

Universidade Federal do Rio Grande do Sul  
Instituto de Informática  
Curso de Pós-Graduação em Ciência da Computação

**Aplicação da Técnica de Análise  
*Object Modeling Technique (OMT)*  
ao Problema da IFIP**

por

**Guillermo Bustos Reinoso**

RP - 232

Junho 94

Relatório de Pesquisa

UFRGS - II - CPGCC

Caixa Postal 15064 CEP 91501-970

Porto Alegre - RS - BRASIL

Telefone: (051) 336-8399 e 339-1355

Fax: (051) 336-5576

Email: PGCC @ INF.UFRGS.BR

## SUMÁRIO

Lista de Figuras.....	iii
Resumo .....	iv
Abstract.....	iv
Introdução .....	1
1. Definição do Problema .....	3
2. Aplicação das Técnicas Integracionistas .....	7
2.1. A Análise na Técnica de Modelagem de Objetos (OMT) e o Caso da IFIP .....	7
2.1.1. Modelagem de Objetos .....	7
2.1.2. Modelagem Dinâmica .....	10
2.1.3. Modelagem Funcional.....	12
2.2. As Críticas à Técnica de Análise OMT.....	15
Anexo 1 - Dicionário de Dados das Classes.....	19
Anexo 2 - Diagramas de Fluxo de Eventos .....	20
Anexo 3 - Diagrama de Fluxo de Dados da Modelagem Funcional.....	21
Anexo 4 - Descrição das Funções da Modelagem Funcional.....	24
Referências Bibliográficas.....	28

## LISTA DE FIGURAS

Figura 1 - Modelo de objetos.....	8
Figura 2 - Diagrama de Fluxo de Eventos.....	11
Figura 3 - Diagrama de Estados (statechart) da classe Participante.....	12
Figura 4 - Diagrama de Estados (statechart) da classe Artigo.....	13
Figura 5 - Diagrama de Fluxo de Dados do nível 0.....	14

## RESUMO

É aplicada a técnica de análise orientada a objetos que faz parte da metodologia denominada *Object Modeling Technique (OMT)*, apresentada por Rumbaugh et al., ao caso do problema da IFIP.

A técnica de análise OMT é seguida passo a passo, mostrando os resultados em termos de diagramas: modelo de objetos, dicionário de dados, diagramas de fluxo de eventos, diagramas de estados, diagramas de fluxo de dados e descrição de funções.

Finalmente, são apresentadas algumas observações e críticas à aplicação, conforme o processo de análise, a especificação e representação da análise e outros.

*Palavras-chaves:* desenvolvimento orientado a objetos, análise orientada a objetos, técnica de análise na metodologia OMT, problema da IFIP.

## ABSTRACT

The object-oriented analysis technique from the methodology named *Object Modeling Technique (OMT)*, developed by Rumbaugh et al., is applied to the IFIP problem.

OMT analysis technique is followed in order step by step, showing the outcomes in terms of an object model, a data dictionary, events flow diagrams, state diagrams, data flow diagrams and a functions description.

Finally, some observations and critics are discussed about analysis process, object-oriented specification and others, using this technique.

*Keywords:* object-oriented development, object-oriented analysis, *OMT* methodology analysis technique, IFIP problem.

## INTRODUÇÃO

As técnicas de análise sob o paradigma da orientação a objeto é hoje uma área fértil para a pesquisa. Assim hoje é possível encontrar na literatura um conjunto heterogêneo de propostas, sem que nenhuma técnica de análise orientada a objetos pareça a dominar objetivamente.

Em [BUS 93] foram classificadas estas técnicas, e descritas as propostas mais representativas de cada categoria. Posteriormente foram comparadas utilizando diversos critérios.

Estes critérios foram divididos em aqueles relativos ao processo de análise e aqueles relativos à especificação e representação da análise. Entre os primeiros estavam os critérios de modelagem (estrutural, dinâmica e funcional) de objetos, incluindo os critérios para a identificação e localização de componentes, particionamento da complexidade, reusabilidade, construção de modelos, verificações e procedimento da análise. Entre os critérios de especificação estavam a representação de componentes dos modelos (estrutural, dinâmica e funcional), a quantidade de modelos, a complexidade aparente e a adaptabilidade ao domínio de aplicação.

Para completar devidamente a comparação já realizada, é necessário incluir critérios com relação à aplicação destas diversas técnicas. Neste sentido, o propósito específico deste relatório é fornecer uma aplicação de uma técnica específica de *Análise Orientada a Objetos (AOO)*: a que faz parte da metodologia denominada *Object Modeling Technique*, proposta por Rumbaugh et al. [RUM 91].

O caso utilizado é a definição do problema da IFIP (*International Federation for Information Processing*) que foi apresentado inicialmente em 1982, como base para uma conferência para o estudo comparativo de metodologias de projeto de sistemas de informação. Corresponde um problema amplamente divulgado na literatura e muito utilizado até hoje, portanto parece adequado para esta aplicação, dado que permitirá comparações com outras aplicações já desenvolvidas de técnicas de modelagem.

O relatório foi dividido como segue:

No *Capítulo 1* é fornecida a definição do problema da IFIP, tal e como será usada para a aplicação da técnica. Adicionalmente são indicados os supostos que foram estabelecidos para completar esta definição, tanto gerais, como relativos a objetos e atributos, relacionamento entre objetos e valoração de atributos.

O *Capítulo 2* apresenta a aplicação e crítica da técnica de análise OMT.

## 1. DEFINIÇÃO DO PROBLEMA

O caso escolhido é a definição do problema da IFIP (*International Federation for Information Processing*) proposto por Olle [OLL 82], que foi apresentado inicialmente em 1982, como base para uma conferência para o estudo comparativo de metodologias de projeto de sistemas de informação. O caso da IFIP é um problema amplamente divulgado na literatura e muito utilizado até hoje, portanto é adequado para este trabalho, dado que permitirá comparações com outras aplicações já desenvolvidas de técnicas de modelagem não orientadas a objetos, como por exemplo as apresentadas em [IFI 82].

O texto integral da definição do problema, a partir da referência [OLL 82], é o seguinte:

### “ESTUDO COMPARATIVO DE METODOLOGIAS DE PROJETO DE SISTEMAS DE INFORMAÇÃO

#### Definição do Problema

##### 1. Antecedentes

Uma Conferência de Trabalho da IFIP é uma conferência internacional que tenta reunir especialistas de todos os países da IFIP para discutir tópicos técnicos de interesse específico para um ou mais Grupos de Trabalho da IFIP. O procedimento usual, que será considerado para este propósito, é uma conferência convidada que não é aberta para qualquer um. Para tais conferências, parte do problema é assegurar que todos os membros relacionados com o(s) Grupo(s) de Trabalho e Comitê(s) Técnico(s) da IFIP sejam convidados, mesmo que eles não possam comparecer. Além disto, é importante assegurar suficientes assistentes à conferência de tal maneira que a viabilidade financeira seja atingida sem exceder o máximo indicado pelas facilidades disponíveis.

As políticas da IFIP sobre Conferências de Trabalho sugerem a nomeação de um Comitê de Programa para tratar com o conteúdo técnico da conferência e um Comitê de Organização para tratar das matérias

financeiras, preparação do local, convites e/ou publicidade. É claro que estes comitês precisam trabalhar juntos, possuem necessidades comuns de informação e precisam manter registros de informação consistentes e atualizados.

## 2. Sistema de informação a ser projetado

O sistema de informação a ser projetado deve suportar as atividades dos Comitês de Programa e Organização vinculados à preparação de uma Conferência de Trabalho da IFIP. O envolvimento destes dois comitês é considerado análogo a duas entidades organizacionais em uma estrutura corporativa que usam informação em comum.

Devem ser suportadas as seguintes atividades dos comitês.

### Comitê de Programa:

1. Preparar uma lista de pessoas para enviar a chamada de trabalhos.
2. Cadastrar as cartas de intenção em resposta à chamada.
3. Cadastrar os artigos de contribuição recebidos.
4. Distribuir os artigos entre aqueles que realizam a avaliação.
5. Coletar os relatórios dos avaliadores e selecionar os artigos para inclusão no programa.
6. Agrupar os artigos aceitos em sessões para apresentação e selecionar o moderador para cada sessão.

### Comitê de Organização:

1. Preparar uma lista de pessoas que receberão convite para a conferência.
2. Imprimir convites prioritários para Representantes Nacionais, membros do Grupo de Trabalho e membros de grupos de trabalho associados.
3. Assegurar de que todos os autores dos artigos aceitos recebam um convite.
4. Assegurar de que os autores de artigos rejeitados recebam um convite.
5. Evitar o envio de convites duplicados para qualquer pessoa.



