

# Modelo de Autoria de Cursos de Ensino a Distância

---

Gláucia Regina M. A. Sizilio<sup>1</sup>

Nina Edelweiss<sup>2</sup>

---

**Resumo** - O Ensino a Distância (EAD) é caracterizado não só pela separação entre professor e aluno, como também pela comunicação mediada por alguma forma de tecnologia. Identificando a world wide web como o instrumento com maior condição de promover a bilateralidade nessa comunicação (através de aplicações educacionais interativas) e o workflow como uma alternativa adequada à modelagem de cursos de EAD (permitindo a definição de um curso, com a clara especificação de todas as atividades a serem executadas, dos relacionamentos entre as mesmas e dos agentes responsáveis pela sua execução), este trabalho apresenta um método para a modelagem de cursos de EAD e o workflow completo da autoria de cursos.

---

**Abstract** - Not only the physical distance of the student to the teacher characterizes Long Distance Learning (LDL), but also the need of any kind of communication between them. This communication must be provided by some kind of technology. Nowadays the world wide web appears as the best tool to assure this communication, through educational interactive applications. Workflow Modeling has proven to be an appropriate tool to model LDL courses, representing, specifying clearly all the activities to be executed, their relationships and the agents responsible for their execution. This paper presents a method to be used in LDL courses modeling. The complete workflow to be followed with the aim of creating a new course (authoring workflow) is also presented.

---

**Palavras-chave** - Ensino a distância, ensino a distância via world wide web, workflow, modelagem de workflow, workflow de autoria para ensino a distância.

## 1 INTRODUÇÃO

Um *workflow* (Casati et al., 1995; Barthelmes et al., 1995; Workflow Management Coalition, 1996) define as tarefas e atividades a serem desenvolvidas para a execução de um trabalho, sua coordenação e os agentes responsáveis por cada uma delas. A coordenação entre as tarefas/atividades inclui a possibilidade de estas serem executadas sequencialmente ou em paralelo, além de permitir o controle de disparo de cada uma delas.

---

<sup>1</sup> Prof<sup>ª</sup> Msc. Eng<sup>ª</sup> Gláucia Regina Medeiros Azambuja Sizilio - Universidade para o Desenvolvimento do Estado e da Região do Pantanal Núcleo de Educação a Distância/UNIDERP; Sizilio@terra.com.br

<sup>2</sup> Prof<sup>ª</sup> Dra. Eng<sup>ª</sup> Nina Edelweiss - Universidade Federal do Rio Grande do Sul - Instituto de Informática/UFRGS; nina@inf.ufrgs.br

As técnicas de modelagem de *workflow*, amplamente utilizadas para a modelagem de negócios e de processos, têm-se apresentado como possibilidade para modelar as atividades inerentes aos cursos de ensino a distância, uma vez que permitem a “definição de todo um processo, com a especificação clara de todas as atividades a serem executadas, de seus relacionamentos, e incluindo os agentes responsáveis pela sua execução, é conhecida como *workflow*” (Nicolao et al., 1998).

Oliveira et al. (1998) propõem um modelo de curso para tele-ensino, utilizando o TF-ORM (modelo de dados temporal orientado a objetos), baseado na modelagem conceitual do curso como um *workflow* e em agentes inteligentes que seguem a especificação e monitoram cada um dos estudantes.

Como resultado de pesquisa realizada sobre técnicas de modelagem de *workflow* aplicadas à autoria de execução de cursos de ensino a distância (Sizilio, 2000), além de uma proposta de estrutura para cursos de EAD, desenvolveu-se a modelagem completa tanto da autoria quanto da execução de cursos nesta modalidade de ensino.

O *workflow* (WF) de autoria tem por objetivo apoiar o professor na criação de cursos de EAD, englobando todas as tarefas relacionadas à criação de um curso (concepção, planejamento, definição e implementação); o de execução engloba as tarefas operacionais inerentes ao curso (todas as tarefas relacionadas à aplicação do curso em uma ou mais alunos).

Neste artigo será detalhado somente o WF de autoria, considerando-se a importância que ele tem no auxílio à elaboração de cursos. Norteado pelo *workflow* aqui apresentado, o autor de um curso tem a indicação de quais tarefas/atividades podem ser executadas a cada momento, de quais as ferramentas instrucionais disponíveis em cada uma das fases, e de como podem ser inseridas no contexto, definindo-se, portanto, um “caminho” claro e completo para a autoria de um curso.

A seção 2 deste trabalho apresenta o método utilizado para representar o WF, cujo conhecimento será necessário à compreensão do mesmo. O WF propriamente dito é apresentado na seção 3. Um exemplo de sua aplicação é mostrado na seção 4, além de avaliação da utilização do *workflow* de autoria. Algumas

considerações conclusivas, sinalizando inclusive trabalhos futuros nesta linha, são apresentadas na seção 5.

## 2 MÉTODO DE MODELAGEM UTILIZADO

Esta seção apresenta o método de modelagem utilizado neste artigo para a representação do WF de autoria. O entendimento deste método é fundamental para possibilitar a sua utilização.

O WF foi expresso através de um modelo de representação de WF proposto por Casati et al. (1995), estendido por Sizilio (2000) para atender às especificidades da modelagem de cursos de EAD. Foram realizadas algumas alterações com relação, principalmente, à especificação das tarefas. Para tal foi desenvolvido um *template* utilizando-se itens definidos no modelo original, algumas características inerentes ao modelo de gatilhos proposto por Stef Joosten (1994, 1995) e alguns itens utilizados pelo método CIMOSA para modelagem de negócios (Zelm, 1995).

Os principais aspectos que influenciaram na escolha deste modelo foram (Sizilio, 2000):

- ◆ a capacidade de o modelo representar qualquer tipo de WF, sendo considerado um dos mais completos modelos para especificação de WF;
- ◆ a especificação de tarefas permitir a inclusão de acesso a banco de dados externos;
- ◆ o modelo apresentar facilidades para a modularização de tarefas;
- ◆ o formalismo requerido pelo modelo permitir a especificação clara e completa das tarefas;
- ◆ o modelo prever a representação do tratamento de exceções.

O WF é definido através de uma representação gráfica, cujos símbolos utilizados, com seus significados e a explicação dos *templates* que os complementam, são apresentados a seguir.

## 2.1 REPRESENTAÇÃO GRÁFICA DO WF

### 2.1.1 TAREFA

Uma **tarefa** é uma unidade elementar de trabalho entregue para execução por um ator. É representada através de um retângulo, contendo o nome da tarefa e um identificador que a relaciona ao *template* que complementa sua descrição (Figura 1-a).

Nessa representação, [**Tarefa: Tn.n**] é o identificador da tarefa, em que **T** significa Tarefa, o primeiro **n** refere-se à fase relativa ao curso de EAD sendo desenvolvido (sendo n=1 para autoria e n=2 para execução) e o segundo **n** refere-se ao número aleatório de identificação da tarefa na fase (Ex.: [**Tarefa: T1.19**] significa a tarefa 19 da fase autoria).

### 2.1.2 SUPERTAREFA

O agrupamento de várias tarefas é denominado de **supertarefa**. A definição de supertarefas permite a modularização das tarefas, visando diminuir a complexidade do WF. São representadas através de um retângulo sombreado, contendo um nome e a indicação de qual figura (constante da lista de figuras do texto ou do conjunto de figuras com a modelagem do WF) tem o seu detalhamento (Figura 1-b). Nessa representação da supertarefa, [**Figura n**] é a indicação de qual figura tem o detalhamento da supertarefa.

### 2.1.3 MULTITAREFA

As **multitarefas** servem para definir um conjunto de tarefas que realizam as mesmas operações em paralelo. Cada multitarefa possui um componente (tarefa ou supertarefa) do qual são geradas múltiplas instâncias para execução, devendo ser definidos os valores *j* (número de instâncias geradas) e *quorum* (número de instâncias que precisam ser concluídas para que se encerre a multitarefa). A Figura 1-c apresenta o símbolo utilizado para representar uma multitarefa, e seu significado. Uma supertarefa pode esta mesclada com uma multitarefa, ficando a representação gráfica como segue (Figura 1-d):

As **conexões entre tarefas** estipulam os tipos de roteamento possíveis entre duas tarefas A e B a serem executadas. O modelo implementa um conjunto dessas construções que são **causalidade**, *forks* e *joins*.

Na construção **causalidade**, o término da tarefa A habilita a execução da tarefa B, que é sua sucessora (Figura 2).

### 2.1.4 CONEXÕES ENTRE TAREFAS

As **conexões entre tarefas** estipulam os tipos de roteamento possíveis entre duas tarefas A e B a serem executadas. O modelo implementa um conjunto dessas construções que são **causalidade**, *forks* e *joins*.

Na construção **causalidade**, o término da tarefa A habilita a execução da tarefa B, que é sua sucessora (Figura 2).

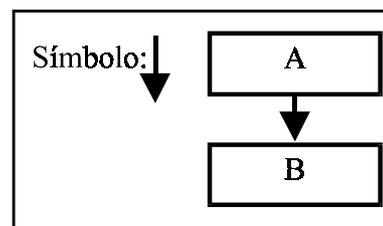


Figura 2 - Causalidade – Conexão direta entre duas tarefas

Nos **forks**, uma tarefa A é seguida por um conjunto de outras tarefas, denominadas sucessoras. Essa relação pode ser de quatro formas, ou seja, Total, Não-determinístico, Condicional e Condicional com Exclusão Mútua.

