any of the combinations, being the resulting rule the same as the original rule. Each atomic rule must be described with a textual notation containing:

- *Code*: This is a unique rule identifier that must reflect the adaptation issue that this variable can influence in, according to the hierarchy of the adaptation aspects.
- *Description*: It must clearly describe their purpose.
- *Parameters*: All the possible input rule parameters (p_j) must be enumerated and described. They can be e.g. certain probability, or user model features.
- *Parallel combination restrictions*: For each different type of parallel combination in which this rule can be used, all the possible rules which can be combined.
- *Required sequence inputs*: The rules which must be applied in a sequence relationship branch before, as well as the output values restrictions that must have the dependency of this adaptation variable with respect to other variables since $x_i = f(x_0, x_1, x_2, .. x_j ... x_{T-1})$, but also other possible additional dependencies that are imposed by the rule.
- *Output sequence*: The rules that can be after this rule in a sequence relationship.

3.2. Advantages of the Proposed Model and Properties of the rules

Next, there is a description of the advantages of the proposed model. Several of them imply a solution to challenges commented in previous works (see the Introduction section). These advantages are based on certain properties of the proposed rule model.

Enables several adaptation rule components working together in a distributed manner. An atomic rule (defined in subsection 3.1) can be seen as an adaptation component. The atomic rules can be combined to form and compose larger rules with the presented four types of combinations (subsection 3.1). Our model enables this compositionality by defining the ways of combining the different rules with the four

proposed combinations and the definitions of rules as well as the way to define possible restrictions in these combinations (subsection 3.1).

Atomic rules from different sources (e.g. different machines in the Internet) can be combined in a distributed way with the presented rule combinations of the model (sequence, union, parallel combination, and conditional). From a technical point of view, a web service architecture would be needed so that different procedures can be invoked from other machines. An atomic rule to be executed in a machine should have available some inputs that are the outputs of other rules (selected options plus generated parameters). The exchange of this information (of selected options plus generated parameters) from one system to another can be done in many different ways and data formats. Although this information exchange is outside of the scope of our proposed rule model, we illustrate a specific possible solution. For example, using semantic web techniques, an atomic rule can receive the inputs from different files in RDF or N3 format. The outputs can also be generated e.g. in RDF or N3. Reasoners in different machines can understand these types of files, even if reasoners are different or are implemented in different programming languages as they work with standardized files.

As an example, an input file or output file for any rules in our hinting e-learning adaptive system is the following in N3, describing all the different candidate options for a categorical variable (the resources to adapt are the hint exercises to present to a student) that can be selected (x_i in the general notation of the model) which are exercises in this case (from $h1$ to hn) that can be candidates to be a hint of a specific exercise ($e1$) and which of these candidate options have been previously discarded in just the previous rule. It can be the case (e.g. when applying the first rule) that there are not any discarded hints previously so the `:discardedhint` indicator would not be present:

```
:e1      :candidatehint :h1
          :h2,
          ...
          :hn .
:discardedhint :h2
          ...
          :hi
```

In addition, the input parameters (p_i in the general notation of the model) might also be passed as input files if required by the rule. We can see two examples of these input files in our hinting adaptive e-learning system in subsection 4.1 for describing users and exercises. There should be an agreement on how to pass the information in an interoperable and reusable way, and this is an orthogonal aspect with respect to our model, since our model focuses on behavior interoperability, and this is a data compatibility issue. For example, an agreement can be done so that all parameters follow different data interoperability standards such as IEEE-PAPI for the description of learner model features, Dublin Core or IEEE-LOM for the description of contents.

All atomic rules should be prepared to interpret the input files with the candidate options and the discarded options generated by other atomic rules as output files. This way, they can be interchangeable if restrictions are fulfilled because the rules will operate over the same input files with the candidate and discarded options, no matter of the previous rules that were applied. For example, in our hinting e-learning adaptive system case study, all the atomic rules have a first part that is the following:

```

@forall :problem, :hint. {
  :problem p:candidatehint :hint.} log:implies{
  @forall :f. {
    <file:/// conclusions1.n3> log:semantics :f .
    :f log:notIncludes { :problem p:discardedhint :hint
  } .} log:implies {:problem      p:possiblehint :hint . }.

```

This part of any rules sets that an exercise has as possible hints the ones that are candidate but had not been discarded. This is marked as *p:possiblehint* for the following parts of the rule. This initial part is common to all the atomic rules in order to guarantee that a rule can be combined with others in the sequence relationship.

An example of execution of a rule (*Rule.n3*) with the CWM reasoner with parameters of the annotations of the exercises (*AllExercises.n3*) which are automatically searched and retrieved from a directory and the output of the execution of another atomic rule (*conclusions2.n3*) is the following:

```

cwm AllExercises.n3 conclusions2.n3 Rule.n3 --think --
filter="Filter1.n3" > conclusions3.n3

```

The annotated data from the exercises (*AllExercises.n3*) are taken as parameters of this rule (other rules have also e.g. as input parameters the users annotations according to the learner model), and CWM reasons over these data (option *--think*) according to the written rule (*Rules.n3*). The results are obtained into the *conclusions3.n3* file. The option *--filter* is in order to obtain a *conclusions3.n3* file with only the information filtered by the rules in *Filter1.n3*. This is to obtain only conclusions that we desire (in this case the candidate hints and discarded hints for the root exercise). This filter assures that the output file only contains the candidate and discarded hints as it is expected by the input of atomic rules. The *Filter1.n3* file is the following:

```

@forall p:problem, p:hint.
{ p:problem p:candidatehint p:hint }
log:implies { p:problem p:candidatehint p:hint }.
{ p:problem p:discardedhint p:hint }
log:implies { p:problem p:discardedhint p:hint }.

```

Regarding the union type of combination, the same data format of candidate and discarded hints enables it in an easy way. The atomic rules of the union can be executed independently and the final output files of each one would be files with the candidate and discarded hints. A new execution of CWM would be required receiving as input all the output files of the rules of the union. This CWM execution would give as output result the union of candidate hints and as discarded hints just the ones that are discarded by all the rules in the union. Finally, the conditional relationship is straightforward and might be taken with any code that evaluate a condition based on the information of an RDF or N3 file and depending on the result invoking the CWM reasoner in one way or another to apply different rules.

Since all of this required information for a rule to be executed can be done according to agreed ontologies, and that all of this can be written in files in RDF (there are automatic translators from N3 to RDF and the another way), then all adaptation

reasoners (with independence of if it is CWM, Pellet, Drools, etc.) can read these RDF files with established data according to ontologies. Therefore, interoperability and reuse are enabled and rules in different machines of Internet can be executed for forming a higher level rule. In this way, rules can be created in a distributed way in Internet and combined from different sources. The final element in charge of the presentation would receive a final RDF file with the selected options as a result of the processing in different sources.

This compositionality implies a lot of advantages such as not having to repeat code as atomic rules can be reused in different cases, distributed programming or specialized code.