6. Registrar a aceitação dos convites.
7. Gerar a lista final dos participantes.

### 3. Fronteiras do sistema

Deve ser notado que os aspectos financeiros e de cronograma do trabalho do Comitê de Organização, a planificação de ambos os comitês, as acomodações em hotéis para os participantes e as matérias relativas à preparação de cópias dos anais tem sido omitidos deste exercício, porém uma submissão pode incluir algum ou todos estes aspectos adicionais se os autores sentem-se motivados.”

Esta definição do problema é a única fonte consultada para a aplicação da técnica de AOO. Por isto é necessário acrescentar alguns supostos para garantir a completeza, consistência e a não ambigüidade do problema. Estes supostos são os seguintes:

#### *Geral*

- Dado que o tempo é um fator crítico na preparação da conferência, deve ser permitido o maior paralelismo possível entre as atividades das comissões, evitando estabelecer seqüências artificiais das mesmas.

#### *Objetos e atributos*

- A definição dos moderadores e avaliadores não é indicada explicitamente na definição do problema, porém é necessária sua inclusão para os efeitos de consistência dos modelos.
- O programa da conferência de trabalho é dividido em sessões agrupadas por assuntos. Estes mesmos assuntos de especialização são válidos para os artigos submetidos, como para os moderadores e avaliadores. Porém, dado que os assuntos não estão definidos no texto do problema, não serão identificados como uma classe independente e relacionada com as sessões, moderadores, avaliadores e artigos, e ficarão como atributos.

*Relacionamento entre objetos*

- Entre as pessoas que recebem a chamada de trabalhos, existem alguns que não participam da conferência, portanto não recebem convite.
- Não todos as pessoas que recebem uma chamada de trabalhos enviam uma carta de intenção.
- Não todas as pessoas que receberam algum tipo de convite, enviam respostas a este convite.
- Não todos os moderadores disponíveis necessariamente moderam alguma sessão.
- Não todos os avaliadores de trabalhos necessariamente avaliam algum artigo.

*Valoração de atributos*

- A carta de intenção como resposta à chamada de trabalhos pode indicar o desejo de participar ou não na conferência.
- A resposta ao convite recebido para a conferência pode ser uma aceitação do mesmo ou uma negativa.
- Os artigos que já foram aceitos ou rejeitados para participar da conferência não podem mais mudar o seu estado, isto é, um artigo aceito não poder ser rejeitado e vice-versa.

## 2. APLICAÇÃO DAS TÉCNICAS INTEGRACIONISTAS

As técnicas integracionistas são aquelas que combinam de maneira balanceada modelos nas diversas dimensões (estrutural, dinâmica e funcional), para integrá-los na especificação orientada a objetos. A técnica de análise OMT (*Object Modeling Technique*) de Rumbaugh et al. [RUM 91] é a mais representativa desta categoria. Outras técnicas são as de Shlaer & Mellor [SHL 89], [SHL 92], e Clyde et al. [CLY 92], [EMB 92].

A técnica integracionista a ser aplicada é a apresentada por Rumbaugh et al. [RUM 91], como parte da denominada *Técnica de Modelagem de Objetos* ou *Object Modeling Technique (OMT)*. Em [BUS 93] é apresentada uma descrição desta técnica e uma comparação com propostas de outras categorias da classificação.

### 2.1. A ANÁLISE NA TÉCNICA DE MODELAGEM DE OBJETOS (OMT) E O CASO DA IFIP

A fase de análise na técnica OMT, consiste na construção de três modelos, um para cada dimensão: modelagem de objetos, modelagem dinâmica e modelagem funcional, nesta mesma ordem.

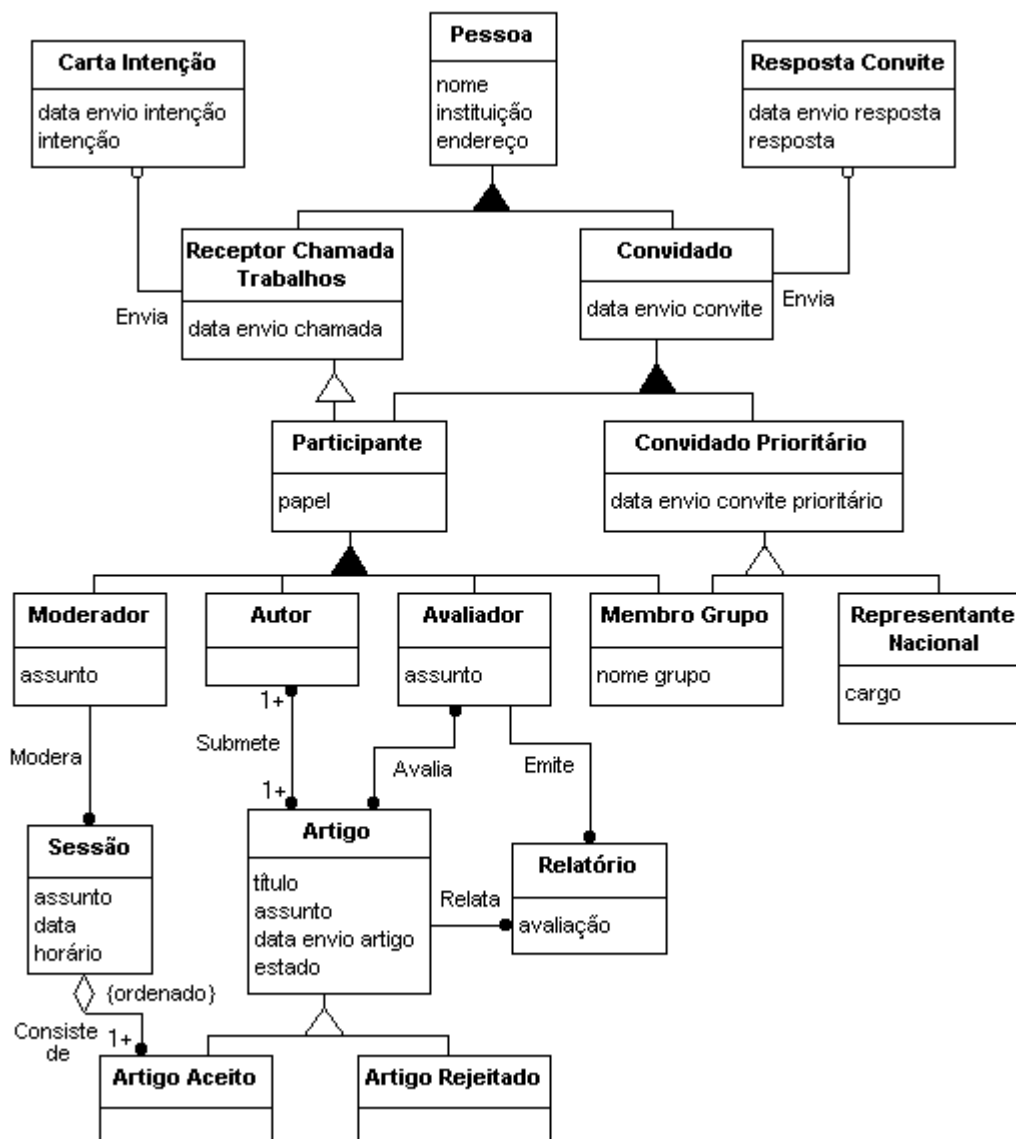
#### 2.1.1. Modelagem de Objetos

A *modelagem de objetos* consiste na construção de um modelo que representa o aspecto estático do sistema. Os passos desta modelagem são:

1. identificar objetos e classes;
2. preparar um dicionário de dados;
3. identificar associações (incluindo agregações ou composições) entre os objetos;
4. identificar atributos dos objetos e associações;
5. organizar e simplificar as classes de objetos usando herança;
6. verificar os passos de acesso para consultas;

7. iterar e refinar o modelo; e
8. agrupar as classes em módulos.

A figura 1 mostra o modelo de objeto resultante da aplicação dos passos 1 ao 7, excluindo o passo 2.