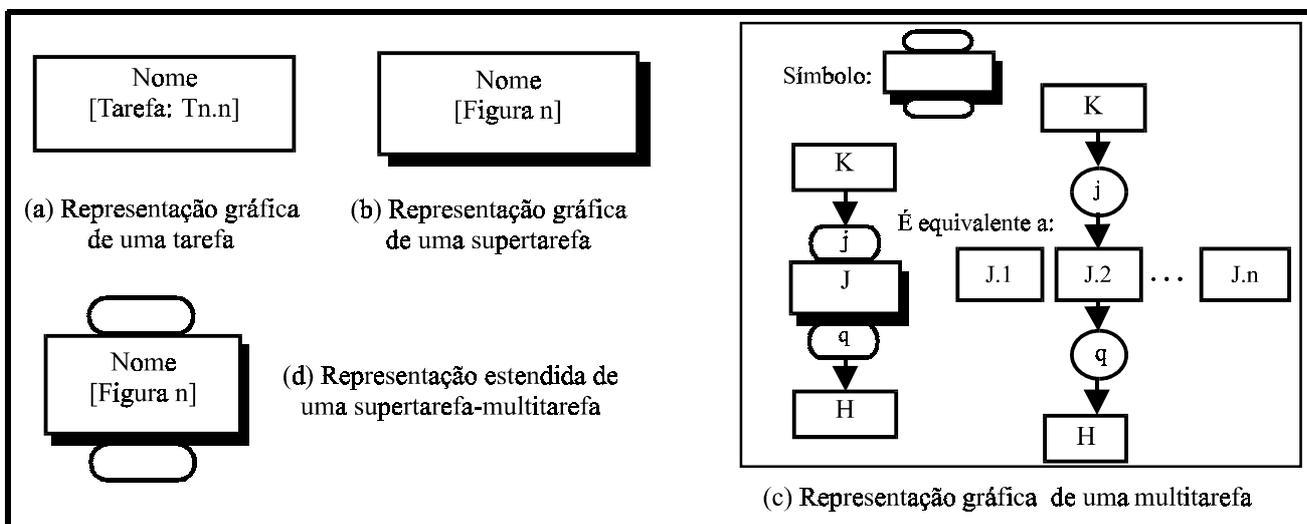


Figura 1 - Representação gráfica de elementos de WF

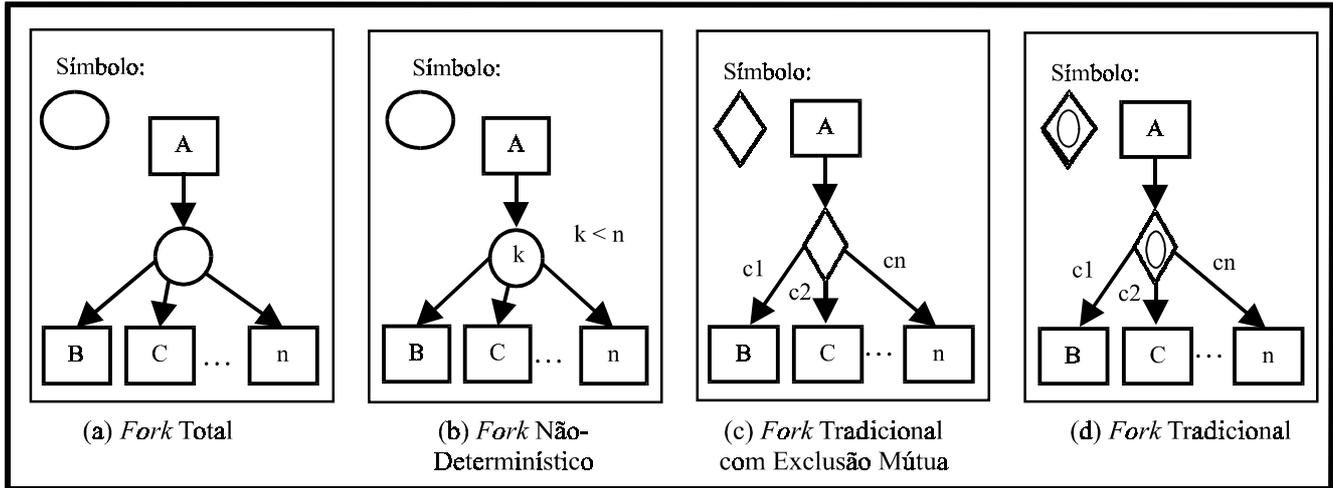


Figura 3: Tipos de Fork

No **fork total**, após o término da tarefa A, todas as suas atividades sucessoras são habilitadas (Figura 3-a). No **fork não-determinístico** após o término da tarefa A, um número K de tarefas sucessoras é escolhido não deterministicamente para habilitação (Figura 3-b). No **fork condicional com exclusão mútua**, a condição é avaliada e somente uma tarefa é habilitada (Figura 3-c). No **fork condicional**, a condição é avaliada e somente as tarefas sucessoras com condição verdadeira são habilitadas (Figura 3-d).

Nos **joins** uma tarefa A é precedida por um conjunto de outras tarefas predecessoras, com uma das quatro formas de relação: no **total** a tarefa M é habilitada somente após o término de todas as suas tarefas predecessoras (Figura 4-a); no **parcial** a tarefa M é habilitada, após o término de um número k de tarefas predecessoras (Figura 4-b); no **iterativo**, a tarefa M é habilitada após cada término de um número K de tarefas predecessoras. Assim, a tarefa M pode ser habilitada várias vezes, gerando várias execuções em paralelo (Figura 4-c).

2.1.5 INÍCIO E FIM

Os símbolos de **início e fim** indicam respectivamente o início e o fim de uma instância de WF. Após o símbolo de início (Figura 5) habilita-se a atividade sucessora dele; e quando se habilita o símbolo de fim, a instância do WF é terminada, implicando o cancelamento de qualquer atividade em execução.

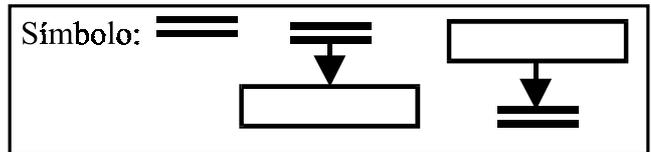


Figura 5 - Símbolos de início e fim do WF

2.2 TEMPLATES QUE COMPLEMENTAM O MODELO DE WF

O detalhamento de uma tarefa é representado através do *template* apresentado na Tabela 1. Cada um dos itens relacionados é explicado a seguir.

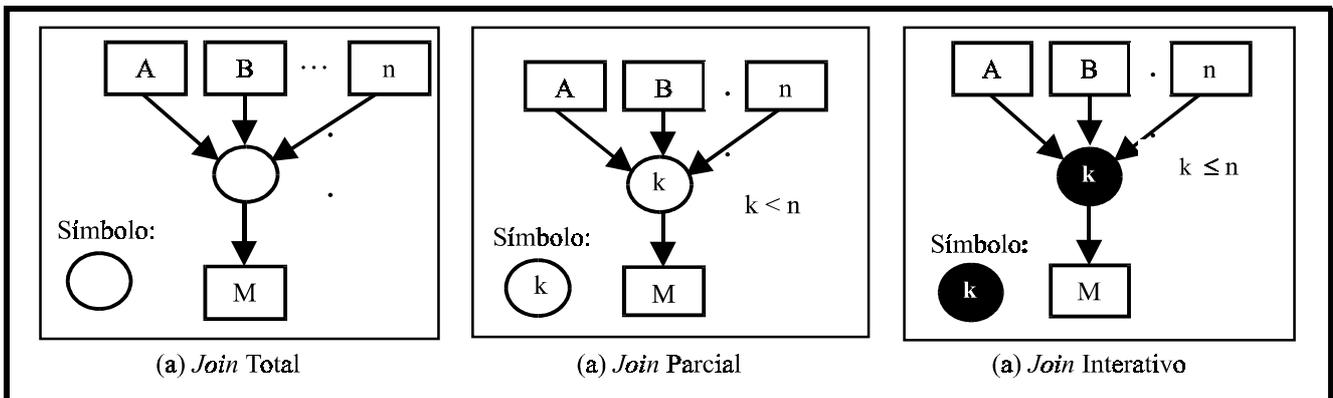


Figura 4 - Tipos de Join

Item	Caracterização da Tarefa	
Domínio	Fase:	Supertarefa:
Nome da Tarefa	Tn.n:	
Descrição		
Ações		
Pré-condição		
Pós-condição		
Exceção/Reação		
Agentes		
Recursos		