Improvements in authoring, reuse, completeness, collaboration and maintainance.

Usually, key stakeholders (like teachers or learning analytics experts in adaptive e-learning applications) cannot contribute actively in the design and authoring of the adaptation rules, because it is too difficult for them to describe the rules. Our model enables a way of describing rules with graphs and natural language textual notations so that these type of stakeholders can easily contribute in the authoring process. The authoring process for applying our model consists on the following phases:

- 1) Determination of the general adaptation resource and composition of a tree with all the different adaptation aspects and the leaf nodes of adaptation. For each aspect that can be divided into other adaptation aspects, a graphical diagram might be provided.
- 2) For each leaf node, determination of the type of variable that represents such leaf node and its possible range of values. A textual representation of it is required.
- 3) Identification of all the atomic rules for each leaf node with their textual descriptions.
- 4) Graphical description of higher level usual rules as the composition of atomic rules using the four different explained relationships.
- 5) Implementation of all the atomic rules using some semantic web rule language (e.g. N3), programming language, etc.

Stakeholders who have a strong background for the description of semantics of adaptation (e.g. teachers might have a strong knowledge about how to apply pedagogical strategies to adapt resources to students) can be in charge of making all of these phases of this model with the exception of phase 5 that implies programming skills, which would be done by software experts.

Our model enables the integration of key stakeholders in the authoring process. Therefore, applications can take advantage of the strong knowledge of these stakeholders on the design of rules, using their valuable knowledge about best practices for adaptation. Without a model, this would be very difficult since key stakeholders will not have a way to describe rules in a convenient and easy way and cannot give their results to other experts in a standardized way.

This improvement of the authoring process driven by the model implies other set of related advantages:

- Collaborative creation. Different rule designers and creators (e.g. teachers) can interchange their models in a standard way which is understandable by all, so it promotes the collaboration in the creation of rules.
- Reuse. Existing public rule descriptions, or which are accessible with the description of

the model, can be reused by other designers or improved in a moment.

Other advantages related to the authoring process are the improvement about the completeness and maintenance of the possible rules. The improvement in completeness and maintenance are derived directly by the property of atomicity. Designers must decide the aspects and atomic rules in a way that they cannot be divided into smaller chunks with the four types of defined combinations. As each atomic rule specifies its own combination restrictions with others, then a designer has only to focus on each atomic rule and their possible combinations with others, but not on the many global rule possibilities (composed by several chunks) which can make to forget some of the possibilities and not being complete or making more difficult the maintenance.

In any case, all of these advantages related to the authoring process are based on an analytic reasoning so far. Nevertheless, they should be confirmed by the opinion of authoring creators for not being just hypotheses based on reasonings. In this direction, an evaluation with teachers and learning analytics experts is presented in section 4.5. This evaluation confirms these advantages in the field of e-learning adaptive systems.

4. Case Study: Hinting Adaptive E-learning System

This section presents a case study of how the proposed model can be applied for a specific adaptive hinting adaptive e-learning system. This system provides learners with exercises which must be solved. The main objective of this system is that students can learn while solving exercises and if students are stuck, the system can help them giving different types of hints. These exercises can have related adaptive hints to help students in their learning process. These hints can depend e.g. on the student profiles, the subject topics or the course materials. Hints can be text but also other exercises.

4.1. Overview of the hinting adaptive e-learning system

A general architecture for combining semantic web techniques with Intelligent Tutoring Systems was proposed and a specific implementation of this architecture for adaptive hints was described [48]. Fig. 3 shows this architecture, where the graph has been adapted from [48] for a better understanding of this case study.

The learner model provides information about the learner knowledge level in each of the different course topics. The IEEE-PAPI specification is used for describing it. Next, there is an example for describing the students' knowledge level for the *errorCode* and *closeRemoveSem* course topics of Interprocess communication. The following description states that the student has a performance level of 0 in the *errorCode* concept, while a 9 in the *closeRemoveSem* concept.

```
le:errorCodeU1 a papi:Performance.
le :closeRemoveSemU1 a papi:Performance.
:U1 papi:hasPerformance le:errorCodeU1.
:U1 papi:hasPerformance le:closeRemoveSemU1.
le:errorCodeU1 papi:learning_competency ipc:errorCode.
```

```

le:errorCodeU1 papi:performance_value "0".
le:closeRemoveSemU1 papi:learning_competency
ipc:closeRemoveSem; papi:performance_value "9".
    
```

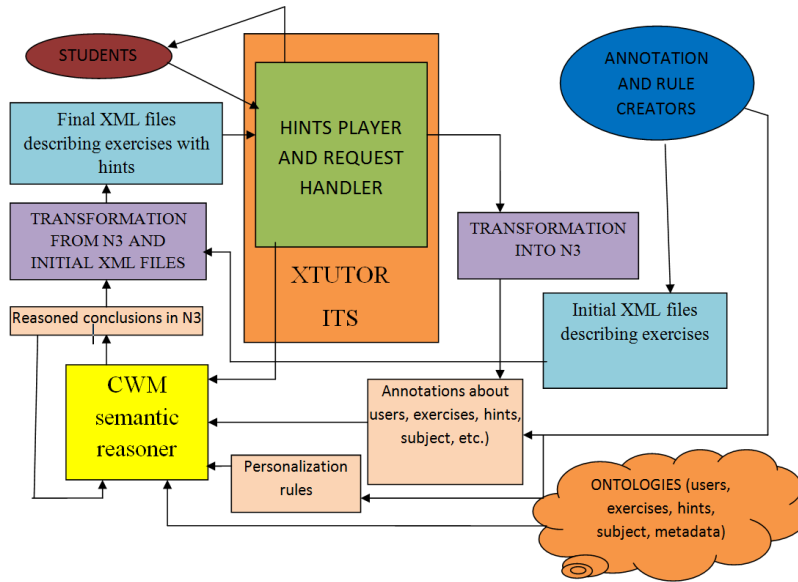


Fig. 3. Specific architecture of the hinting adaptive e-learning system

The learner model is used to make decisions about the different hints to generate for different learners. In addition, there are other information that is used for the adaptation and personalization: course topics, and information about exercises and hints.

The concepts involved in each specific course topic and their relationships must be described. Fig. 4 shows the concepts involved in a Semaphore Inter Process Communication (IPC) lab and the hierarchical structure relationships between them. It shows the different parts in which each concept is divided. Each part is another concept that can be again divided in other concepts and so on.