**Figura 1** - Modelo de objetos.

Na figura 1, cada caixa representa uma classe, com o nome em **negrito**, e seus atributos na parte inferior. As classes estão relacionadas por meio de associações

nomeadas (linhas que as unem) com uma notação especial para a cardinalidade. O símbolo ( $\odot$ ) na classe associada representa que 0 (nenhuma) ou 1 (uma) ocorrência faz parte da associação, a linha simples indica que exatamente 1 (uma) ocorrência está associada. Por exemplo, uma ocorrência de *Convidado* pode estar associada com nenhuma ocorrência de *Resposta Convite* ou no máximo uma ocorrência através de *Envia*; por sua parte uma ocorrência de *Resposta Convite* está associada a exatamente uma ocorrência de *Convidado*. O símbolo ( $\bullet$ ) na classe associada representa que 0 ou mais ocorrências daquela classe participam da associação, quando indicado um número, por exemplo 1+, o limite inferior é redefinido para 1 (um) e portanto a associação é de 1 ou mais ocorrências como ocorre no caso de *Autor* e *Artigo* e a associação *Submete*. Já o símbolo ( $\diamond$ ) indica uma associação de composição (ou agregação nos termos de Rumbaugh et al.), onde a notação das cardinalidades também é aplicável, como por exemplo *Sessão* e *Artigo Aceito*, na associação *Consiste de* com a restrição *{ordenado}* sobre o conjunto de ocorrências. As hierarquias de generalização-especialização são representadas por triângulos: ( $\triangle$ ) para subclasses sem interseção e ( $\blacktriangle$ ) para subclasses com sobreposição. Na figura por exemplo, um *Moderador* também pode ser *Membro Grupo*, já um *Artigo* pode ser *Artigo Aceito* ou *Artigo Rejeitado*, mas não ambos ao mesmo tempo.

O resultado da aplicação específica do passo 2 é apresentado no anexo 1 “Dicionário de Dados das Classes”, onde é fornecido o dicionário de dados para as classes identificadas no problema da IFIP. As classes resultantes de processos de generalização e especialização são excluídas do dicionário por serem obtidas no passo 5. Isto ocorre com as classes *Pessoa* e *Participante*.

Devido ao tamanho do problema da IFIP, não é preciso agrupar as classes em módulos como sugere o passo 8. A idéia é que cada módulo seja descrito em um diagrama de uma página. Contudo, um possível particionamento das classes poderia gerar dois módulos (vide figura 1): 1) módulo *Pessoa*, que inclui as classes *Carta Intenção*, *Receptor Chamada Trabalho*, *Resposta Convite*, *Convidado*, *Participante*, *Convidado Prioritário*, *Moderador*, *Autor*, *Avaliador*, *Membro Grupo* e *Representante Nacional*; e 2) módulo *Artigo*, que inclui as classes *Sessão*, *Artigo*, *Artigo Aceito*, *Artigo Rejeitado* e *Relatório*.

### 2.1.2. Modelagem Dinâmica

A *modelagem dinâmica* consiste na construção de modelos que representam o comportamento dinâmico da interação dos objetos em termos de estados, transições, eventos e ações. Os passos para esta modelagem são:

1. preparar cenários de seqüências de interações típicas;
2. identificar eventos entre objetos;
3. preparar um seguimento de eventos para cada cenário;
4. construir um diagrama de estado; e
5. verificar os eventos entre os objetos para verificar consistência.

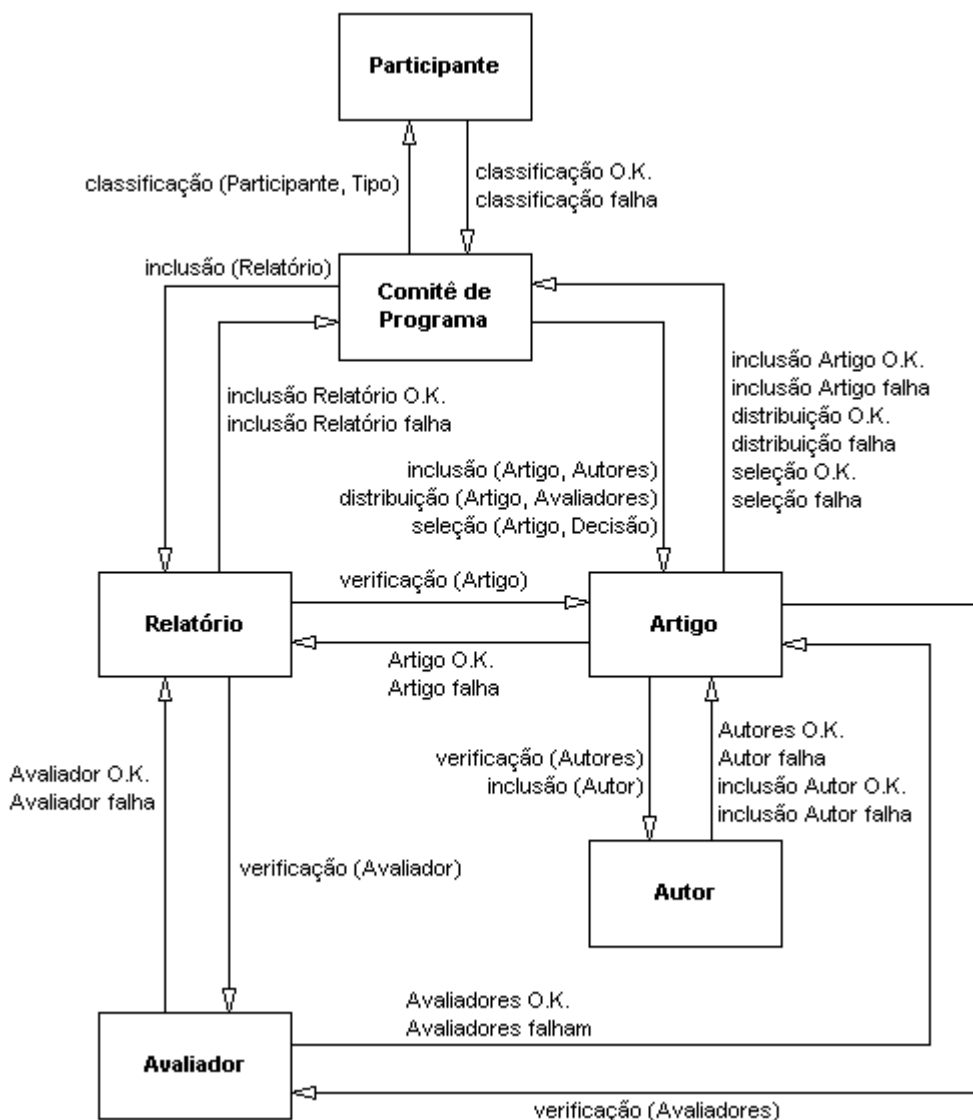
A aplicação dos passos 1 ao 3 dá como resultado o *Diagrama de Fluxo de Eventos* apresentado na figura 2, onde apenas as classes com maior relevância, desde o ponto de vista da modelagem dinâmica, foram incluídas. As outras classes são mostradas no anexo 2 “Diagramas de Fluxo de Eventos”.

Na figura 2, as classes ou agentes externos (*Comitê de Programa*) são representados por caixas, as setas representam fluxos de eventos entre as classes e agentes externos, sem considerar freqüências ou seqüências. Os eventos incluem também parâmetros, como por exemplo *classificação (Participante, Tipo)*, onde *Participante* e *Tipo* são parâmetros do evento *classificação*; e situações de erros, como por exemplo *seleção falha* entre o *Comitê de Programa* e *Artigo*.

As figuras 3 e 4 mostram a aplicação do passo 4 da modelagem dinâmica: a construção dos Diagramas de Estados (*statecharts* [HAR 87]) das classes. Especificamente, a figura 3 apresenta o modelo do comportamento da classe *Participante*, cuja interação com outros componentes é mostrada na figura 2. O mesmo acontece com a figura 4 que apresenta a classe *Artigo*. As classes restantes da figura 2 não são modeladas com diagramas de estados por apresentarem comportamentos triviais.