Tabela 1 - *Template* para descrição de uma tarefa

- ◆ **Domínio**<sup>3</sup> (informação obrigatória): onde é informado o contexto em que a tarefa está inserida. Definiu-se que neste item deveriam ser acrescentados os subitens **Fase** (de autoria ou de execução) e a **Supertarefa** (do *workflow*) na qual a tarefa está inserida. Este último subitem também pode ser preenchido com informações da multitarefa relacionada, devendo ser informado todo o caminho para se acionar a referida tarefa. As supertarefas nas quais a tarefa descrita estiver diretamente inserida deverão ser precedidas pelo símbolo  $\Rightarrow$ .
- ◆ **Nome da tarefa**<sup>4</sup> (informação obrigatória): campo que receberá o nome da tarefa, devendo ser semelhante à nomenclatura utilizada no *workflow*. Definiu-se que neste item, precedendo o nome da tarefa, deverá ser acrescentado um identificador<sup>5</sup> **Tn.n**, onde **T** significa Tarefa, o primeiro **n** refere-se à fase (sendo  $n=1$  para autoria e  $n=2$  para execução) e o segundo **n** refere-se ao número aleatório de identificação da tarefa na fase (Ex.: **T1.5** significa a tarefa 5 da fase autoria).
- ◆ **Descrição**<sup>6</sup> (informação não obrigatória): deverão ser utilizadas poucas linhas, em linguagem natural, descrevendo-se o propósito da tarefa e, se necessário, colocando-se alguma observação relevante sobre a execução da tarefa.
- ◆ **Ações**<sup>7</sup> (informação obrigatória): descrição, em linguagem natural, das ações que deverão ser realizadas durante a execução da tarefa. As ações definidas não serão efetuadas necessariamente na seqüência apresentada, sendo possível que algumas não tenham ordem pré-estabelecida para efetivação. Caberá ao agente responsável definir a seqüência de execução das mesmas. As ações serão responsáveis por manipular as informações através de recursos disponíveis para tal (no caso, os recursos disponíveis são computador, banco de dados (BD), multimídia e *internet*). Os tipos básicos de ações que poderão ser executadas são: definir um valor; escolher entre valores; obter, inserir e atualizar dados em um banco de dados; obter e enviar dados para um dispositivo externo; filtrar informações; testar condição para disparar outras ações; associar e validar informações; elaborar e armazenar instrumento de estudo; atualizar, em um banco de dados, o *status* do atributo indicativo de conclusão da tarefa (conclusão de todas as ações previstas). É importante ressaltar que esse conjunto de ações não é completo, tratando-se de uma sugestão de possíveis ações. Concluídas todas as ações previstas, será automaticamente executada a ação de atualizar positivamente o *status* de conclusão da tarefa que, por sua vez, será o evento que ativará as tarefas subseqüentes.
- ◆ **Pré-condição**<sup>8</sup> (informação obrigatória): definição, em linguagem natural, de quais eventos dispararam a execução da tarefa. Tais eventos podem ser: (1) conclusão da tarefa predecessora; (2) resultado de um *fork*; (3) resultado de um *join*; e (4) início do *workflow*. Existindo mais de um evento a ser tratado na pré-condição, deverá ser utilizada expressão *booleana* que, se verdadeira, disparará a execução da tarefa.
- ◆ **Pós-condição**<sup>9</sup> (informação não obrigatória): definição, também em linguagem natural, da condição que deverá estar satisfeita quando da

<sup>3</sup> Extraído de CIMOSA (Zelm, 1995).<sup>4</sup> Extraído de Casati/Ceri (Casati et al., 1995) e de CIMOSA (Zelm, 1995).<sup>5</sup> CIMOSA (Zelm, 1995) prevê um item em separado para receber o identificador do processo.<sup>6</sup> Extraído de Casati/Ceri (Casati et al., 1995) e de CIMOSA (Zelm, 1995).<sup>7</sup> Mescla dos itens ações (extraído de Casati/Ceri (Casati et al., 1995)) e *events* (extraído de CIMOSA (Zelm, 1995)).<sup>8</sup> Mescla dos itens pré-condições (extraído de Casati/Ceri (Casati et al., 1995)) e *boundary* (extraído de CIMOSA (Zelm, 1995)).<sup>9</sup> Item que não consta nos métodos de referência (Casati/Ceri (Casati et al., 1995) e CIMOSA (Zelm, 1995)), tendo sido extraído do Modelo de Gatilhos (Joosten, 1994, 1995).

conclusão da tarefa. Na prática, será a verificação, no banco de dados, do *status* do indicativo de conclusão da tarefa (ressaltando-se que a última ação realizada na execução da tarefa será sempre a atualização do *status* de tal atributo).

- ◆ **Exceção/reação**<sup>10</sup> (informação não obrigatória): onde deverão ser previstas as situações excepcionais que podem acontecer durante a execução do *workflow* e quais as ações necessárias para tratá-las. Também neste item será utilizada a linguagem natural, semelhante à utilizada no item ações.
- ◆ **Agentes**<sup>11</sup> (informação obrigatória): relação de todos os agentes envolvidos na execução da tarefa, sendo os responsáveis pela realização das ações previstas para a mesma.
- ◆ **Recursos**<sup>12</sup> (informação não obrigatória): relação dos recursos necessários para a execução da tarefa. Os recursos são utilizados pelos agentes como instrumentos para a realização das ações.

### 3 WF DE AUTORIA DE UM CURSO

O WF de autoria (Sizilio, 2000) engloba as tarefas relacionadas à criação do curso (concepção, planejamento, definição e implementação do mesmo). Sugere-se que na execução das tarefas estejam envolvidos os seguintes agentes: autor do curso (responsável pelo projeto e por convocar, se necessário, os demais agentes) e profissionais das áreas de computação, pedagogia, comunicação e psicologia. Sugere-se, ainda, que sejam disponibilizados recursos tais como: computador; banco de dados; *internet* e multimídia.

A definição/elaboração de todo o material didático, atividades, formas de comunicação e tutoria é realizada durante a execução deste modelo, sendo este o

momento mais demorado da implementação. É, também, o que apresenta maior custo, envolvendo a maior parte dos recursos e dos profissionais da equipe.

No WF apresentado, várias tarefas são opcionais (a critério do autor do curso), sendo várias as possibilidades de seqüência de execução das mesmas, gerando, conseqüentemente, cursos com estruturas diferentes.

#### 3.1 O *WORKFLOW* DE AUTORIA

O WF de autoria de cursos é apresentado na Figura 6. Quatro tarefas básicas são identificadas: cadastrar curso, definir critérios de análise e desempenho do aluno, implementar módulos e liberar curso para a execução. Como cada uma destas é composta por outras atividades, são representadas através de supertarefas, e são representadas por WFs específicos, em separado. Para cada supertarefa que for definida nesses WFs adicionais corresponde também um WF específico. Deste modo, o WF completo de autoria é composto pelo conjunto de todos os WFs correspondentes. Neste artigo apresentamos somente os WFs das supertarefas que compõem o WF inicial. O conjunto completo do WF de autoria, com todos os *templates* que completam sua definição, está disponível na *world wide web*, nas URLs <http://www.uniderp.br/nead/sizilio> e <http://www.inf.ufrgs.br/~tapejara>.

Um WF define não somente as tarefas a serem executadas, mas também sua seqüência e coordenação. Deste modo, o WF básico de autoria (Figura 6) define que o início da autoria de um curso se dá com o cadastro do curso, durante o qual são definidas ementas, módulos, conteúdos por módulo etc. Uma vez terminado esse cadastro, devem ser definidos critérios para a análise do desempenho do aluno, passando-se, em seguida, à implementação de todos os módulos definidos durante o cadastro do curso, e finalizando com a liberação do curso para execução. A seguir são detalhadas tais supertarefas.

<sup>10</sup> Mescla dos itens exceção/reação (extraído de Casati/Ceri (Casati et al., 1995)) e constrains (extraído de CIMOSA (Zelm, 1995)).

<sup>11</sup> Item que não consta nos métodos de referência (Casati/Ceri (Casati et al., 1995) e CIMOSA (Zelm, 1995)), tendo sido definido a partir dos conceitos básicos de workflow (Barthelmes et al., 1995; Workflow Management Coalition, 1996).

<sup>12</sup> Item que não consta nos métodos de referência (Casati/Ceri (Casati et al., 1995) e CIMOSA (Zelm, 1995)), tendo sido definido a partir dos conceitos básicos de workflow (Barthelmes et al., 1995; Workflow Management Coalition, 1996).