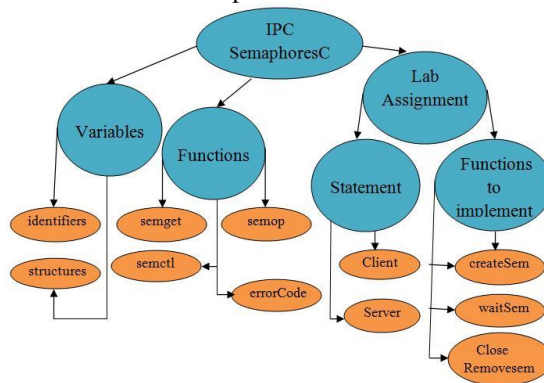



Fig. 4. Lab IPC concepts and relationships

An N3 annotation file describes this diagram of Fig. 4. Each circle represents an IPC concept and this is expressed as e.g. `:semget a :IPCConcept`. An arrow denotes that some concept has as part another one and this is expressed using the `dc:hasPart` property from the Dublin Core, e.g. `:Functions dc:hasPart :semget`. It is possible to use ontologies and annotations of concepts which describe richer types of relationships among them, as e.g. in [32]. The exercises and hints are annotated with the correspondent information which is used for the adaptation. Next, there is an example of exercise annotation:


```



p:P1 a p:Problem.
p:P1 dc:subject ipc:errorCode, ipc:closeRemovesem.
p:P1 p:mayberoot 1; p:maybehint 0.
p:P1 p:difficulty "8".
    
```


For the simulated FAT of the lab (You will lose 3 points for each incorrect attempt) 

 It is sure that when the first Zero (EMPTY_CLUSTER) is found then the next positions until the end will be Zero If a new file is added to the filesystem, then the first free entry of the FAT will change from 0 to a new value It is possible that two consecutive positions of the FAT, have the FINAL_CLUSTER value All the information of the FAT can be inferred from the directory table

a)


 Next, two hints are offered to you. You can select a maximum of 1. You will not lose points for viewing hints.




-  This hint is about adding a file
-  This hint is about erasing files


For the simulated FAT of the lab (You will lose 3 points for each incorrect attempt) 

 It is sure that when the first Zero (EMPTY_CLUSTER) is found then the next positions until the end will be Zero If a new file is added to the filesystem, then the first free entry of the FAT will change from 0 to a new value It is possible that two consecutive positions of the FAT, have the FINAL_CLUSTER value All the information of the FAT can be inferred from the directory table

b)

 Next, two hints are offered to you. You can select a maximum of 1. You will not lose points for viewing hints.

-  This hint is about adding a file
-  When a file is erased from the simulated FAT, Which of these things happen? 
 - All the clusters where the file was located in the file system fs are set to 0 The corresponding records in the directory table are initialized All the FAT positions related to clusters where the file was located are set to 0

For the simulated FAT of the lab (You will lose 3 points for each incorrect attempt) 

 It is sure that when the first Zero (EMPTY_CLUSTER) is found then the next positions until the end will be Zero If a new file is added to the filesystem, then the first free entry of the FAT will change from 0 to a new value It is possible that two consecutive positions of the FAT, have the FINAL_CLUSTER value All the information of the FAT can be inferred from the directory table

c)

Fig. 5. Different phases of a user interaction with an exercise in our e-learning hinting system

The `dc:subject` property from the Dublin Core vocabulary is used, while the rest of properties (beginning with prefix `p`, which denotes our own namespace) are of our own creation. The `dc:subject` sets the different IPC concepts that the exercise covers; `p:mayberoot` at 1 indicates that the exercise can be a root exercise while `p:maybehint` at 0 indicates that the exercise cannot be a hint of another exercise; `p:difficulty` can take a value among the range [0,10] and it is the estimated difficulty of the exercise by a

teacher (the difficulty property of IEEE LOM standard was not taken because it has only a predefined value space). All of these properties are annotated at the beginning and they do not change during the process.

Fig. 5 shows a student interaction with the system in three different phases:

- Fig. 5, a: A student accesses a URL with a given exercise (root exercise) about the FAT file system (exercise *Pa*), that can try to solve pressing the “Check” button. The student can see that the maximum scoring is 10. In addition, the student is advised that each incorrect attempt will have a penalty of 3 points.
- Fig. 5, b: The student selects the “+” button to request for a hint and some meta-information is generated saying that one hint is about adding a file and another about erasing a file. In addition, the student is advised that she can only select 1 out of the 2 hints, but she will not have any penalties for it.
- Fig. 5, c: The student presses one of the “+” buttons to select one of the hints and then a new exercise (hint exercise) about erasing files is shown (exercise *Pb*). This new exercise is a hint with respect to the initial exercise.

Fig. 5 illustrates the interaction of a student with an exercise, while fig. 6 illustrates how each of the four users receives a different hint depending on the adaptation rule.

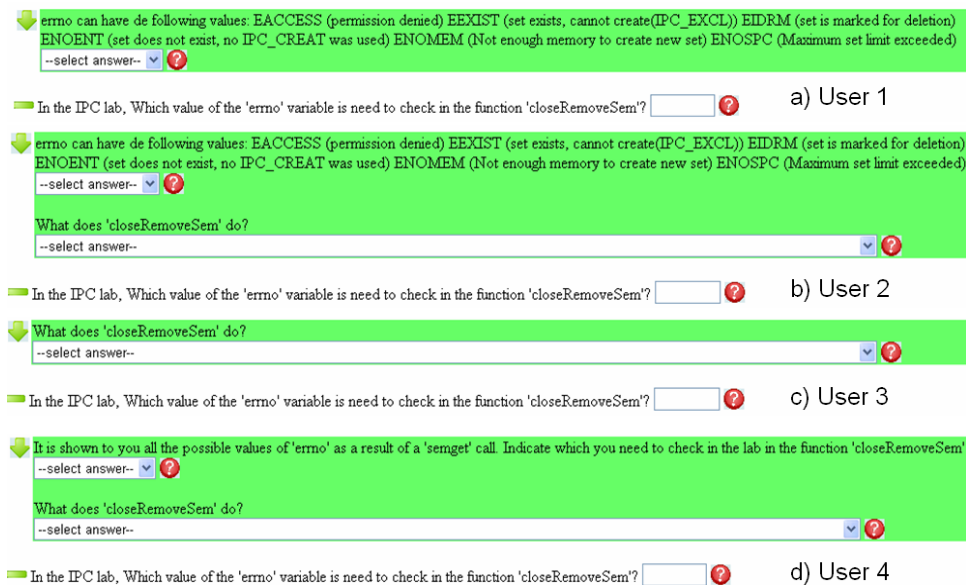


Fig. 6. Different hints depending on the student

4.2. Adaptation Aspects

Three high level aspects to adapt for the hinting computer based system are:

- Contents and types of hints (category R1). The adaptation of the hint contents has a strong pedagogical support, e.g. with cognitive educational theories. It is

recommended that a student receives a personalized content as help, in a way that the difficulty of that content should be according to their knowledge level, or a student should receive content according to the way she learns.