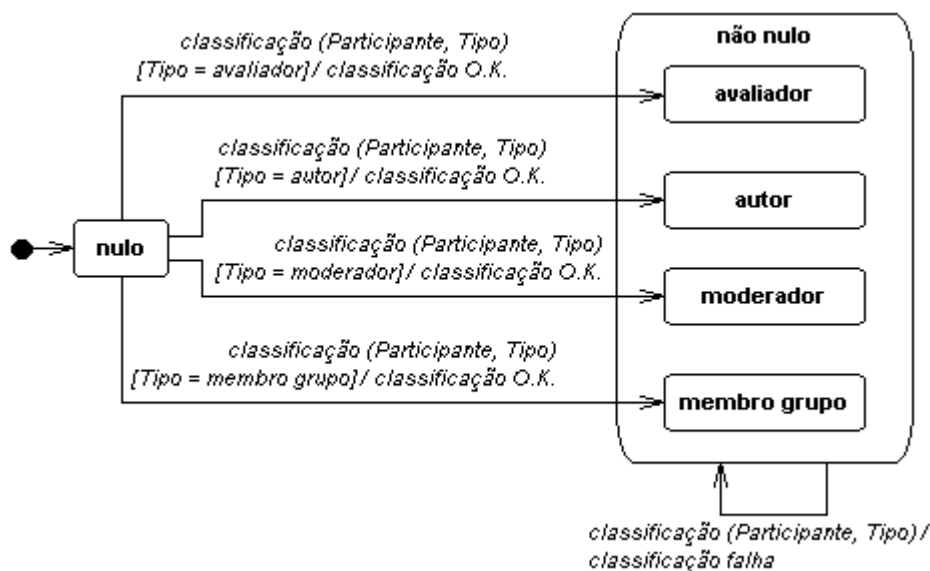
Nestas figuras, os estados possíveis são representados por caixas de cantos arredondados e as transições pelos arcos dirigidos. Cada transição segue a notação: *evento (parâmetro) [condição] / ação*, onde a transição é habilitada, ativando a *ação*, se o *evento* com o *parâmetro* ocorrer e a *condição* for verdadeira. Os estados podem apre-

sentar *atividades*<sup>1</sup> indicadas como *do: atividade*, como por exemplo *verificação (Avaliador)* na figura 4. Existem também duas transições especiais, a transição inicial indicada pelo círculo preto pequeno e a transição final indicada pelos círculos concêntricos. No exemplo, *inclusão (Artigo, Autor)* é a transição inicial (*default*) e *inclusão Autor falha / inclusão Artigo falha* e *Artigo falha / inclusão Artigo falha* são as transições finais na figura 4.



<sup>1</sup> A principal diferença entre uma *ação* nas transições e uma *atividade* nos estados, é que as ações sempre são instantâneas para os efeitos práticos, e as atividades podem tomar algum tempo para sua completção [RUM 91].

**Figura 2** - Diagrama de Fluxo de Eventos.



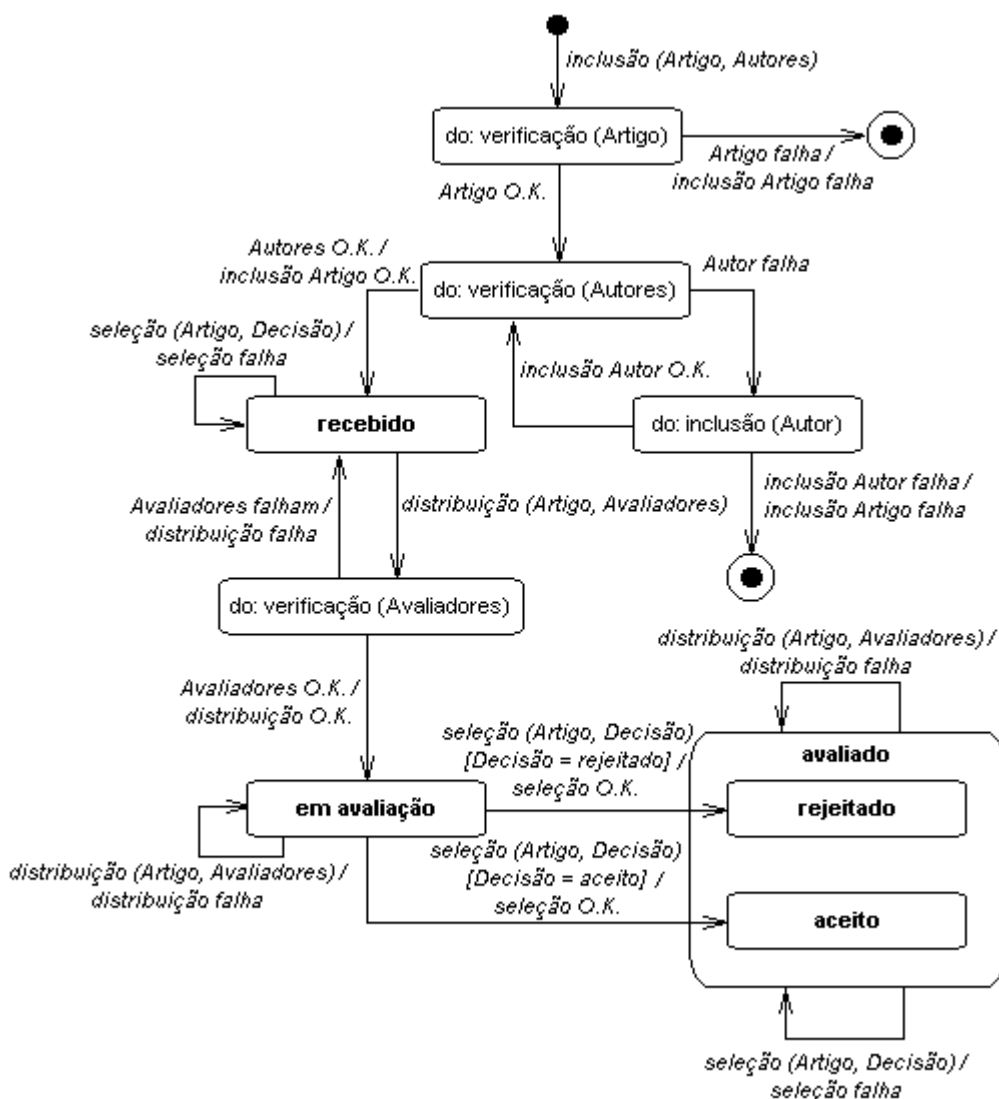
**Figura 3** - Diagrama de Estados (*statechart*) da classe *Participante*.

### 2.1.3. Modelagem Funcional

A *modelagem funcional* consiste na construção de um modelo funcional baseado em DFDs para representar as transformações ao interior do sistema. Os passos que devem ser realizados para esta modelagem são:

1. identificar valores de entrada e saída;
2. construir um diagrama de fluxo de dados que mostre as dependências funcionais;
3. descrever as funções;
4. identificar as restrições; e
5. especificar critérios de otimização.



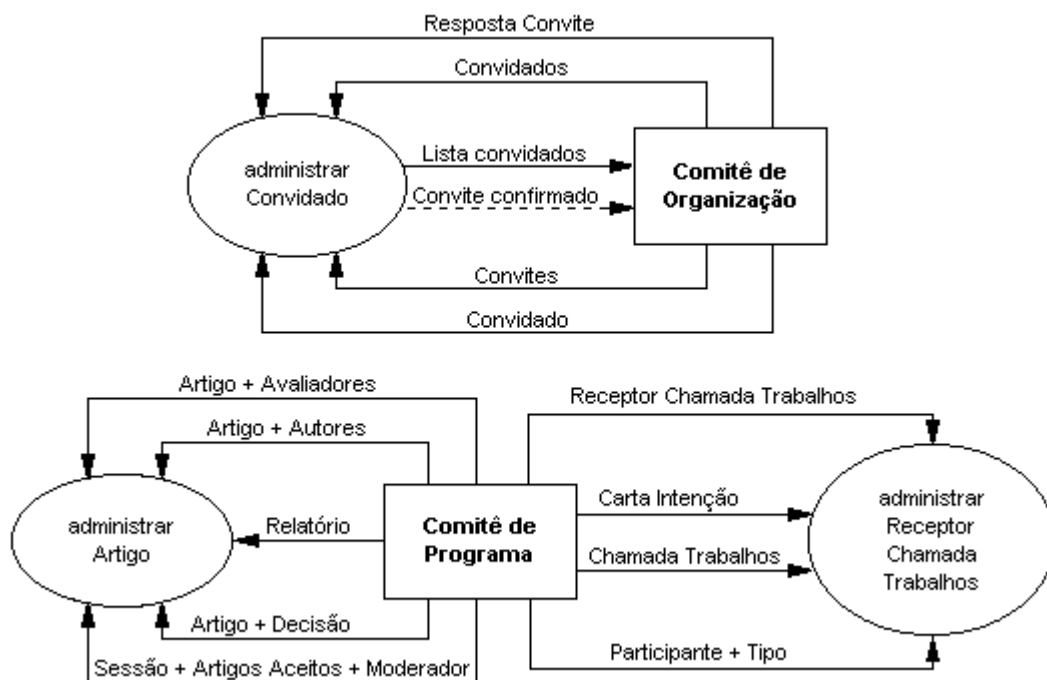


**Figura 4** - Diagrama de Estados (*statechart*) da classe *Artigo*.

A aplicação dos passos 1 e 2 entrega como resultado o DFD de alto nível da figura 5. Pode observar-se que as notações utilizadas nos diagramas são as convencionais da análise estruturada. A exceção corresponde aos fluxos de controle (variáveis lógicas) que são representados por setas de linhas descontínuas, como por exemplo *Convite confirmado*.