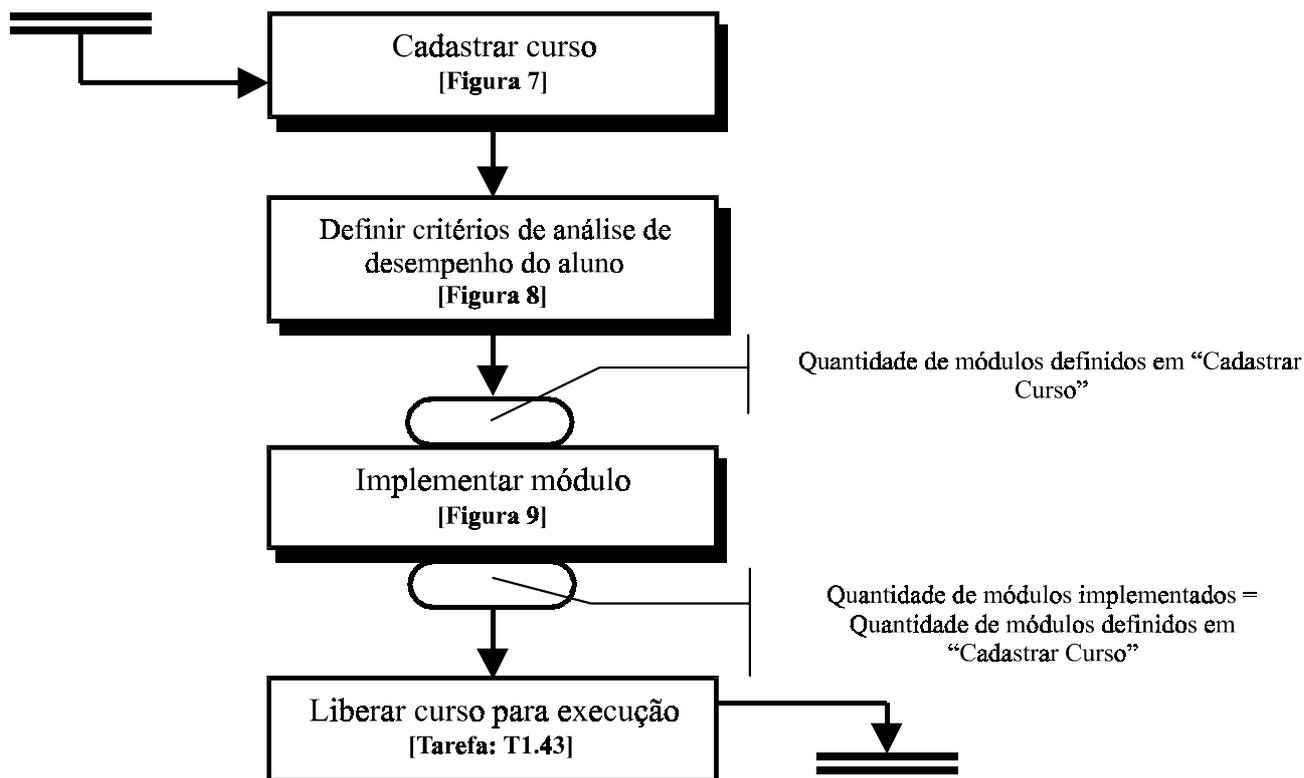


Figura 6 - Workflow de autoria de curso de EAD

### 3.2 SUPERTAREFA CADASTRAR CURSO

A supertarefa cadastrar curso (Figura 7) é responsável pela identificação e definição dos parâmetros básicos do curso. Observando o WF correspondente a esta supertarefa, nota-se que é composta por várias tarefas, que deverão ser detalhadas em *templates* correspondentes, cujos dados são definidos pelo autor ao executar as tarefas.

O cadastro de um curso inicia-se com a identificação deste curso, através da tarefa identificar curso, quando então os dados relativos a este curso são inseridos no *template* correspondente, T1.1. Uma vez finalizada esta identificação, duas tarefas devem ser executadas (em qualquer ordem, ou paralelamente): definir ementa, na qual são definidos os objetivos do curso, e definir quantidade de módulos, em que o conteúdo é decomposto (um curso deverá ter no mínimo um módulo).

O WF representa que as duas tarefas anteriores devem estar concluídas para que seja dado prosseguimento à definição do curso. Neste momento cinco outras tarefas podem ser iniciadas, podendo também ser executadas em qualquer ordem ou em paralelo.

Estas tarefas tratam da definição de bibliografia, de instrumentos de auto-estudo que serão disponibilizados, de instrumentos de comunicação que serão utilizados no decorrer do curso, de instrumentos de avaliação e dos eventuais momentos presenciais. Em cada uma delas devem ser analisadas todas as possibilidades disponíveis, e como se adaptam ao tipo de curso a ser implementado. Deste modo, ao finalizar todas estas tarefas, estarão selecionados e definidos todos os elementos relevantes para a execução do curso, sendo sinalizada a finalização desta supertarefa.

### 3.3 SUPERTAREFA DEFINIR CRITÉRIOS DE ANÁLISE DE DESEMPENHO DO ALUNO

Esta supertarefa (Figura 8) é utilizada para que o autor possa definir os critérios de análise de desempenho que serão utilizados para avaliar o aluno, verificando se os objetivos inicialmente colocados foram atingidos. Caso o autor opte por um curso sem tutoria (por exemplo, aplicação hipermídia para instrução distribuída sem tutoria), poderá descartar todos os critérios de desempenho disponíveis. Se, no entanto, for importante a avaliação do aluno, um ou mais dos seguintes aspectos podem ser utilizados como critérios:

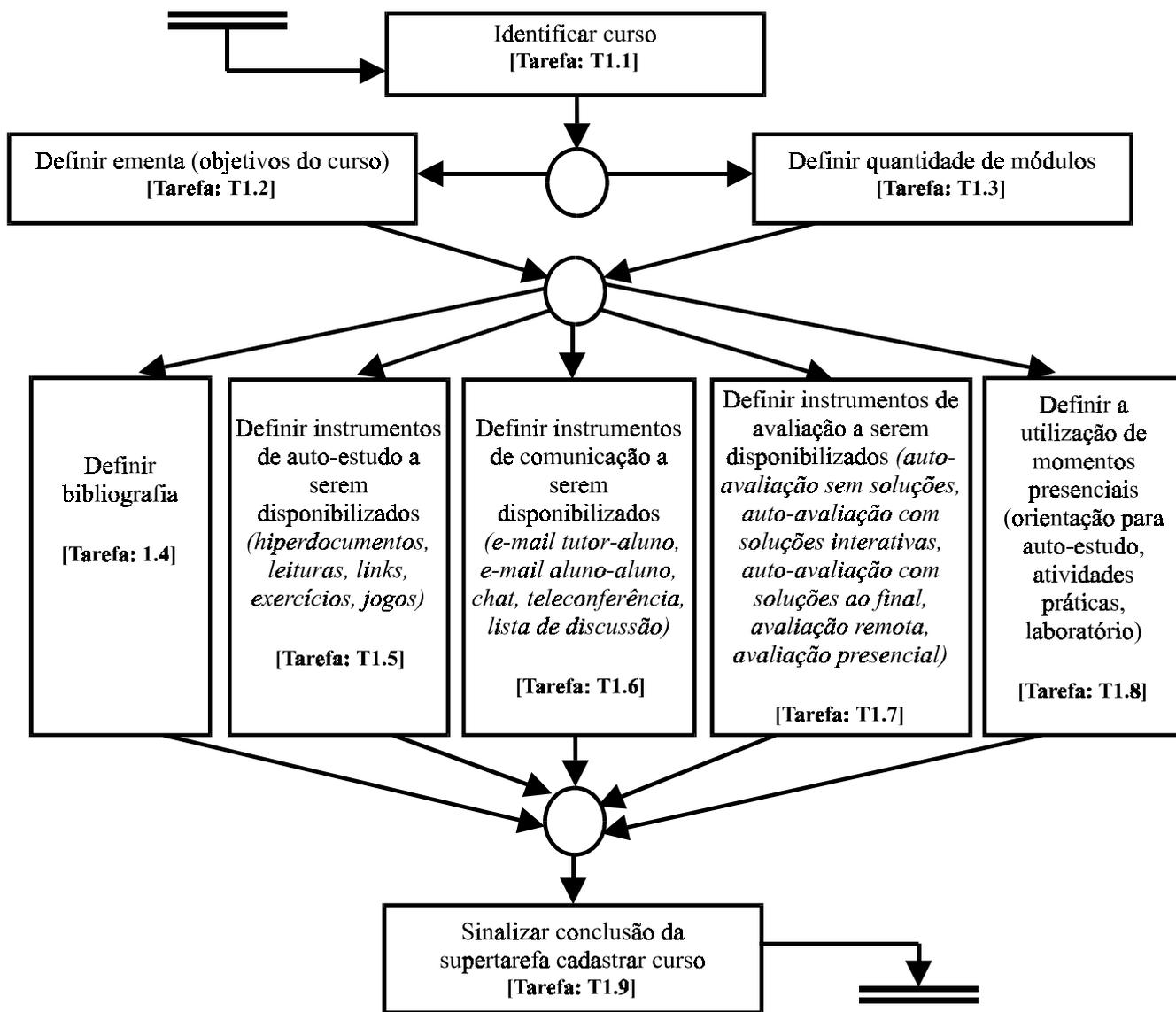


Figura 7 - Workflow de autoria de curso de EAD ⇒ Supertarefa: cadastrar curso

(1) utilização de “nota” obtida a partir de avaliações; (2) análise do tempo que o aluno levou para executar determinadas tarefas; e (3) registro de frequência do aluno ao curso. O WF guia o autor, fazendo com que sejam analisadas as possibilidades de utilização de cada um desses critérios e indicando quando devem ser definidos valores para a avaliação.

### 3.4 SUPERTAREFA-MULTITAREFA IMPLEMENTAR MÓDULO

A supertarefa-multitarefa implementar módulo (Figura 9) deve ser executada para cada um dos módulos anteriormente definidos. Inicia-se com a identificação do módulo e de seus objetivos, sendo definida a sua ementa, que deve estar compatível com a ementa do curso. Em seguida devem ser definidas quais as tare-

fas, dentre as que definem aspectos genéricos para o módulo, deverão obrigatoriamente ser implementadas.

Uma série de tarefas deve ser executadas a seguir, levando à definição de (1) utilização ou não de pré-condições para a execução do módulo; (2) se deve ou não ser aplicada uma avaliação inicial quando o aluno iniciar o módulo; (3) aspectos temporais para o módulo, como por exemplo, tempo para executar uma ou mais atividades; (4) definição de indicativo de conclusão do módulo, com possível sugestão do próximo módulo a ser cursado.