- Hinting strategies (category R2). The different hinting strategies to be selected are the following which define new sub-aspects: penalties for viewing hints or without penalties (R2.1); hints directly available or not (R2.2); rewards, penalties or no effect on the scoring for hint resolution (R2.3); and the limitation in the maximum number of hints to select (R2.4). Each of these four aspects cannot be divided more so they are adaptation variables (three of them categorical and one numerical). The reason under adapting the hinting techniques relies on increasing students learning gains in some situations as analyzed in [23].
- Penalties for each incorrect attempt (category R3). This aspect includes the adaptation of the number of penalty points for each incorrect attempt for a specific exercise. This aspect is a numerical adaptation variable. Reasons on using this feature are mainly two: on one hand to try to avoid the “try abuse” effect studied in the literature (a student that like to make a lot of attempts in an exercise even if she does not know the answer); on the other hand, an assessment system must evaluate in a fair way, and penalties should be applied to detect the exact knowledge.

There are four different sub-aspects to adapt for the contents category:

- Type of hint (R1.1). It is a leaf aspect, so an adaptation variable and is represented as a categorical variable with 2 possible values: sequence or group. The sequence type enables that the correct ordered structure of exercises can benefit the learning process. The group type enables a student to decide among different hint options according to active learning theories, based on some texts or meta-informations.
- Group parameters (R.1.2). Depends on the type of hint being a group, and is composed by 2 sub-aspects: text for meta-information and for the global text.
- Number of hints to show (R1.3). It is a leaf aspect and can be represented as an integer adaptation variable. The pedagogical reason under it is that depending on the student profile she can need more or less help in different concepts.
- Selection of hint problems (R1.4). Sets the specific exercise contents that will be selected as hints related to a specific root exercise. It is a categorical adaptation variable. There is a dependency of this adaptation variable with respect to R1.3, because it is necessary to know first the number of hint exercise to have. The system should select the hint exercises in a way to improve students’ learning process (e.g. according to their knowledge level in the different topics).

Next, an example atomic rule is represented textually out of the more than 20 atomic rules which have been designed for the different aspects. The rule code 1.4.7 represents that this rule belongs to the aspect of contents and types of hints (number 1), selection of hint problems (number 4), and finally the last number is to identify univocally the rule (number 7). The aspect 1.4 is a leaf node of the tree, and it can be represented as a categorical variable with all the possible options of hint problems. But finally, there is a final number to identify all the possible rules that can be applied to the aspect 1.4.

Code: R.1.4.7

Description: Select for each concept where a student requires hints, such hint exercises which difficulty level according to the system data of the students' interactions is closer to the student knowledge level in such concept

Parametric: Yes, the concept to be selected, and the number of required exercises, and all the selected parameters of the system to determine the problem difficulty

Linear combination: Yes, with R.1.4.6

Required sequence inputs: None

Possible sequence outputs: Any

4.3. Codified Adaptation Example with N3

A global adaptation rule example is shown in fig. 7 and combines several atomic rules. Each time that a user requests an exercise, this rule is applied. Our example rule in natural language is as follows, and tries to select the hints materials.

Example of Rule Definition: “A requested root exercise will have as hint a sequence of hint exercises. For each course concept covered by the requested initial root exercise in which the student has a lack (student's knowledge level less than a threshold in that concept), an initial exercise will be assigned as hint for such concept. This assigned exercise must be annotated such as that may be hint, the exercise must cover such concept but it must not cover other new concepts that are not present in the initial requested root exercise. In addition, the difference between the teacher's estimated difficulty of the exercise and the student's knowledge level in that concept must be the least between all the possible hint exercises that fulfill the other criterions. Finally, the resulted hint is a sequence of the combination of the different assigned hint exercises for each concept in which the student has a lack”

Next, there is an implementation in N3 of the different atomic rules involved in the larger rule of fig. 7. The output of the execution of each rule by the CWM reasoner is a file with information that can be used as input by other rules.

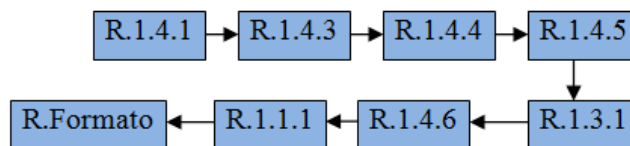


Fig. 7. Example rule composed by several atomic rules in sequence

Rule 1.4.1: Remove Exercises that Cannot be Hint Exercises. This rule discards the exercises that cannot be a hint (i.e. with the property *p:maybehint* to 0).

```
@forall :problem, :hint. {
  :problem p:possiblehint :hint .
  :hint p:maybehint 0 .
} log:implies {
  :problem p:discardedhint :hint .}
```

Rule 1.4.3: Remove Exercises that are the Same Root Exercise. This rule takes from all possible exercises that can be hints, these of them that are not the same as the initial exercises (because it does not make sense to have the same exercise as hint of itself). The rule is in the *R.1.4.3* file with the following N3 code:

```
@forall :problem, :hint. {
  :problem p:possiblehint :hint .
  :problem log:notEqualTo :hint .
} log:implies {
  :problem p:candidatehint :hint .}
```

Rule 1.4.4: Select Exercises that Cover some Concept that is also Covered in the Root Exercise. The rule is defined in the file *R.1.4.4.n3*:

```
@forall :problem, :concept1, :hint. {
  :problem p:possiblehint :hint .
  :concept1 a ipc:IPCConcept .
  :hint dc:subject :concept1 .
  :problem dc:subject :concept1;
} log:implies {
  :problem p:candidatehint :hint .}
```

If there is a hint that covers a specific concept (i.e. *:hint dc:subject :concept1*), which is also covered by the initial exercise (i.e. *:problem dc:subject :concept1*), then this hint exercise is a candidate to be a related hint of the initial exercise, based on concept similarity. These problems that are not according to this property, will not be hint candidates.

The CWM lines reasoned executions are omitted here, but one example was given in subsection 3.2. In this case, the annotations about the different problems should be included for the CWM reasoner, because we need to know which concepts will be covered by each problem. In addition, the annotations about the different IPC concepts should be included.

Rule 1.4.5: Remove as Hints such Exercises that Have some Concept that is not Covered in the Root Exercise. As a hint exercise can cover some concept (i.e. *:hint dc:subject :concept2*) which is also covered in the root exercise, but also some concept that is not covered in the root exercise (*:t log:notIncludes {:problem dc:subject :concept2}*), then it is possible to discard such types of exercises:

```
@forall :concept2, :t. {
  :problem p:possiblehint :hint .
  :concept2 a ipc:IPCConcept .
  :hint dc:subject :concept2 .
<file:ProblemsUnion.n3> log:semantics :t.
```

```

:t log:notIncludes {:problem dc:subject :concept2}.}
log:implies {
  :problem p:discardedhint :hint .}.