A explosão dos processos do DFD da figura 5 é apresentado no anexo 3 “Diagrama de Fluxo de Dados da Modelagem Funcional”. Nele pode ser observada uma

notação adicional para os fluxos de dados, que corresponde à criação de ocorrências (objetos) nas classes e são representadas com setas com pontas vazias maiores.



**Figura 5** - Diagrama de Fluxo de Dados do nível 0.

O anexo 4 “Descrição das Funções da Modelagem Funcional”, apresenta a descrição das funções do último nível de decomposição do DFD da figura 5 mostradas no anexo 3, segundo o estabelecido no passo 3 da modelagem funcional da técnica de Rumbaugh et al.

Em relação ao passo 4, as restrições já foram incluídas nos diferentes modelos, por exemplo o fato que uma ocorrência de *Artigo* não pode mudar seu estado uma vez aceito ou rejeitado. Esta restrição está representado no diagrama de estado desta classe (figura 4) e na descrição da função *selecionar Artigo*.

Finalmente, considerando o passo 5, um critério de otimização possível é consequência da consideração de que o tempo é um fator crítico na preparação da confe-

rência, portanto deve ser considerada a maior concorrência possível nas atividades realizadas pelas comissões.

Uma vez completadas as três modelagens, os autores propõem um passo específico para acrescentar operações. Estas operações podem vir do modelo de objetos (acesso a atributos), dos eventos, ações e atividades do modelo dinâmico, e das funções do modelo funcional. Indicam que devem ser mostradas no modelo de objetos apenas aquelas funções de relativa complexidade computacional. No caso do problema da IFIP, todas as funções são relativamente simples (vide descrições do anexo 4) e, portanto, não foram representadas no modelo de objetos.

O resultado final da aplicação da técnica de análise OMT consiste no seguinte:

1. Modelo de objetos (figura 1)
2. Dicionário de dados (anexo 1)
3. Modelo dinâmico:
  - Diagramas de fluxo de eventos (figura 2 e anexo 2)
  - Diagramas de estado (*statecharts* das figuras 3 e 4)
4. Modelo funcional:
  - Diagramas de fluxo de dados (figura 5 e anexo 3)
  - Descrição das funções (anexo 4)
  - Restrições e critérios de otimização (acima indicados)

## 2.2. AS CRÍTICAS À TÉCNICA DE ANÁLISE OMT

A seguir são indicadas algumas observações realizadas a partir da aplicação da técnica de Rumbaugh et al. ao problema da IFIP:

- *Estratégia “middle-out”*: A técnica de análise OMT apresenta uma estratégia *middle-out*, isto é, não é claramente *top-down* nem *bottom-up*, identificando, estruturando e especificando *linearmente* os objetos em termos estruturais, dinâmicos e funcionais.

- *Procedimento da técnica*: O procedimento da análise da metodologia OMT é estruturado em torno das três dimensões da modelagem, o que simplifica a sua aplicação ao centrar o processo de modelagem em um aspecto de cada vez.
- *Modelagem de objetos*: Todos os mecanismos fornecidos para a modelagem de objetos permitem uma grande flexibilidade na construção dos modelos. A potencialidade oferecida dificilmente pode ser totalmente utilizada em apenas um modelo.
- *Definição de atributos*: A técnica não indica explicitamente a necessidade de uma definição mais detalhada dos atributos que inclua uma descrição da semântica e dos domínios. Uma definição deste tipo é necessária para o problema da IFIP, para aqueles atributos de valores enumerados como *estado* da classe *Artigo* ou *papel* de *Participante*.
- *Modelagem dinâmica*: A potencialidade das ferramentas da modelagem dinâmica é muito grande, incluindo até aspectos funcionais. Isto permite que alguns elementos possam ser modelados duas vezes: no modelo dinâmico e no modelo funcional. Por exemplo, os parâmetros dos eventos dos diagramas de fluxo de eventos correspondem aos fluxos de dados nos DFDs, sendo que nos primeiros já foi mostrado o fluxo entre as classes e o DFD apenas mostra este fluxo ao nível das funções.
- *Herança*: A herança é recomendada pelos autores mas não é tratada explicitamente nos modelos dinâmico e funcional. As atividades, ações e funções poderiam ser padronizadas usando herança nestes modelos. Por exemplo, a função de criar uma ocorrência de *Pessoa* tanto como *Receptor Chamada Trabalhos* e *Convidado* não é fatorizada e é representada repetidamente para cada subclasse (vide figura 1 e anexo 4). Esta situação é mais marcada nos DFDs, que por construção são ortogonais às hierarquias de herança.
- *Redefinição de atributos e funções*: A técnica não apresenta nenhuma forma de renomear ou redefinir atributos e funções. Isto é consequência do indicado acima: ausência de herança nos modelos dinâmico e funcional. Por exemplo, seria preciso apenas renomear o atributo *data envio convite* da classe *Convidado* para a subclasse *Convidado Prioritário* (vide figura 1).

- *Valores “default”*: Os valores *default* de alguns atributos só aparecem na descrição das funções ou nos diagramas de estados, porém parece mais razoável definir tais valores em uma descrição separada dos atributos.
- *Troca de mensagens*: Ao usar DFDs para a modelagem funcional, não aparece então a necessidade de especificar troca de mensagens entre as classes. Esta troca só aparece nas fases seguintes do desenvolvimento segundo a metodologia OMT.
- *Alocação de funções*: O uso do DFD com particionamento funcional produz alguns problemas: 1) alguns DFDs podem apresentar funções extremamente difíceis de alocar nos objetos, particularmente se a decomposição é funcional; e 2) algumas funções podem acessar vários objetos, violando com isso o princípio de encapsulamento, isto implica realizar um particionamento posterior dos procedimentos destas funções para poder alocar as funções de cada objeto. Por exemplo, a função *incluir Artigo* (vide DFD 3 do anexo 3 e a descrição da função no anexo 4) possui um procedimento específico para *incluir Autor*. Os autores não indicam um critério que poderia ser útil para deter a explosão dos DFDs: decompor até obter funções aplicadas em apenas um objeto.
- *Funções de classes e objetos*: A técnica de análise OMT não faz distinção explícita entre funções que fazem parte das classes e aquelas que fazem parte dos objetos. Por exemplo, assume-se que correspondem a funções de classes, entre outras, *incluir Convidado* e *registrar recepção Resposta Convite* do DFD 1 do anexo 3, dado que criam ocorrências das classes respectivas (indicados pela ponta de seta maior).
- *Interdependência nos modelos*: Existe uma multiplicidade de referências entre os diagramas das diferentes modelagens o que implica em uma grande interdependência entre os distintos modelos. Esta interdependência ocasiona múltiplas modificações em outros diagramas ao ter que modificar algum elemento em um deles. Isto é particularmente notório entre os diagramas de fluxo de eventos, de estados, de fluxo de dados e a descrição de funções. Por exemplo, o acréscimo de um parâmetro no evento *classificação (Participante, Tipo)* no diagrama de fluxo de eventos da figura 2, implica realizar esta modificação no diagrama de

estados da figura 3, no fluxo de dados no DFD de alto nível da figura 5, no fluxo de dados no DFD 2 do anexo 3, e nos parâmetros e procedimento da descrição da função *classificar Participante* do anexo 4.

Ao considerar o aspecto da especificação e representação da análise, pode ser indicada a seguinte observação em relação à técnica de Rumbaugh et al.:

- *Notação do modelo de objetos*: A notação utilizada no modelo de objetos é muito pouco comum, o que pode confundir. Parece mais razoável utilizar algumas das notações próprias das extensões do modelo entidade-relacionamento.

Outras observações à proposta de Rumbaugh et al. podem ser encontradas em [BUS 93]. Entre as carências principais estão:

- localização de métodos ou funções nas classes
- reusabilidade

## ANEXO 1 - DICIONÁRIO DE DADOS DAS CLASSES

*Artigo* - Um trabalho submetido à conferência por um ou mais autores.

*Artigo Aceito* - Um artigo submetido à conferência e aprovado pela Comissão de Programa para ser apresentado em alguma sessão.

*Artigo Rejeitado* - Um artigo submetido à conferência não aceito pela Comissão de Programa para ser apresentado.