Paralelamente a essas definições devem ser estabelecidos quais e quantos instrumentos serão implementados para desenvolver a ementa definida para o

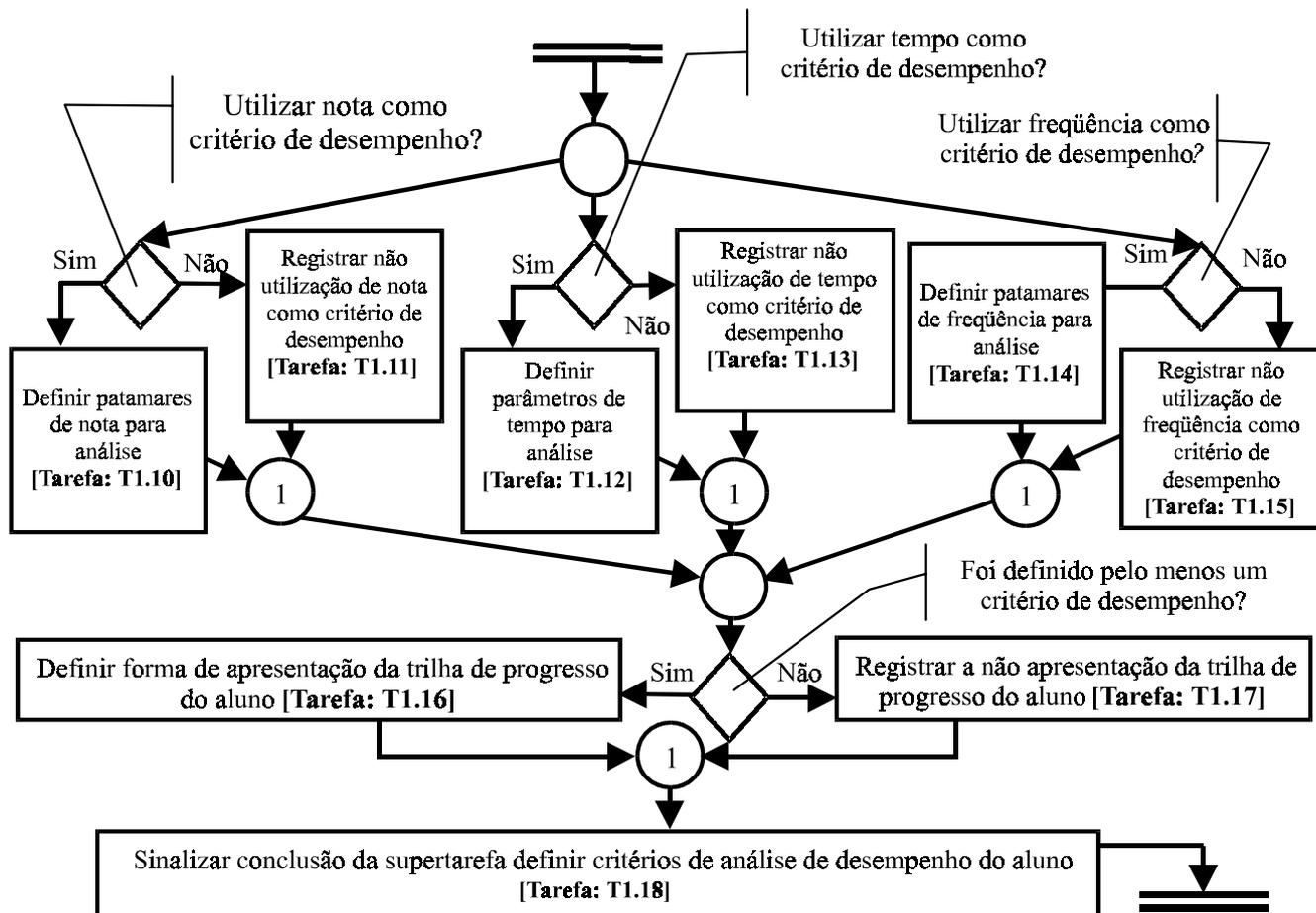


Figura 8 - Workflow de autoria de curso de EAD ⇒ Supertarefa: definir critérios de análise de desempenho do aluno

módulo. Esses instrumentos de estudo são então efetivamente elaborados, através da supertarefa multitarefa elaborar instrumentos (Figura 10).

### 3.5 EXEMPLOS DE TEMPLATES

A tarefa definir quantidade de módulos que comporão o curso (Figura 7) apresenta um exemplo simples de *template* (Tabela 2), onde o autor do curso (único agente envolvido) utiliza os recursos disponíveis (computador e banco de dados) para executar as ações previstas, condicionado ao atendimento das pré e pós condições estabelecidas. Observe-se, ainda, que um tratamento de exceção é previsto.

A tarefa T1.5 (definir instrumentos de auto-estudo a serem disponibilizados), detalhada na Tabela 3, é fundamental para que o autor do curso, logo no início do planejamento, defina quais os tipos de instrumento de auto-estudo serão utilizados no desenvolvimento do curso. Também, neste caso, é previsto um tratamento de exceção. Ressalta-se que definição seme-

lhante deverá ser feita para instrumentos de comunicação, instrumentos de avaliação e momentos presenciais.

Observe-se que na pré-condição para execução da tarefa definir patamares de nota para análise (Tabela 4), além de testar se o evento predecessor foi concluído, é realizado um teste de condição, em função da existência, no *workflow*, de um *fork* condicional com exclusão mútua. Existem outros *templates* responsáveis por definir outros parâmetros para análise (tempo e frequência).

Já na tarefa T1.16 (Tabela 5), a pré-condição é verificada através de uma sentença lógica.

Nas tarefas que envolvem várias exceções, deverá haver um tratamento específico para cada uma delas, conforme caracterizado na tarefa T1.16 (Tabela 6).

É fundamental a definição da quantidade de instrumentos de cada tipo a serem elaborados. A tarefa T1.32 (Tabela 7) norteia a definição da quantidade de

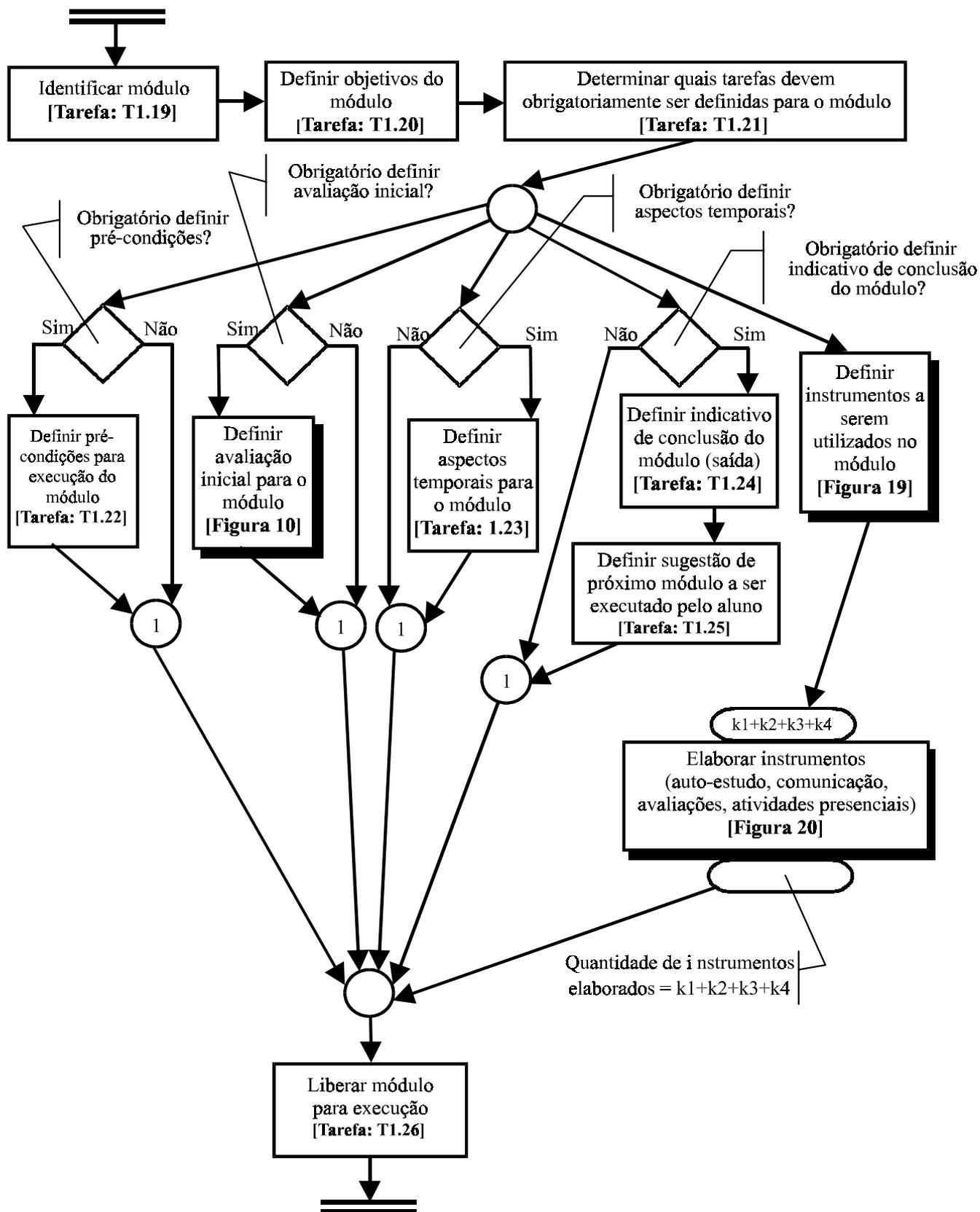


Figura 9 - Workflow de autoria de curso de EAD ⇒ Supertarefa-Multitarefa: implementar módulo