```

Rule 1.3.1: Selection of the Number of Hints Depending on the Users Needs. The following rule takes into account the students' knowledge level for each of the concepts covered in the root problem, in order to select the number of hint exercises. This rule sets the concepts in which the student has a knowledge level less than 8 (i.e. *:level math:lessThan "8"*), and these concepts are marked as concepts in which the student requires hints (i.e. *:student le:requireshint :concept*). The result of this rule is an enumeration of the concepts for which a user requires hints, instead of the typical candidate or discarded.

```

@forAll :problem, :hint. {
  :problem p:candidatehint :hint.} log:implies{
@forAll :student, :hint, :knowledge, :concept, :level.{
  :problem p:isinexecution 1 .
  :problem dc:subject :concept .
  :student papi:hasPerformance :knowledge.
  :knowledge papi:learning_competency :concept.
  :knowledge papi:performance_value :level.
  :level math:lessThan "8" .} log:implies{
  :student le:requireshint :concept. }.

```

Rule 1.4.6: Select Hints with a Similar Difficulty Level to the Students Knowledge Level. This rule discards hints which estimated difficulty level is not closer to the students' knowledge level on that concept. In the last part of the rule, the absolute value difference between the student's knowledge level and the teacher's estimated difficulty of a possible hint exercise is calculated for each concept. Only the hint exercises with the lowest difference for each concept are not marked as discarded. The described rule (R.1.4.6.n3) in N3 is the following:

```

@forAll :student, :problem, :hint, :concept. {
  :problem p:possiblehint :hint.
  :student a le:Learner.
  :hint dc:subject :concept.} log:implies{ @forAll :f. {
  <file:conclusions6.n3> log:semantics :f.
  :f log:notIncludes {:student le:requireshint :concept
} .} log:implies {
  :problem p:discardedhint :hint . }.}.
@forAll :student, :problem, :concept, :hint1, :hint2,
:knowledge, :level, :dif1, :dif2, :a1, :a2, :b1, :b2. {
  :student a le:Learner; le:requireshint :concept.
  :problem p:possiblehint :hint1; :hint2.
  :hint1 dc:subject :concept; p:difficulty :dif1.
  :hint2 dc:subject :concept; p:difficulty :dif2.
  :student papi:hasPerformance :knowledge.
  :knowledge papi:learning_competency :concept.
  :knowledge papi:performance_value :level.
  (:level :dif1) math:difference :a1.
  (:level :dif2) math:difference :a2.

```

```
:a1 math:absoluteValue :b1.  
:a2 math:absoluteValue :b2.  
:b2 math:lessThan :b1.)  
log:implies { :problem p:discardedhint :hint1.}).
```

Rule 1.1.1: Selection of a Hint as of Sequence Type. For this rule, there is no need to make any reasoning, because implies the automatic selection of the type of hint to use, that will be a sequence one.

4.4. Generalization to other Adaptive E-learning systems

Our rule model cannot only be applied for hinting adaptive systems but for other e-learning systems with other functionality and aspects to adapt. We have done an analysis of adaptive e-learning systems that follow the IMS-QTI (Question and Test Interoperability) specification and the IMS-LD (Learning Design) specification and we have concluded that our rule model can be applied to any system that follow IMS-QTI and a wide spectrum covering IMS-LD. This paper does not include all the details of these analyses because of space constraints but give some indications.

For example, the IMS-QTI and IMS-LD specifications enables the use of an undefined number of nested *if/else* structures to define different adaptation conditions. The output variables of an *if/else* might not include the specific aspects for adaptation, but they might be e.g. intermediary variable calculations. In that case, this does not suit as a rule in our model because some change of the aspect to adapt should be included. But if that single *if/else* is combined with other *if/else* that might involve a change on the variable to adapt, then this can suit to our definition of rule. In addition, a single *if/else* can change different outcome variables (that would be adaptation variables in our model), but according to our model an atomic rule must only focus on one adaptation variable. Therefore, one atomic rule of our model might be composed by several single *if/else*, and one single *if/else* might result in several atomic rules in our model, but also a single *if/else* might be an atomic rule. An atomic rule is the minimum unit that enables some filter of options for a specific adaptation variable, while a single *if/else* can change different adaptation variables or might not involve the change for the adaptation variable but being an intermediary step. More advantages from our model can be obtained if there are a lot of atomic parts in which the *if/else* constructions can be divided. But in any case, it is always possible to map an *if/else* structure to our model (the worst case is that all the *if/else* constructions would be mapped into only one rule for each of the adaptation variables).

This analysis reveals that the AR model cannot only be applied to the hinting adaptive e-learning system, but it can also be applied to other very generic e-learning systems that follow the IMS-QTI and IMS-LD specifications. The theoretical analysis reveals the possibility of its application and the definition of the rules would follow similar patterns as in the case study. Doing a similar theoretical analysis, the AR model could be applied to other adaptive non e-learning systems, such as those whose use similar *if/else* constructions for the adaptation of non-learning resources.

4.5. Evaluation with teachers and learning analytics experts

In order to check several of our hypotheses about the advantages of our AR model for the e-learning area, we made an evaluation with teachers and learning analytics experts. In this work, we consider learning analytics experts as people who is used to interpret, analyze and make proper decisions about the learning process. A total of 30 people participated in a survey, being 15 teachers, and 15 learning analytics experts. Participants were explained the AR model in general and its application to the specific hinting adaptive e-learning system in particular for about 30 minutes. Next, participants were asked to rate a total of eight questions in a likert scale (1. Strongly disagree, 2.- Disagree, 3.- Neutral, 4.- Agree, 5.- Strongly agree), which results are shown in table 1.

Table. Survey results about the AR model for adaptive e-learning

	1	2	3	4	5	Mean	Conf. interval, 95%
1.- I would be able to understand the adaptive functionality of a learning system based on a description of the AR model, including its graphs as well as their textual description of rules	0	0	2	14	14	4.40	[4.17, 4.63]
2.- The AR model offers a clear vision of the atomic rules to apply and how the rules can be combined	0	0	3	16	11	4.27	[4.03, 4.51]
3.- I would be able to make the graphs and the textual descriptions of the rules following the AR model in order to describe the learning adaptation of a system	0	0	4	16	10	4.20	[3.95, 4.45]
4.- The application of the AR model helps me in determining and considering all the learning adaptation options that I wish to implement	0	0	10	10	10	4	[3.69, 4.31]
5.- The application of the AR model helps me to work in a collaborative and distributed way, so that different descriptions of models can be interchanged among several team members	0	0	6	13	11	4.17	[3.89, 4.45]
6.- The AR model helps me to reuse adaptation rules defined in other contexts by other rule designers in similar contexts	0	0	4	9	17	4.43	[4.16, 4.71]
7.- The AR model let improve the rule maintenance in an adaptive system	0	0	9	11	10	4.03	[3.73, 4.34]
8.- The AR model implies a set of advantages with respect to your traditional system of codifying rules in an adaptive system	0	0	7	13	10	4.10	[3.82, 4.38]

Results show that teachers and learning analytics experts think that the authoring process is easy with our proposed AR model: for the understanding of the descriptions (questions 1 and 2) as well as for the own creation of the graphs and texts of the model (question 3). In addition, teachers and learning analytics experts think that our proposed AR model helps for completeness (question 4), collaborative authoring (question 5), rule reuse (question 6) and maintenance (question 7). Confidence intervals at 95% are provided for each one of the survey questions in table 1, as well as the mean. Looking at the worst case of the low value of the limit of the confidence intervals for each question (to discard conclusions by chance), we can see that the evaluation of each aspect/question of the model is positive even at the worst case.