*Autor* - Uma pessoa que submete um ou mais artigos à conferência.

*Avaliador* - Uma pessoa especialista em um assunto técnico que avalia artigos submetidos à conferência.

*Carta Intenção* - Uma carta que envia um receptor de chamada de trabalhos em que indica sua intenção ou não de participar da conferência.

*Convidado* - Uma pessoa que recebe um convite para a conferência. Pode ser um autor de artigo aceito ou rejeitado, avaliador de artigos, moderador de sessões, membro de grupo ou algum representante nacional.

*Convidado Prioritário* - Uma pessoa que recebe um convite prioritário para comparecer à conferência. Pode ser membro de grupos de trabalhos ou algum representante nacional.

*Membro Grupo* - Uma pessoa que faz parte de um grupo de trabalho da IFIP associado à conferência.

*Moderador* - Uma pessoa que modera as sessões que fazem parte do programa da conferência.

*Receptor Chamada Trabalhos* - Uma pessoa que trabalha na área de pesquisa da conferência de trabalho. Recebe a chamada de trabalhos porque é um potencial autor de um ou mais artigos.

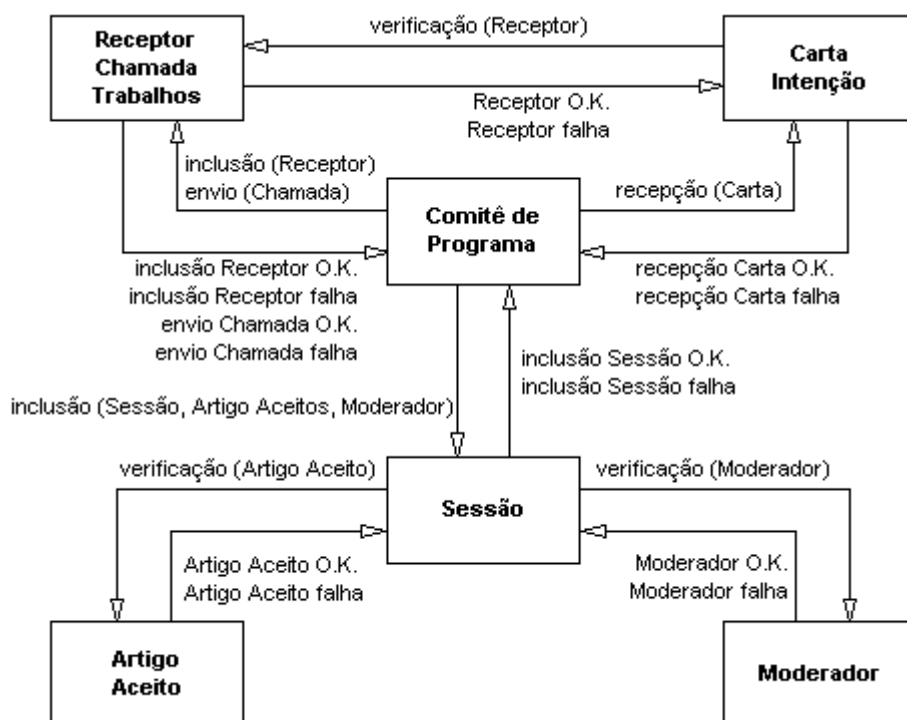
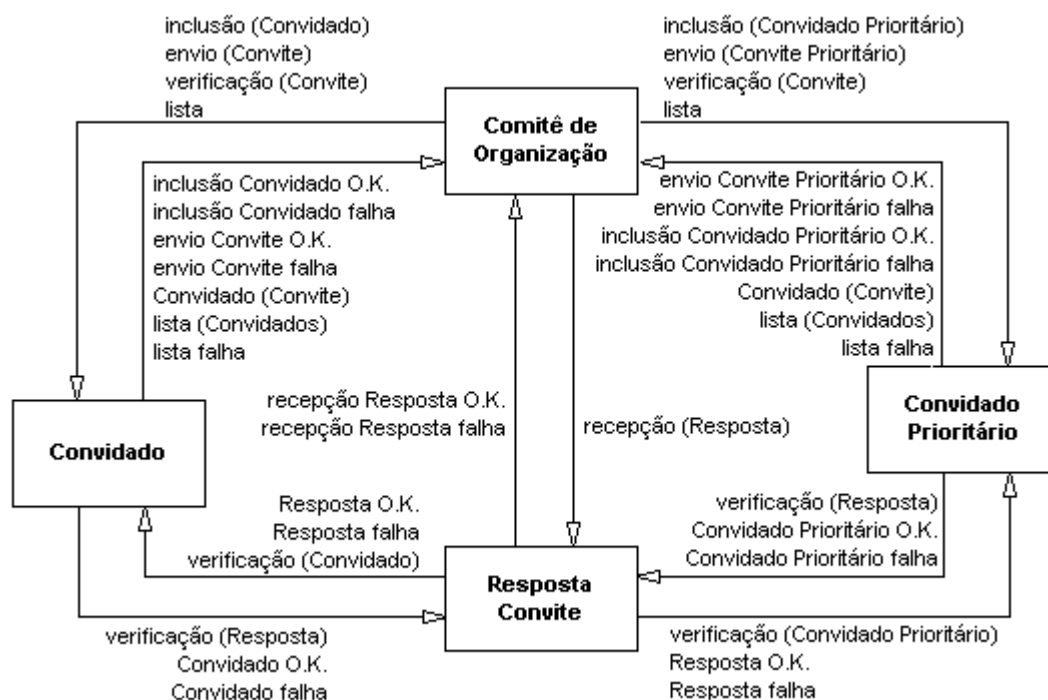
*Relatório* - Uma avaliação realizada por algum avaliador sobre um artigo determinado.

*Representante Nacional* - Uma pessoa que desempenha algum cargo vinculado à área de pesquisa no país em que é realizada a conferência.

*Resposta Convite* - Uma carta enviada por uma pessoa que recebe um convite. Nela expressa a aceitação ou negativa para comparecer à conferência.

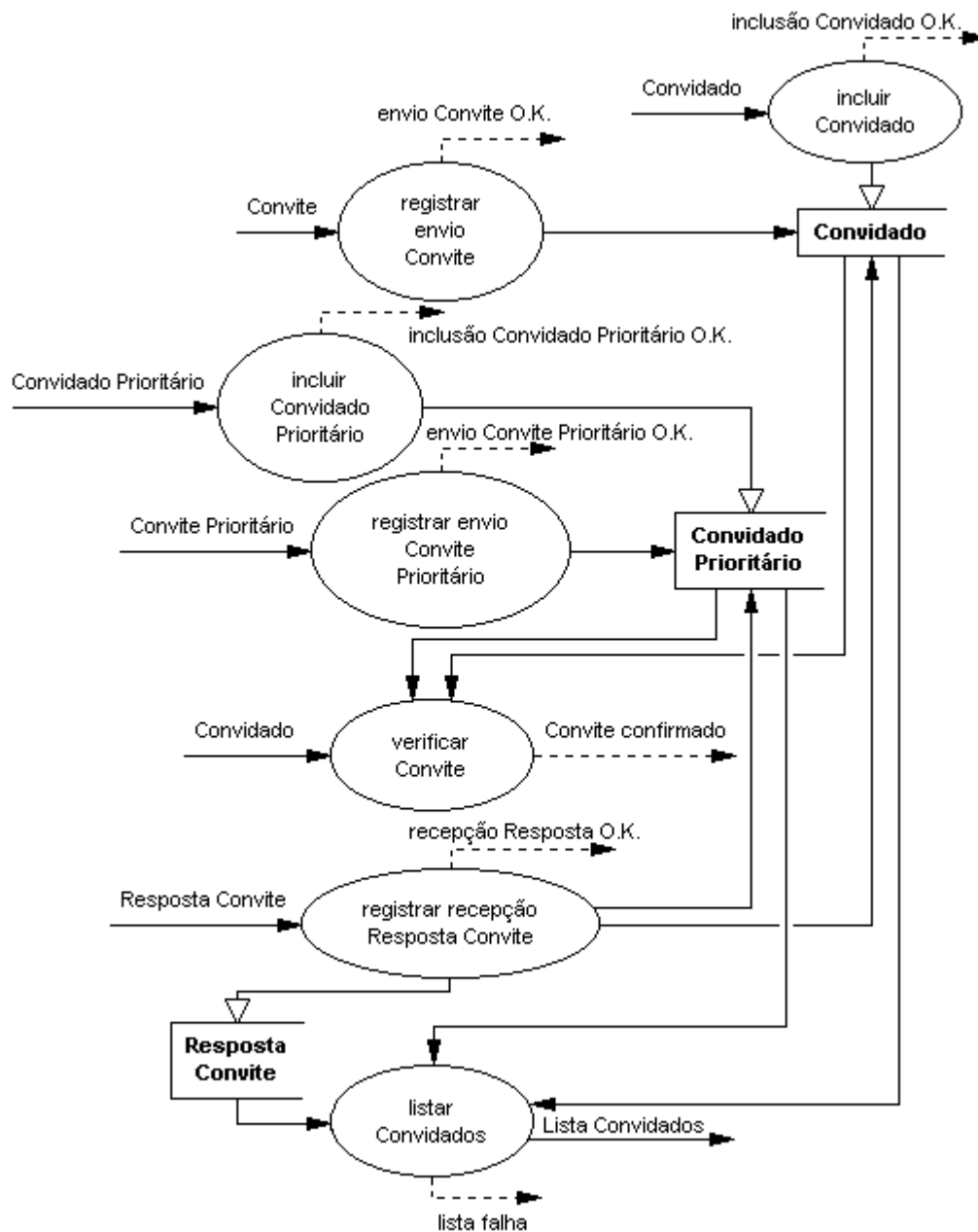
*Sessão* - Um agrupamento por assunto de artigos a ser apresentados na conferência. Possui data e horário específicos.