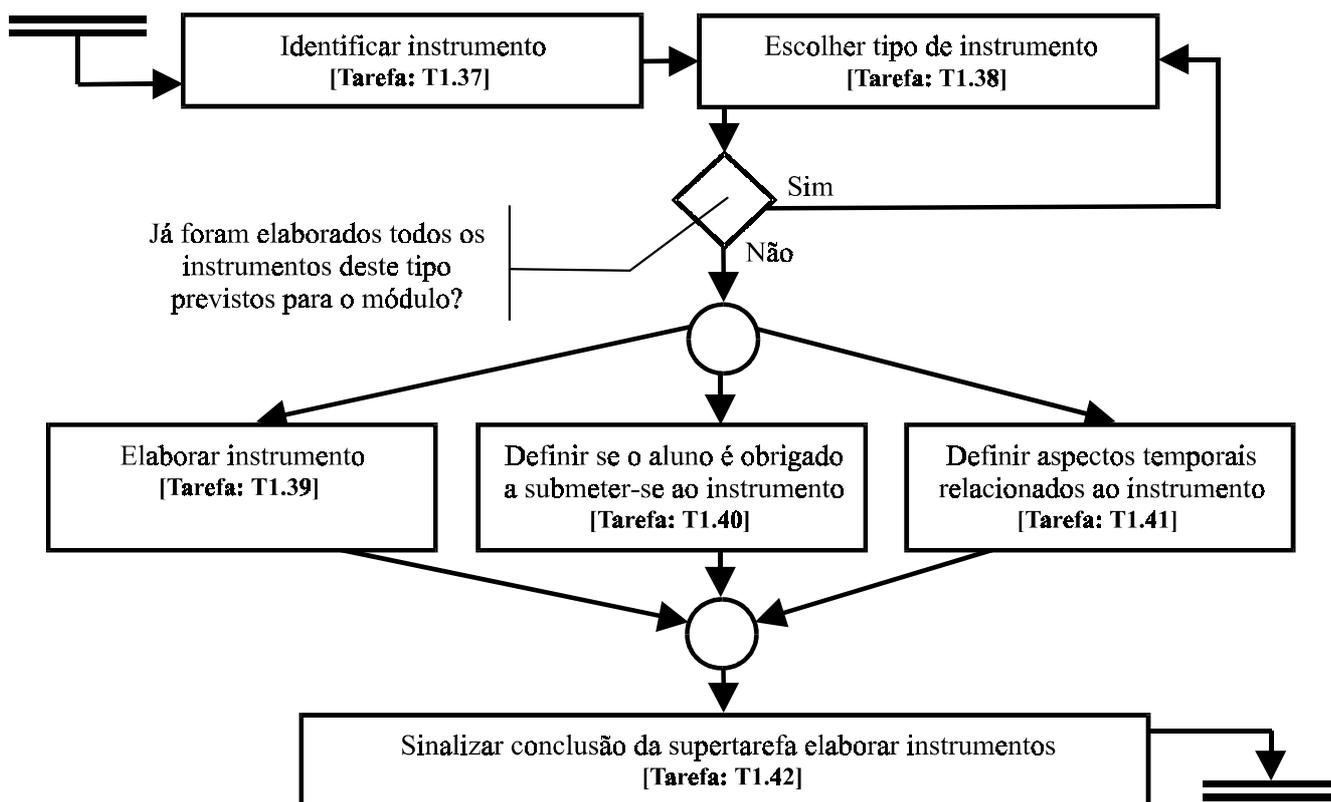


Figura 10 - *Workflow* de autoria de curso de EAD ⇒ Supertarefa-Multitarefa: elaborar instrumentos (oriunda da Supertarefa-Multitarefa: implementar módulo)

instrumentos de auto-estudo que deverão ser produzidos, existindo procedimento semelhante para a definição também de instrumentos de comunicação, avaliação e momentos presenciais.

Tendo havido um bom planejamento, ou seja, execução com qualidade de todas tarefas antecessoras à Tarefa 1.39 (Tabela 8), incluindo aqui adequadas tomadas de decisão no decorrer de todo o *workflow* de autoria, a execução desta tarefa terá como resultado final a implementação efetiva do curso.

Observe-se que vários agentes são responsáveis por executar as ações, caracterizando uma equipe multidisciplinar. Além disto, todos os recursos disponíveis serão utilizados neste momento.

Ressalta-se que esta tarefa será executada tantas vezes quantas forem necessárias, para elaborarem-se todos os instrumentos definidos para o curso.

## 4 EXEMPLO DE UTILIZAÇÃO DO WF DE AUTORIA

Nesta seção é mostrada a utilização do WF aqui proposto, na autoria de um tipo particular de curso - um curso na modalidade de aplicação hipermídia para instrução distribuída, sem tutoria. A definição dos requisitos que o curso deve apresentar, quando aplicada ao WF proposto, resulta em uma seqüência de passos a serem executados na autoria de qualquer curso que apresente as mesmas características - no caso, um curso sem tutoria.

### 4.1 SEQÜÊNCIA DE EXECUÇÃO DO WF DE AUTORIA DE UMA APLICAÇÃO HIPERMÍDIA PARA INSTRUÇÃO DISTRIBUÍDA SEM TUTORIA

O objetivo deste exemplo é a autoria de um curso sem tutoria, que poderá ser executado por alunos de forma totalmente independente, a qualquer momento,

Item	Caracterização da Tarefa
Domínio	Fase: Autoria Supertarefa: ⇒ Cadastrar curso
Nome da Tarefa	T1.3: DEFINIR QUANTIDADE DE MÓDULOS
Descrição	O autor define a quantidade de módulos (>=1) que comporão o curso, suficiente para o desenvolvimento da ementa definida.
Ações	Obter e inserir no BD a quantidade (Q) de módulos que comporão o curso, devendo Q ser maior ou igual a 1.
Pré-condição	Conclusão do evento de atualização positiva do <i>status</i> indicativo de conclusão da tarefa T1.1.
Pós-condição	Quantidade de módulos estar informada, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.3.
Exceção/Reação	Exceção: valor obtido para Q é menor ou igual a zero. Reação: execução do <i>workflow</i> de autoria para o curso é cancelada.
Agentes	Autor.
Recursos	Computador, Banco de Dados.

Tabela 2 - *Template* T1.3 – Definir quantidade de módulos

Item	Caracterização da Tarefa
Domínio	Fase: Autoria Supertarefa: ⇒ Cadastrar curso
Nome da Tarefa	T1.5: DEFINIR INSTRUMENTOS DE AUTO-ESTUDO A SEREM DISPONIBILIZADOS
Descrição	Registrar quais tipos de instrumentos de auto-estudo (hiperdocumento, leitura, exercício, jogo, link, etc) serão disponibilizados na elaboração dos módulos e definir um código para cada tipo de instrumento de auto-estudo.
Ações	Inserir no BD informação de que instrumentos de auto-estudo serão disponibilizados. Obter e inserir no BD os códigos dos tipos de instrumento de auto-estudo que serão disponibilizados.
Pré-condição	Conclusão dos eventos de atualização positiva do <i>status</i> indicativo de conclusão das tarefas T1.2 e T1.3.
Pós-condição	Estarem selecionados e codificados todos os tipos de instrumento de auto-estudo que serão disponibilizados, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.5.
Exceção/Reação	Exceção: Código informado para o tipo de instrumento é desconhecido. Reação: Ignorar a informação e enviar mensagem indicando que a informação foi ignorada.
Agentes	Autor.
Recursos	Computador, Banco de Dados.

Tabela 3 - *Template* T1.5 – Definir instrumentos de auto-estudo a serem disponibilizados

Item	Caracterização da Tarefa
Domínio	Fase: Autoria Supertarefa: ⇒ Definir critérios de análise de desempenho do aluno
Nome da Tarefa	T1.10: DEFINIR PATAMARES DE NOTA PARA ANÁLISE
Descrição	Registrar a utilização de nota como critério de avaliação. Registrar as notas máxima e mínima a serem atribuídas ao aluno em uma avaliação. Registrar quantos patamares de nota serão utilizados para análise pelos tutores. Definir o percentual máximo da nota a ser relacionado a cada patamar registrado.
Ações	Inserir no BD informação de que o critério nota será utilizado na análise do desempenho do aluno. Obter e inserir no BD informações relativas ao critério nota, tais como: nota mínima, nota máxima, quantidade de patamares de nota para classificação do estudante e seus respectivos valores máximos (percentuais).
Pré-condição	Conclusão do evento de atualização positiva do <i>status</i> indicativo de conclusão da tarefa T1.9 e do evento resultante do teste de condição “Utilizar nota como critério de avaliação?” = “Sim”.
Pós-condição	Estar registrada a utilização de nota como critério de avaliação e seus respectivos parâmetros, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.10.
Exceção/Reação	Exceção: Algum parâmetro da nota deixar de ser informado. Reação: Inserir no BD informações <i>default</i> para tais parâmetros.
Agentes	Autor.
Recursos	Computador, Banco de Dados.