In addition, most participants agree that the model implies a set of advantages with respect to their traditional way of codifying rules (question 8). Participants were all questioned about the method they use for codifying the rules and all of them answered either they do not follow any software methodology for the codification of rules or they use some UML diagram like flow diagrams.

The provided evaluation with teachers and learning analytics experts is restricted to e-learning systems, which is a very important research area. The evaluation of the model in other applications (e.g. recommendation of services or customization of portals) with experts related to these other applications would be a nice future work direction. In any case, this provided evaluation can give useful insights for some specific issues of other areas due to its similarities. For example, the recommendation of services might be based on the cultural level (which might be analogous to a student's grade), the user preferences (which might be similar to the student's profile) or the user previous actions on accesses to similar pages (which might be similar to students' interactions with educational contents).

This work has evaluated the AR model with teachers and learning analytics experts since the model is focused on the authoring process and these are the key stakeholders for the creation and design of the rules. The students have not been involved in the evaluation because the main focus of the paper is on validating the AR authoring model but not to evaluate the generated system itself or the quality of the rules. We assume that rule designers and creators already know whether these proposed adaptation rules are good and the rules they want to propose.

However, the students or users of the system might give some feedback about the adaptation rules that can imply a redesign of them. In addition, real experiences with students might give conclusions about which adaptation rules are more suitable e.g. for improving students' learning gains. But this would be an interesting future work with a different focus of the research based on which are the semantics of the best rules instead of how to improve the process of creation of rules once the designers know which are these rules. Indeed, in previous work [23], we analyze the real data of the hinting system with students in different experiences in order to extract conclusions about the type of adaptation rules that might be useful and might work well for students, taking into account their learning gains.

Another interesting future work involving evaluation with students would be to test the maintainability of the defined rule model. Experiences with real students might bring redefinition of the rules during the time based on students' feedback and students' results with the system. Monitoring of creators and designers during this process would

be able to give insights about the maintainability of the authoring process and compare it with other methodologies.

Finally, it is interesting to note that evaluation of authoring models with experts and teachers is a widespread technique in e-learning. In this direction, evaluation of patterns for the authoring of learning design scripts have been evaluated with three teachers [51] or a tool for making better use of the European qualifications framework (EFQ) has been evaluated with 20 teachers [52]. In this case, the evaluation is about the authoring of adaptation rules.

5. Relationship of our Approach with other Works

The presented adaptive hypermedia systems in the related work neither do not focus on the rules design or they do not implement any software engineering methodologies for the description of such rules. On the contrary, our work shows a new methodology of software engineering for the description and modeling of rules in a service oriented environment, but some of the commented works presented little steps in the direction of our model. The philosophy of first selecting all the possible candidate resources was also introduced in [8], the distinction among different independent aspects of adaptation was done e.g. in [7] and [8], or the semantics of a particular combination of rules to join different decisions is provided in [47].

From the software and web engineering area, we have already presented different approaches (in the related work section) which can target applications for adaptation. These other software and web engineering approaches do not go into deep details about the rules: their atomicity, reuse, manners of combination in a service oriented environment, restrictions, distributed use, etc. but they are more generic models in the sense that they can describe a complete web application but not just adaptation rules.

Our presented model follows a component based approach and may use a Service Oriented Architecture (SOA), in which each rule can be invoked from outside. Our solution is not binding to any SOA solution and can be implemented with different SOA approaches that provide service registry, lookup or security. There are other proposed service oriented architectures for e-learning that are not focused on the invocation of rules such as OKI [50] or KnowledgeTree [24]. OKI defines a set of common educational services and applications, and for each service defines a collection of web services to implement it. These web services can be invoked from other external sources in a distributed way. An educational system can be built using invocations to these different methods. For example, the assessment is one educational service, and one web service method is for the creation of an item. But OKI do not define any models for the description of rules, and the semantics of the web services are different to our model.

Finally, we follow a component-based architecture, as the atomic rules can be seen as independent components that can form bigger ones. As commented in [53], “finding appropriate formal approaches for describing components, and the methods for component-based software construction, is correspondingly challenging”.

6. Conclusion

This work presents a new model for the description of rules for adaptive software applications. The model establishes a set of rule properties (e.g. atomicity, compositionality according to 4 defined combinations or restrictions of these combinations) which imply a set of advantages such as reusability, interoperability or improvements in the authoring process. Some of these advantages are the same as from traditional software engineering but made stronger with the consideration of a service oriented Internet in which rules can be composed from different Internet sources according to the rule restrictions combinations.

Some of these advantages have not only been analyzed in an analytic form, but they have also been confirmed for the adaptive e-learning area, with a survey to teachers and learning analytics experts. In addition, this survey also confirmed that our model also implies a set of advantages with respect to the traditional approach of codifying rules without any methodologies or with other traditional methodologies like UML flow diagrams. The proposal is fully illustrated with a real application we have developed in the context of the e-learning area using semantic web technologies. Specific implementations of the rules are presented with the N3 semantic web language for the adaptation of hints. The provided case study presents an evaluation of the feasibility of the proposed model for a typical type of e-learning system that generates adaptive hints. Many other software and web applications can also benefit from this model, such as e-learning systems which follow the IMS-QTI or IMS-LD specifications.

The case study use adaptation rules that can be divided into many atomic parts. The most we can divide into adaptation parts, the more benefits the model presents.

As a future vision, the real implementation of different atomic rules might be in different repositories and could be accessible by anyone to form larger rules, with discovery or searching services. For each different software aspect, a list of complete atomic rules might be defined. Each software application would take the ones that would be required for their purposes.

Acknowledgments. This work has been supported by the “GEEWHEZ: ManaGEMEnt and leisure middleWare for tHemE parks and Zoo” project (European Commission under grant FP7-SME-2011-286533), the EEE project (Spanish Ministry of Science and Innovation, “Plan Nacional de I+D+I TIN2011-28308-C03-1), the Remediss project (Spanish Ministry of Economy and Competitiveness under grant IPT-2012-0882-430000) and the eMadrid project (Regional Government of Madrid under grants S2009/TIC-1650 and S2013/ICE-2715).