## ANEXO 2 - DIAGRAMAS DE FLUXO DE EVENTOS

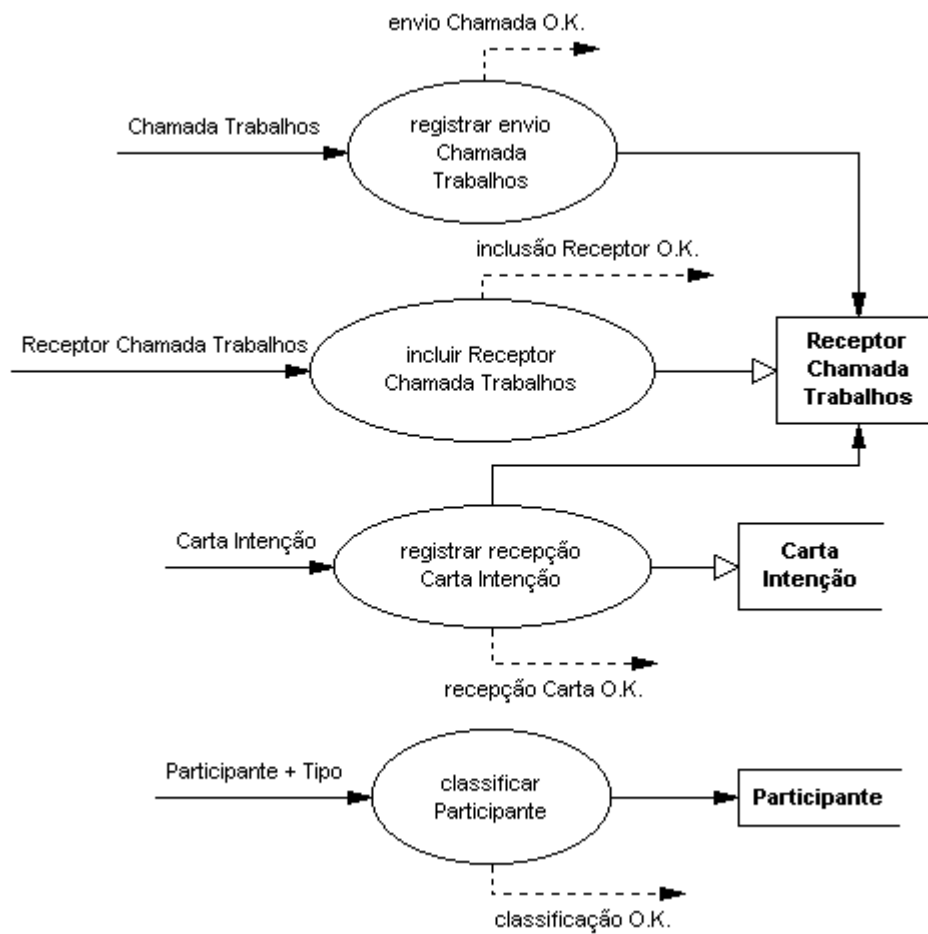




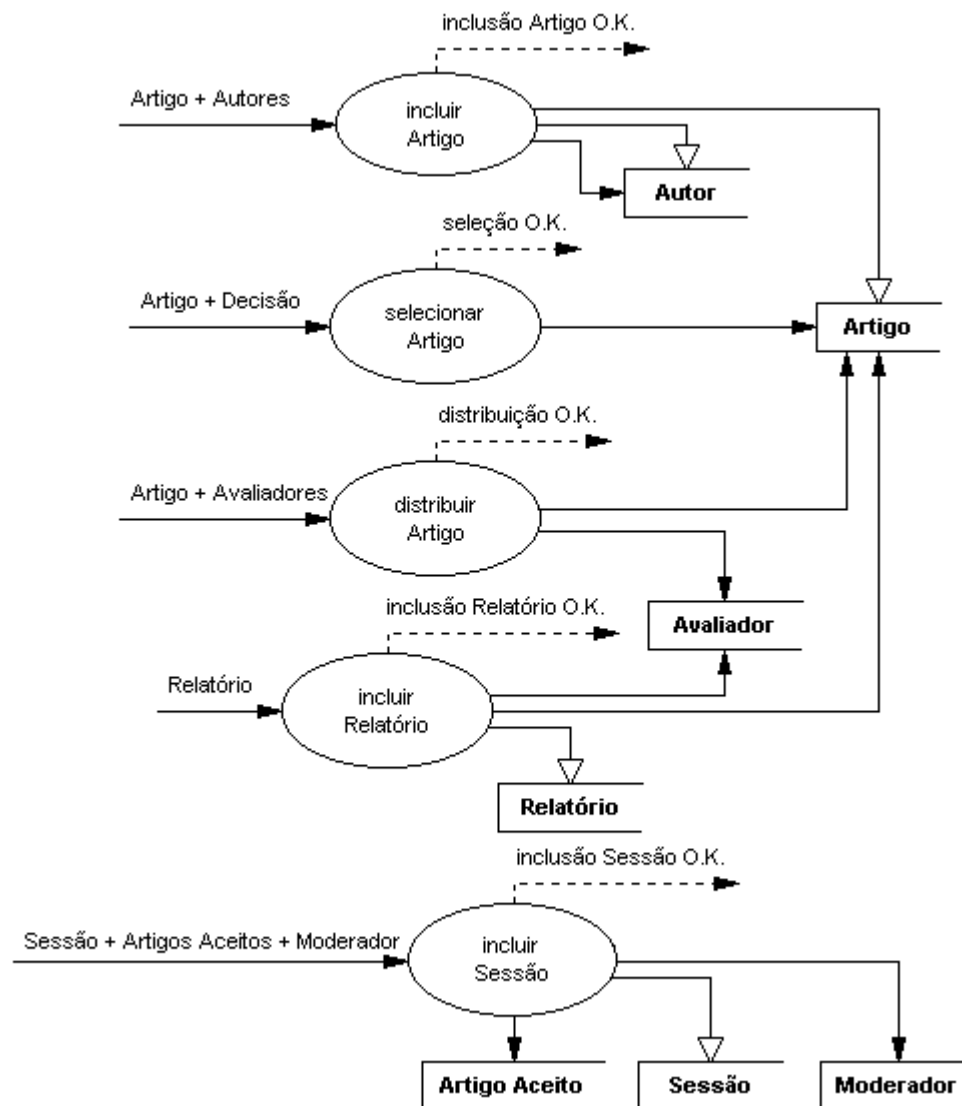
## ANEXO 3 - DIAGRAMA DE FLUXO DE DADOS DA MODELAGEM FUNCIONAL



DFD 1 Administrar Convidado



**DFD 2** Administrar Receptor Chamada Trabalhos



DFD 3 Administrar Artigo

## ANEXO 4 - DESCRIÇÃO DAS FUNÇÕES DA MODELAGEM FUNCIONAL

classificar Participante (Participante, Tipo) → papel, mensagem  
 verificar o valor de Tipo  
 se o valor está O.K.  
     se o papel é nulo  
         estabelecer papel como o valor de Tipo  
         enviar mensagem classificação O.K.  
     senão  
         enviar mensagem classificação falha  
 senão  
     enviar mensagem classificação falha