Tabela 4 - *Template* T1.10 – Definir patamares de nota para análise

Item	Caracterização da Tarefa
Domínio	Fase: Autoria    Supertarefa: ⇒ Definir critérios de análise de desempenho do aluno
Nome da Tarefa	T1.16: DEFINIR FORMA DE APRESENTAÇÃO DA TRILHA DE PROGRESSO DO ALUNO
Descrição	Verificar quais os critérios a serem utilizados para análise de desempenho do aluno e definir a forma de apresentação da trilha de progresso do aluno: (1) Nota, (2) Tempo, (3) Frequência, (4) Nota e Tempo, (5) Nota e Frequência, (6) Tempo e Frequência, (7) Nota, Tempo e Frequência.
Ações	Inserir no BD, a partir dos critérios de análise de desempenho selecionados, a forma de apresentação da trilha de progresso do aluno.
Pré-condição	Conclusão dos eventos de atualização positiva do <i>status</i> indicativo de conclusão das tarefas e do evento resultante do teste de condição citados a seguir, segundo resultado verdadeiro da sentença lógica: ((T1.10 ou T1.11) e (T1.12 ou T1.13) e (T1.14 ou T1.15)) e ("Foi definido pelo menos um critério de desempenho?" = "Sim").
Pós-condição	Estar definida a forma de apresentação da trilha de progresso do aluno, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.16.
Exceção/Reação	
Agentes	Autor.
Recursos	Computador, Banco de Dados.

Tabela 5: *Template* T1.16–Definir forma de apresentação da trilha de progresso do aluno

Item	Caracterização da Tarefa
Domínio	Fase: Autoria    Supertarefa: ⇒ Multitarefa implementar módulo
Nome da Tarefa	T1.22: DEFINIR PRÉ-CONDIÇÕES PARA EXECUÇÃO DO MÓDULO
Descrição	Definir quais os módulos que deverão ter sido concluídos com sucesso pelo aluno, para que o módulo seja liberado para ser executado pelo aluno. Obs.: Isto permite que módulos sejam executados (estudados) em paralelo e/ou em seqüência aleatória por um aluno, desde que ele tenha cumprido os pré-requisitos.
Ações	Obter e inserir no BD as identificações de todos os módulos que forem pré-requisito para a execução do módulo corrente.
Pré-condição	Conclusão do evento de atualização positiva do <i>status</i> indicativo de conclusão da tarefa T1.21 e do evento resultante do teste de condição "Obrigatório definir pré-condições?" = "Sim".
Pós-condição	Estarem definidas as pré-condições para execução do módulo, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.22.
Exceção/Reação	Exceção 1: serem obtidas identificações de módulos não existentes. Reação 1: rejeitar tais módulos, enviando mensagem informando tal rejeição. Exceção 2: identificação obtida ser igual à identificação do módulo corrente. Reação 2: rejeitar tal identificação, enviando mensagem informando que um módulo não pode ser pré-requisito para execução dele próprio.
Agentes	Autor.
Recursos	Computador, Banco de Dados.

Tabela 6 - *Template* T1.22 – Definir pré-condições para execução do módulo

Item	Caracterização da Tarefa
Domínio	Fase: Autoria Supertarefa(s): Multitarefa implementar módulo ⇒ Definir instrumentos a serem utilizados no módulo
Nome da Tarefa	T1.32: DEFINIR QUANTIDADE (k1) DE INSTRUMENTOS DE AUTO-ESTUDO A SEREM ELABORADOS PARA O MÓDULO
Descrição	Registrar a quantidade de instrumentos de auto-estudo a serem elaborados para o módulo. Esta Quantidade (k1) irá determinar a quantidade de execuções da multitarefa "Elaborar atividades de auto-estudo".
Ações	Obter e inserir no BD a quantidade (k1) de instrumentos de auto-estudo a serem elaborados para o módulo, devendo k1 ser maior ou igual a 1.
Pré-condição	Conclusão do evento de atualização positiva do <i>status</i> indicativo de conclusão da tarefa T1.21 e do evento resultante do teste de condição "Foi definida a disponibilização de pelo menos um tipo de instrumento de auto-estudo?" = "Sim".
Pós-condição	Estar definida a quantidade de instrumentos de auto-estudo a serem elaborados para o módulo, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.32.
Exceção/Reação	Exceção: valor obtido para k1 é menor ou igual a zero. Reação: ignorar a informação e solicitar que a quantidade de instrumentos de auto-estudo seja reinformada.
Agentes	Autor.
Recursos	Computador, Banco de Dados.

Tabela 7 - *Template* T1.32 – Definir quantidade (k1) de instrumentos de auto-estudo a serem elaborados para o módulo

Item	Caracterização DA TAREFA
Domínio	Fase: Autoria Supertarefa(s): Multitarefa implementar módulo⇒ Multitarefa elaborar instrumentos
Nome da Tarefa	T1.39: ELABORAR INSTRUMENTO
Descrição	Elaborar instrumento.
Ações	Utilizar os recursos disponíveis para elaborar o instrumento, considerando o tipo de instrumento a ser utilizado.
Pré-condição	Estar sugerido o próximo módulo a ser executado pelo aluno, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.28.
Pós-condição	Estar elaborado o instrumento, através da verificação de situação positiva do <i>status</i> indicativo de conclusão da tarefa T1.39.
Exceção/Reação	
Agentes	Autor, profissionais das áreas de Computação, Pedagogia, Comunicação e Psicologia.
Recursos	Computador, Banco de Dados, <i>Internet</i> , Multimídia.

Tabela 8 - *Template* T1.39 – Elaborar instrumento

sem avaliação formal. Não deverá apresentar interação entre alunos.

A seqüência de execução do WF de autoria para implementação de um curso com as características acima é apresentada na Tabela 9.

Na autoria de um curso, as tarefas relacionadas devem ser executadas passo-a-passo, ressaltando que a execução de uma tarefa (Tn.n) nada mais é do que efetivamente realizar todas as **ações** previstas no *template* da referida tarefa, ou seja, deverá ser executado o que estiver indicado no item **ações** do *template*.

O resultado final da execução da seqüência de passos é a implementação efetiva do curso.

#### 4.2 APLICAÇÃO EM ESTUDOS DE CASOS REAIS

Para verificar a exatidão do WF gerado, foi simulada sua aplicação para gerar um curso de línguas pela Internet. A análise foi baseada no curso *Fluency Trough Fables*<sup>13</sup> (Comenius English Language Center, 1999), que é uma de muitas seções de ofertas do *website* do centro da língua inglesa Comenius (*Comenius English Language Center*). O objetivo do curso é ajudar estudantes a melhorar suas habilidades na língua inglesa, através de exercícios interativos. Nesse curso, os estudantes selecionam uma dentre seis pequenas fábulas disponíveis. Após a leitura, o estudante é submetido a exercícios de múltipla escolha envolvendo vocabulário e compreensão do texto.

Os estudantes podem verificar suas respostas logo após a conclusão de cada exercício, sendo informados do seu índice de acerto.

Após analisar o curso existente, procurou-se verificar se ele poderia ter sido gerado através da aplicação dos passos apresentados na Tabela 9. Isto seria possível se cada uma das fábulas apresentadas no curso fosse considerada como um módulo. Assim, os passos 15 e 16 da Tabela 9 corresponderiam à escolha da fábula, e dos objetivos que seriam buscados em cada uma delas. No passo 17 seriam definidas as tarefas a serem percorridas em cada fábula. Como instrumentos de auto-estudo (passo 18) todos os módulos (fábulas) apresentariam um texto pelo menos – a fábula apresentada. Além deste, outros instrumentos de auto-estudo podem ser acrescentados aos módulos. Dependendo dos objetivos buscados em cada fábula apresentada, o número de instrumentos de auto-avaliação seria variável (passo 19). Seguindo os passos indicados, os instrumentos de auto-estudo e de auto-avaliação seriam definidos e implementados. Deste modo, foi comprovado que a aplicação do WF gerado poderia dar origem ao curso analisado.

Foram realizadas, também, algumas experiências isoladas de utilização do WF de autoria (proposto neste artigo) na modelagem de cursos *on-line* da modalidade salas de aulas virtuais. Para essas experiências foram escolhidos autores procedentes de áreas de conhecimento distintas (ciência da computação, psicologia e educação). Foram modelados seis cursos de extensão (apenas os dois módulos iniciais de cada curso): (1) Algoritmos e Estrutura de Dados; (2) Java: Acesso a Banco de Dados; (3) Introdução ao Processamento de Dados; (4) Metodologia do Ensino; (5) Psicologia na Educação; e (6) Operações Matemáticas Básicas. As modelagens dos cursos, que tiveram como principal objetivo validar a utilização geral do método proposto, foram realizadas da seguinte forma:

<sup>13</sup> Disponível em <http://www.comenius.com/fable/index.html>

Passo	Tarefa	Descrição
1	T1.1	Identificar curso.
2	T1.2	Definir ementa [objetivos do curso].
3	T1.3	Definir quantidade de módulos.
4	T1.4	Definir bibliografia.
5	T1.5	Definir instrumentos de auto-estudo a serem disponibilizados.
6	T1.6	Definir instrumentos de comunicação a serem disponibilizados [tratando-se um curso sem tutoria, não é prevista nenhuma comunicação entre alunos e/ou tutor].
7	T1.7	Definir instrumentos de avaliação a serem disponibilizados [neste caso, apenas auto-avaliação].
8	T1.8	Definir utilização de momentos presenciais [neste caso não haverá momentos presenciais].
9	T1.9	Sinalizar conclusão da supertarefa cadastrar curso.
10	T1.11	Registrar não-utilização de nota como critério de desempenho.
11	T1.13	Registrar não-utilização de tempo como critério de desempenho.
12	T1.15	Registrar não-utilização de frequência como critério de desempenho.
13	T1.17	Registrar não-apresentação da trilha de progresso do aluno.
14	T1.18	Sinalizar conclusão da supertarefa Definir Critérios de Análise de Desempenho do Aluno. Para cada módulo identificado, repetir passos 15 a 33.
15	T1.19	Identificar módulo.
16	T1.20	Definir objetivos do módulo.
17	T1.21	Determinar quais tarefas devem obrigatoriamente ser definidas para o módulo.
18	T1.32	Definir quantidade (k1) de instrumentos de auto-estudo a serem elaborados para o módulo.
19	T1.34	Definir quantidade (k3) de instrumentos de auto-avaliação a serem elaborados para o módulo.
20	T1.36	Sinalizar conclusão da supertarefa definir instrumentos a serem utilizados no módulo.
21	T1.37	Identificar instrumento de auto-estudo.
22	T1.38	Escolher tipo de instrumento de auto-estudo.
23	T1.39	Elaborar instrumento de auto-estudo.
24	T1.40	Definir se o aluno é obrigado a submeter-se ao instrumento de auto-estudo.
25	T1.41	Definir aspectos temporais relacionados ao instrumento de auto-estudo.
26	T1.42	Sinalizar conclusão da supertarefa Elaborar Instrumentos (auto-estudo). Para cada instrumento de avaliação escolhido, repetir passos 27 a 32.
27	T1.37	Identificar instrumento de avaliação.
28	T1.38	Escolher tipo de instrumento de avaliação.
29	T1.39	Elaborar instrumento de avaliação.
30	T1.40	Definir se aluno é obrigado a submeter-se ao instrumento de avaliação.
31	T1.41	Definir aspectos temporais relacionados ao instrumento de avaliação.
32	T1.42	Sinalizar conclusão da supertarefa Elaborar Instrumentos.
33	T1.26	Liberar módulo para execução.
34	T1.43	Liberar curso para execução.

Tabela 9 - Sequência de execução do WF de autoria para curso da modalidade aplicação hiperfídia para instrução distribuída sem tutoria

os dois primeiros cursos foram modelados por professores de curso de Ciência da Computação da UNIDERP (implementados utilizando-se o *software* de autoria ToolBook); o terceiro por um aluno concluinte do curso de Ciência da Computação na modelagem de protótipos para a sua monografia de conclusão de curso, em fase final, sobre estudo comparativo entre *softwares* de autoria (implementado em duas versões: uma utilizando ToolBook e outra VisualClass); os quarto e quinto cursos foram modelados, respectivamente, por professores das áreas de educa-

ção e psicologia da UNIDERP (não tendo sido implementados); e o sexto por um aluno concluinte do curso de Matemática (implementado utilizando-se ToolBook). As modelagens não estão apresentadas neste artigo por serem muito extensas – como exemplo, cita-se a modelagem do curso de Algoritmos e Estrutura de Dados (Sizilio, 2000) que tem 147 passos.

Inicialmente buscou-se identificar somente se os autores teriam capacidade de entender a simbologia

utilizada no WF proposto, de modo a poder utilizá-lo na elaboração de cursos. Verificou-se que o entendimento do WF dependia do conhecimento prévio do autor:

- ◆ **autor da área de ciência da computação** – quando o autor era da área de ciência da computação, estando já habituado a utilizar modelos para representar aplicações, o entendimento do modelo proposto foi bastante simples. Quando o autor já tinha conhecimento prévio do conceito de modelagem de *workflow*, um conceito não muito comum em ciência da computação, o entendimento foi melhor ainda. Entretanto, a falta de conhecimento da área educacional fez com que este tipo de autor apresentasse dificuldades na definição dos instrumentos disponíveis para EAD, sendo necessário um estudo inicial nesta área;
- ◆ **autor das áreas educacional e de psicologia** – neste caso, o entendimento do modelo foi dificultado pela falta de conhecimentos de modelagem em geral, e de *workflow* em específico. Foi necessário uma capacitação nesses assuntos para que o WF proposto pudesse ser utilizado.

Para ambos os tipos de usuários avaliados foi necessário algum treinamento (de modelagem ou de técnicas e recursos educacionais). Esta conclusão era esperada, considerando-se que a proposta engloba mais de uma área de conhecimento.

A partir das avaliações iniciais realizadas, foi possível concluir que o WF somente poderá ser utilizado se for proporcionado um treinamento prévio aos autores, explicando-lhes claramente: (1) o objetivo do próprio WF; (2) o método utilizado para expressar o WF, e; (3) como o WF deve ser utilizado para apoiar a construção de um curso de EAD.

A eficácia da utilização do WF durante a autoria de um curso, entretanto, ainda não foi avaliada. Foram iniciadas algumas experiências para medir a eficiência do WF de autoria na geração de cursos, em diferentes áreas de conhecimento e para diferentes modalidades de EAD via *www* (principalmente salas de aulas virtuais). Para estas experiências estão sendo definidos parâmetros mensuráveis, de modo a permitir a obtenção de resultados concretos da relação

custo x benefício quanto à utilização do WF para apoiar a autoria de um curso de EAD.

## 5 CONCLUSÕES E PERSPECTIVAS

O ensino a distância é uma realidade mundial, constituindo-se imperativo face às exigências de um mundo globalizado, facilitado pela acelerada evolução da tecnologia. Neste contexto, a informática, especialmente a rede mundial de informação proporcionada pela *Internet*, constitui-se em um instrumento privilegiado a ser utilizado em prol da democratização do conhecimento. Verifica-se, no entanto, que, mesmo face à demanda crescente no setor, e às exigências do novo paradigma, não existe em nossa realidade uma especificação clara e completa para o desenvolvimento de cursos de EAD.

Este trabalho apontou as técnicas de modelagem de *workflow* como uma tecnologia capaz de modelar a criação de cursos de EAD e implementar os conceitos relacionados à interação entre os processos, proporcionando um tratamento adequado aos aspectos dinâmicos que se apresentaram, às restrições temporais para sua execução e à troca de controle entre tarefas. Neste contexto, procurou-se cobrir uma lacuna com relação à definição de técnica de modelagem conceitual para cursos remotos implementados na *world wide web*.

A continuidade deste ou de outros trabalhos nesta linha envolve, principalmente, aprofundamento em todos os aspectos relacionados à modelagem de *workflow* aplicada ao ensino a distância, abrangendo desde a validação completa e extensão do modelo proposto até o desenvolvimento de ferramentas voltadas para implementação de cursos de EAD.

Uma vertente interessante para se pesquisar é a definição e formalização de pseudo-linguagem para ser utilizada nos *templates*, tanto para expressar as ações e exceções/reações como para definir as pré e pós-condições, o que pode ser relevante para dar um maior formalismo ao modelo. Também se pode sugerir que seja efetuado um estudo mais aprofundado do tratamento dos aspectos temporais e dados dinâmicos inerentes à modelagem de *workflow* aplicada ao EAD.

Certamente a aplicação e a evolução dos resultados obtidos neste trabalho contribuirão para o desenvol-

vimento da tecnologia para implementação de cursos de ensino distância via *world wide web*.

## REFERÊNCIAS BIBLIOGRÁFICAS

- CASATI, F., & CERI, S., & PERCINI, B., & POZZI, G. Conceptual modeling of workflows. In Proceedings of The Object-Oriented and Entity-Relationship Conference, Gold Coast/Austrália: 1995.
- COMENIUS ENGLISH LANGUAGE CENTER. Fluency trough fables. Disponível em <http://www.comenius.com/fable/index.html> (15/04/1999).
- JOOSTEN, S. Trigger modelling for workflow analysis. In: CON'94: WORKFLOW MANAGEMENT, CHALLENGES, PARADIGMS AND PRODUCTS, 1994. Viena. Proceedings. Viena: [s.n.], 1994.
- JOOSTEN, M.M.S. A method for analysing workflow. Atlanta/GA: Georgia State University/Computer Information System Department, 1995.
- NICOLAO, M., & VICCARI, R.M., & EDELWEISS, N., & OLIVEIRA, J.P.M. Modelagem conceitual de workflow para cursos na internet. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 9., 1998, Fortaleza, CE. Anais... Fortaleza: Sociedade Brasileira de Computação, 1998. CD-ROM.
- OLIVEIRA, J.P.M., & NICOLAO, M., & EDELWEISS, N. Conceptual workflow modelling for remote courses. In: INTERNATIONAL IFIP CONFERENCE DISTANCE LEARNING, TRAINING AND EDUCATION, 1998, Viena. Proceedings... Viena: TELETEACHING, 1998. P.789-798.
- SIZILIO, G.R.M.A. Técnicas de modelagem de *workflow* aplicadas à autoria e execução de cursos de ensino a distância. PPGC da UFRGS, janeiro de 2000. Dissertação de Mestrado.
- WORKFLOW MANAGEMENT COALITION. Interface 1: process definition interchange. Bruxelas, Jun. 1996. 42p. Disponível em <http://www.aiai.ed.ac.uk/project/wfmc.index.html> (06/04/1999).
- ZELM, M., & VERNADAT, F.B., & KOSANKE, K. The CIMOSA business modelling process. Computers in Industry, n. 27, pp. 123-142, Elsevier, 1995.