References

1. Park, W., Kang, S., Kim, Y.: A Personalized Multimedia Contents Recommendation Using a Psychological Model. *Computer Science and Information Systems*, Vol. 9, No. 1, 1-21. (2012)
2. Muñoz-Organero, M., Muñoz-Merino, P.J., Delgado Kloos C.: Personalized Service-Oriented E-Learning Environments. *IEEE Internet Computing*, Vol. 14, No. 2, 62-67 (2010)

3. Dagger D., O'Connor, A., Lawless, S., Walsh, E., Wade, V.P: Service-Oriented ELearning Platforms: From Monolithic Systems to Flexible Services. *IEEE Internet Computing*, Vol. 11, No. 3, 28-35 (2007)
4. Sterling L, Juan T.: The software engineering of agent-based intelligent adaptive systems. In *Proceedings of the 25th International Conference on Software engineering*, 704-705 (2005)
5. He, J., Gao, T., Hao, W., Yen, I., Bastani, F.: A Flexible Content Adaptation System Using a Rule-Based Approach. *IEEE Trans. on Knowledge and Data Eng.* Vol. 19, No. 1 (2007)
6. Vassileva D, Bontchev B. Adaptation engine construction based on formal rules. In *Proceedings of the 1st Int. Conference on Computer Supported Education*, 327-332 (2009)
7. Aroyo, L., Dicheva, D.: The New Challenges for E-learning: The Educational Semantic Web. *Educational Technology and Society* Vol. 7, No. 4, 59-69 (2004)
8. Karampiperis, P., Sampson, D.: Adaptive Learning Resources Sequencing in Educational Hypermedia Systems. *Educational Technology & Society* Vol. 8, No. 4, 128-147 (2005)
9. Hehner, E.C.: *The Logic of Programming*. Prentice-Hall, Inc. (1984)
10. Jones, C.B.: *Systematic Software Development using VDM*. Prentice-Hall (1990)
11. Maciaszek, L.A.: Modeling and engineering adaptive complex systems. In *Proceedings of the 26th International Conference on Conceptual modeling*, 31-38 (2007)
12. de Sousa Saraiva, J., Rodrigues da Silva, A.: Development of CMS-Based Web-Applications Using a Model-Driven Approach. In *Proceedings of the 4th International Conference on Software Engineering Advances*, 500-505 (2009)
13. Demirkol, S., Challenger, M., Getir, S., Kosar, T., Kardas, G., Mernik, M.: A DSL for the Development of Software Agents working within a Semantic Web Environment. *Computer Science and Information Systems*, Vol. 10, No. 4, 1525-1556. (2013)
14. Wang, Q.: Towards a rule model for self-adaptive software. *ACM SIGSOFT Software Engineering Notes* Vol. 30, No. 1, 1-5 (2005)
15. Ceri, S., Fraternali, P., Bongio, A., Brambilla, M., Comai, S., Matera, M.: *Designing Data-Intensive Web Applications*. Morgan Kaufmann, Elsevier (2003)
16. Troyer, O.D.: Audience-driven web design. In M. Rossi, & K. Siau (Eds.) *Information Modeling in the New Millennium*, 463-484, Hershey, PA: Idea Group Publishing (2001)
17. Schwabe, D., Pontes, R., Moura, I.: OOHDWeb: An Environment for Implementation of Hypermedia Applications in the WWW. *ACM SigWEB Newsletter*, Vol. 8, 18-34 (1999)
18. Hennicker, R., Koch, N.: A UML-based Methodology for Hypermedia Design. In: Evans, A., Kent, S., Selic, B. (Eds.) *The Unified Modeling Language. Lecture Notes in Computer Science*, Vol. 1939. Springer-Verlag, Berlin Heidelberg New York, 410-424 (2001)
19. Gomez, J., Cachero, C.: OO-H Method: extending UML to model web interfaces. *Information modeling for internet applications*, 144-173 (2003)
20. De Bra, P.: Adaptive educational hypermedia on the Web. *Communications of the ACM*, Vol. 45, No. 5, 60-61 (2002)
21. Brusilovsky P. Methods and techniques of adaptive hypermedia. In *Proceedings of the Adaptive Hypertext and Hypermedia Conference*, 87-129 (1996)
22. Shutte, V., Towle, B.: Adaptive E-Learning. Ed. *Psychology*. Vol. 38, No. 2, 105-114 (2003)
23. Muñoz-Merino, P.J., Delgado Kloos, C., Muñoz-Organero, M.: Deciding on Different Hinting Techniques in Assessments for Intelligent Tutoring Systems. *International Journal of Innovative Computing, Information and Control*. Vol. 7, No. 2, 841-858 (2011)
24. Brusilovsky, P.: KnowledgeTree: A distributed architecture for adaptive e-learning. In *Proceedings of the International World Wide Web conference*, 104-111 (2004)
25. Anderson, J.R., Conrad, F.G., Corbett, A.T.: Skill Acquisition and the LISP Tutor. *Cognitive Science* Vol. 13, No. 4, 467-505 (1989)
26. Harskamp, E., Ding, N.: Structured collaboration versus individual learning in solving physics problems. *Int. Journal of Science Education*, Vol. 28, No. 14, 1669-1688 (2006)
27. Gertner, A., VanLehn, K.: Andes: A Coached Problem Solving Environment for Physics. In Gauthier, Frasson, VanLehn (Eds.), *Intelligent Tutoring Systems: 5th International*