distribuir Artigo (Artigo, Avaliadores) → estado, mensagem  
 se o estado é recebido  
     enquanto houver Avaliadores do Artigo ou Avaliador.assunto não é equivalente  
     ao assunto  
     se Avaliador.assunto não é equivalente a assunto  
         enviar mensagem distribuição falha  
     senão  
         estabelecer estado como em avaliação  
         associar com Avaliadores  
         enviar mensagem distribuição O.K.  
 senão  
     enviar mensagem distribuição falha

incluir Artigo (Artigo, Autores) → Artigo, Autores, mensagem  
 verificar os valores de título, assunto, data envio artigo e a associações com Autores  
 se os valores estão O.K.  
     enquanto houver Autores do Artigo  
     se a associação não está O.K.  
         verifica os valores de nome, instituição, endereço  
         se os valores estão O.K.  
             criar novo Autor  
             enviar mensagem inclusão Autor O.K.  
         senão  
             enviar mensagem inclusão Autor falha  
     se todas as associações estão O.K.  
         criar novo Artigo  
         estabelecer estado como recebido  
         associar com Autores  
     senão  
         enviar mensagem inclusão Artigo falha  
 senão  
     enviar mensagem inclusão Artigo falha

incluir Convidado (Convidado) → Convidado, mensagem  
 verificar os valores de nome, instituição, endereço  
 se os valores estão O.K.

criar novo Convidado  
 estabelecer data envio convite como nula  
 enviar mensagem inclusão Convidado O.K.  
 senão  
 enviar mensagem inclusão Convidado falha

incluir Convidado Prioritário (Convidado Prioritário) → Convidado Prioritário, mensagem  
 verificar os valores de nome, instituição, endereço  
 se os valores estão O.K.  
 criar novo Convidado Prioritário  
 estabelecer data envio convite prioritário como nula  
 enviar mensagem inclusão Convidado Prioritário O.K.  
 senão  
 enviar mensagem inclusão Convidado Prioritário falha

incluir Receptor Chamada Trabalhos (Receptor Chamada Trabalhos) → Receptor Chamada  
 Trabalhos, mensagem  
 verificar os valores de nome, instituição, endereço  
 se os valores estão O.K.  
 criar novo Receptor Chamada Trabalhos  
 estabelecer data envio chamada como nula  
 enviar mensagem inclusão Receptor O.K.  
 senão  
 enviar mensagem inclusão Receptor falha

incluir Relatório (Relatório) → Relatório, mensagem  
 verificar o valor de avaliação e as associações com Avaliador, Artigo  
 se o valor e as associações estão O.K.  
 criar novo Relatório  
 associar com Avaliador, Artigo  
 enviar mensagem inclusão Relatório O.K.  
 senão  
 enviar mensagem inclusão Relatório falha

incluir Sessão (Sessão, Artigos Aceitos, Moderador) → Sessão, mensagem  
 verificar os valores de assunto, data, horário  
 se os valores estão O.K.  
     criar nova Sessão  
     enquanto o número de artigos seja insuficiente e houver Artigos Aceitos  
         se Artigo Aceito.assunto é equivalente a assunto  
             associar com Artigo Aceito  
     se Moderador.assunto é equivalente a assunto  
         associar com Moderador  
         enviar mensagem inclusão Sessão O.K.  
 senão  
     enviar mensagem inclusão Sessão falha  
 senão  
     enviar mensagem inclusão Sessão falha

listar Convidados → Lista Convidados, mensagem  
 enquanto houver Convidados ou Convidados Prioritários  
     se existe Resposta Convite associada e a resposta é aceitação  
         incluir Convidado ou Convidado Prioritário na Lista Convidados  
 se a Lista Convidados é vazia  
     enviar mensagem lista falha

registrar envio Chamada Trabalhos (Chamada Trabalhos) → data envio chamada, mensagem  
 se a data envio chamada é nula  
     estabelecer data envio chamada como o valor da data de hoje  
     enviar mensagem envio Chamada O.K.  
 senão  
     enviar mensagem envio Chamada falha

registrar envio Convite (Convite) → data envio convite, mensagem  
 se a data envio convite é nula  
     estabelecer data envio convite como o valor da data de hoje  
     enviar mensagem envio Convite O.K.  
 senão  
     enviar mensagem envio Convite falha

registrar envio Convite Prioritário (Convite Prioritário) → data envio convite prioritário,  
 mensagem  
 se a data envio convite prioritário é nula  
     estabelecer data envio convite prioritário como o valor da data de hoje  
     enviar mensagem envio Convite Prioritário O.K.  
 senão  
     enviar mensagem envio Convite Prioritário falha

registrar recepção Carta Intenção (Carta Intenção) → Carta Intenção, mensagem  
 verificar os valores de data envio intenção, intenção e a associação com Receptor  
 Chamada Trabalhos  
 se os valores e a associação estão O.K.  
     criar nova Carta Intenção  
     associar com Receptor Chamada Trabalhos  
     enviar mensagem recepção Carta O.K.  
 senão  
     enviar mensagem recepção Carta falha

registrar recepção Resposta Convite (Resposta Convite) → Resposta Convite, mensagem  
 verificar os valores de data envio resposta, resposta e a associação com Convidado ou  
 Convidado Prioritário  
 se os valores e a associação estão O.K.  
     criar nova Resposta Convite  
     associar com Convidado ou Convidado Prioritário  
     enviar mensagem recepção Resposta O.K.  
 senão  
     enviar mensagem recepção Resposta falha

selecionar Artigo (Artigo, Decisão) → estado, mensagem  
 verificar o valor de Decisão  
 se o valor está O.K.  
     se o estado é em avaliação  
         estabelecer estado como o valor de Decisão  
         enviar mensagem seleção O.K.  
     senão  
         enviar mensagem seleção falha  
 senão  
     enviar mensagem seleção falha

verificar Convite (Convidado) → Convite confirmado  
 se a data envio convite ou data envio convite prioritário é nula  
     estabelecer Convite confirmado como falso  
 senão  
     estabelecer Convite confirmado como verdadeiro

## REFERÊNCIAS BIBLIOGRÁFICAS

- [BUS 93] BUSTOS, Guillermo. **Estudo Comparativo de Técnicas de Análise Orientada a Objetos**. Trabalho Individual I nº 317, Porto Alegre: CPGCC-UFRGS, Mar. 1993.
- [CLY 92] CLYDE, Stephen; EMBLEY, David; WOODFIELD, Scott. Tunable Formalism in Object-Oriented Systems Analysis: Meeting the Needs of Both Theoreticians and Practitioners. **ACM SIGPLAN Notices**, New York, v.27, n.10, p.452-465, Oct. 1992. Trabalho apresentado na Annual Conference on Object-Oriented Programming, Systems, Languages, and Applications - OOPSLA '92, 7., 1992, Vancouver.
- [EMB 92] EMBLEY, David; KURTZ, Barry; WOODFIELD, Scott. **Object-Oriented System Analysis: A Model-Driven Approach**. Englewood Cliffs: Prentice-Hall, 1992. 302p.
- [HAR 87] HAREL, David. Statecharts: A Visual Formalism for Complex Systems. **Science of Computer Programming**, Amsterdam, v.8, n.3, Jun. 1987.
- [IFI 82] IFIP WG 8.1 WORKING CONFERENCE ON COMPARATIVE REVIEW OF INFORMATION SYSTEMS DESIGN METHODOLOGIES, 1982, Noordwijkerhout. **Proceedings...** Amsterdam: North-Holland, 1982.
- [OLL 82] OLLE, T. William. Comparative Review of Information Systems Design Methodologies: Problem Definition. In: IFIP WG 8.1 WORKING CONFERENCE ON COMPARATIVE REVIEW OF INFORMATION SYSTEMS DESIGN METHODOLOGIES, 1982, Noordwijkerhout. **Proceedings...** Amsterdam: North-Holland, 1982. p.8-9.
- [RUM 91] RUMBAUGH, James; BLAHA, Michael; PREMERLANI, William; EDDY, Frederick; LORENSEN, William. **Object-Oriented Modeling and Design**. Englewood Cliffs: Prentice-Hall, 1991. 500p.



- [SHL 89] SHLAER, Sally; MELLOR, Stephen. An Object-Oriented Approach to Domain Analysis. **ACM SIGSOFT Software Engineering Notes**, New York, v.14, n.5, p.66-77, Jul. 1989.
- [SHL 92] SHLAER, Sally; MELLOR, Stephen. **Object Life Cycles: Modeling the World in States**. Englewood Cliffs: Yourdon Press, 1992.