- Conference, Lecture Notes in Computer Science, Vol. 1839, Springer-Verlag Berlin & Heidelberg GmbH & Co. K, 133-142 (2000)
28. Hough, M., Marlin, T.: Web-based interactive learning modules for process control. *Computers and Chemical Engineering*, Vol. 24, No. 2, 1485-1490 (2000)
 29. Gertner, A.S., Conati, C., Vanlehn K.: Procedural help in Andes: Generating hints using a Bayesian network student model. In *Proceedings of the 15th National Conference on Artificial Intelligence*, 106–111. (1998)
 30. Suebnukarn, S., Haddawy, P.: A Bayesian approach to generating tutorial hints in a collaborative medical problem-based learning system. *Artif Intell Med*, Vol. 38, 5-24. (2006)
 31. Manouselis N., Sampson D. A Multi-criteria Model to Support Automatic Recommendation of e-Learning Quality Approaches. In *Proceedings of the 16 World Conference on Educational Multimedia, Hypermedia and Telecommunications*, 518-526 (2004)
 32. Boyce, S., Pahl, C.: Developing Domain Ontologies for Course Content. *Journal of Education Technology and Society*, Vol. 10, 275-288. (2007)
 33. Gasevic D, Hatala M. Ontology mappings to improve learning resource search. *British Journal of Educational Technology*, Vol. 37, No. 3, 375–389. (2006)
 34. Dicheva D, Sosnovsky S, Gavrilova T, Brusilovsky P. Ontological Web Portal for Educational Ontologies. In *Proceedings of the International Workshop on Applications of Semantic Web technologies for E-Learning*. (2005)
 35. Dublin Core Metadata Initiative. [Online]. Available: <http://dublincore.org/> [current December 2013]
 36. IEEE LOM, "Learning Object Metadata". [Online]. Available: <http://ltsc.ieee.org/wg12/> [current December 2013]
 37. IEEE PAPI, IEEE P1484.2/D7 (2000)
 38. Aroyo, L., Pokraev, S., Brussee, R.: Preparing SCORM for the Semantic Web. In: Meersman, R., Tari, Z, Smidch, D.C. (eds.): *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*. Lecture Notes in Computer Science, Vol. 2888. Springer-Verlag, Berlin Heidelberg New York, 621-634. (2003)
 39. Psyche ,V., Bourdeau, J., Nkambou, R., Mizoguchi, R.: Making Learning Design Standards Work with an Ontology of Educational Theories. In *Proceedings of the Artificial Intelligence in Education conference 125: 539-546* (2005)
 40. Lama M, Vila X, Amorim R, Sánchez E. Semantic description of the IMS Learning design specification. In *Proceedings of the Applications of Semantic Web Technologies for E-learning*. 37-46 (2005)
 41. Devedzic, V.: *Semantic Web and Education*. Springer (2006)
 42. Mizoguchi, R., Bourdeau, J.: Using Ontological Engineering to Overcome Common AI-ED Problems. *Journal of Artificial Intelligence and Education*, Vol. 11, 107-121 (2000)
 43. Henze, N., Dolog, P., Nejd, W.: Reasoning and Ontologies for Personalized E-learning in the Semantic Web. *Journal of Educational Technology and Society*, Vol. 7, 82-97 (2004)
 44. Cheniti-Belcadhi, L., Henze, N., Braham, R.: An Assessment Framework for eLearning in the Semantic Web. In *Proc. of the Conf. on Adapt. and User Model. in inter. Systems* (2004)
 45. Jovanovic, J., Gasevic, D., Brooks, C., Eap, T., Devedzic, V., Hatala, Richards, G.: Leveraging the Semantic Web for Providing Educational Feedback. In *Proceedings of the IEEE International Conference on Advanced Learning Technologies*, 551-555 (2007)
 46. Liu, H.: QoL Guaranteed Adaptation and Personalization in E-learning Systems. *IEEE Transactions on Education*, Vol. 48, No. 4, 676-687. (2005)
 47. de Virgilio, R, Torlone, R., Houben, G. A.: Rule Based Approach to Content Delivery Adaptation in Web Based Information Systems. In *Proceedings of the International Conference on Mobile Data Management*, 21-25 (2006)
 48. Muñoz Merino, P.J., Delgado Kloos, C.: An Architecture for Combining Semantic Web Techniques with Intelligent Tutoring Systems. In: Woolf, B.P., Aimeor, E., NKambou, R,

- Lajoie, S. (eds.): *Intelligent Tutoring Systems. Lecture Notes in Computer Science*, Vol. 5091. Springer-Verlag, Berlin Heidelberg New York, 540-550. (2008)
49. Muñoz-Merino, P.J., Delgado Kloos C.: A software player for providing hints in problem-based learning according to a new specification. *Computer Applications in Engineering Education*, Vol. 17, No. 3, 272-284. (2009)
 50. OKI (Open Knowledge Initiative), [Online]. Available: http://sourceforge.net/projects/okiproject/?_test=beta [current December 2013]
 51. Hernández-Leo, D, Villasclaras-Fernández, E. D., Asensio-Pérez, J. I, Dimitriadis, Y., Jorrín-Abellán, I. M., Ruiz-Requies, I., & Rubia-Avi, B. (2006). COLLAGE: A collaborative Learning Design editor based on patterns. *Educational Technology & Society*, 9 (1), 58-71.
 52. Florian-Gaviria, B. Glahn, C., & Fabregat Gesa, R. (2013). A Software Suite for Efficient Use of the European Qualifications Framework in Online and Blended Courses. *IEEE Transactions on Learning Technologies*, 6 (3), 283—296.
 53. Jifeng, H., Li, X., Liu, Z.: Component-based software engineering – the need to link methods and their theories. In *Proceedings of the International Colloquium on Theoretical Aspects of Computing*, 72—97 (2005)

Pedro J. Muñoz-Merino is a Visitant Associate Professor at the Universidad Carlos III de Madrid. In 2003, he received his Telecommunication Engineering degree from the Polytechnic University of Valencia, and in 2009 his PhD in Telematics Engineering from the Universidad Carlos III de Madrid. He obtained his accreditation in May 2012 as Associate Professor by the ANECA agency from the Spanish Ministry of Education. He has also received several awards like the AMPER award for the best Master Thesis in Telematics Engineering or a special PhD award by Universidad Carlos III de Madrid. He has made two long research stays: one in Ireland for more than 3 months at the Intel company in 2005, and another in Germany for more than 6 months at the Fraunhofer Institute of Technology in 2009-2010. In addition, he is author of more than 60 scientific publications and has participated in more than 30 research projects.

Carlos Delgado Kloos received the Ph.D. degree in Computer Science from the Technical University of Munich and in Telecommunications Engineering from the Technical University of Madrid. He is Full Professor of Telematics Engineering at the Universidad Carlos III de Madrid, where he is the director of the UNESCO Chair on “Scalable Digital Education for All” and of the GAST research group and coordinator of the eMadrid network about e-learning in the Region of Madrid. He is also Vice-Rector of Infrastructures and Environment.

Mario Muñoz-Organero received the M.Sc. degree in telecommunications engineering from the Polytechnic University of Catalonia, Barcelona, Spain, in 1996, and the Ph.D. degree in telecommunications engineering from the Universidad Carlos III de Madrid, Leganes, Spain, in 2004. He is a Professor of telematics engineering at the Universidad Carlos III de Madrid. He has also more than four years of experience working for the telecommunications industry in companies such as Telefonica R&D and Lucent Technologies, both in Madrid, Spain. He has participated in several European-funded projects such as E-LANE and Spanish-funded projects such as MOSAIC learning, Learn3 and OSAMI. He is currently the lead researcher of the Spanish-funded HERMES, ARTEMISA, HAUS, COMINN, REMEDISS and IRENE projects. He was also the IP of the GEEWHEZ EU-FP7 project and is currently an MC member of TU1305 EU funded COST action about social networks and travel behavior.

Abelardo Pardo is Associate Head of Teaching and Learning and Lecturer at the School of Electrical and Information Engineering, The University of Sydney. He has a PhD in Computer Science by the University of Colorado at Boulder applied to formal verification of digital circuits. He is co-director of the Software Engineering group and his research interest is in the application of technology to explore, understand and influence human behavior. He has experience in the use of technology in learning and behavioral analytics, social networks, computer supported collaboration, personalization, and technology enhanced learning. He has participated in national and international projects funded by the Office for Teaching and Learning (Australia), National Science Foundation (USA), and the European Union. He serves as member of the steering committee of the Society for Learning Analytics Research (www.solaresearch.org), and member of the editorial board of the IEEE Transactions on Learning Technology and the Journal of Social Media and Interactive Learning Environments.

Received: January 03, 2014; Accepted: October 10, 2014

