

COMPARAÇÃO DAS METODOLOGIAS ORIENTADAS A OBJETOS OMT E FUSION, CONSIDERANDO AS FASES DE ANÁLISE E PROJETO.

Luiz Carlos Camargo *

Resumo

O artigo propõe uma comparação através de estudo de caso “Cartão-Ponto Eletrônico” entre as metodologias OMT e FUSION. A comparação terá ênfase somente nas fases de análise e projeto, excluindo qualquer hipótese de implementação. Com objetivo principal de apresentar exemplos que direcionem a análise de complexidade, praticidade e notação de cada metodologia, atém-se aos procedimentos utilizados em estudo de caso através da modelagem.

Abstract

The article proposes a comparison through the case study “Electronic Point-Card” among the OMT and FUSION methodologies. The comparison will have emphasis only in the analysis and project stages, excluding any hypothesis of implementation. The principal aim is showing examples that direct the analysis of complexity, practicality and notation of each methodology, it pays attention to the procedures used in case study through the modeling.

Introdução

O artigo apresenta um estudo de caso. Reportar-se-á às fases de desenvolvimento propostas num processo de modelagem, levando em conta o projeto e sua análise.

Antes, porém, de iniciar a investigação propriamente dita, far-se-á uma breve abordagem comparativa entre as metodologias conhecidas e, em seguida, organizar-se-ão os diversos passos que disponibilizam a construção de modelos, notações e gráficos de interação.

Trata-se de um estudo do “Cartão-Ponto Eletrônico”. Como suporte do estudo, utilizar-se-á o modelo de análise e projeto das metodologias OMT e FUSION.

1. Estudo de caso: relógio-ponto eletrônico

1.1 Descrição do domínio do problema

Toma-se como parâmetro uma determinada organização que possui um sistema informatizado de Folha de Pagamento, cujo apontamento de Cartões-Ponto ainda seja efetuado de forma manual, na suposição de que esteja causando transtornos para o Departamento Pessoal. Diante da situação, a organização solicita um projeto que automatize a coleta

* Mestrando em Ciência da Computação. Docente da UNIPAR

dos dados de início e término das atividades de seus funcionários.

O caso apresenta uma série de detalhes a serem considerados:

- A implantação de Relógio-Ponto Eletrônico;
- O modelo de Cartão-Magnético para a identificação do funcionário, contendo o nome e informações de matrícula;
- Uma estação PC com capacidade de armazenamento

de informações referentes aos funcionários, interligada com o Relógio-Ponto;

- Previsão de transações indispensáveis, como:

- Validação do Cartão-Ponto a cada interação com relógio eletrônico;
- Atualização do arquivo ponto na estação PC;
- Armazenamento quinzenal dos dados da estação PC no banco de dados da Folha de Pagamento.

1.2 Aspectos introdutórios

<u>OMT</u>	<u>FUSION</u>
<p>Fases de desenvolvimento: Análise, projeto de sistemas, projeto de objetos e implantação. A metodologia conceitua objetos, classes e atributos. Faz analogias com a abordagem procedural. A reutilização de código é tratada subjetivamente.</p>	<p>Fases de desenvolvimento: Análise, projeto e implementação. A metodologia se intitula como de segunda geração de desenvolvimento orientado a objetos, e deixa explícito que Rumbaugh, Booch, Wirfs-Brock, Coad, entre outros, são de primeira geração. A questão da reutilização e reengenharia de sistemas é tratada com objetividade. Abordagem procedural não é citada.</p>

Quadro 1

1.3 Fase de análise

1.3.1 OMT

Na fase de análise o modelo aborda três aspectos dos objetos: a estrutura estática ou modelo de objetos, seqüência de interações ou modelo dinâmico e as transformações dos dados ou modelo funcional. [Rum 94].

A metodologia OMT prevê o levantamento dos requisitos abstraídos do mundo real, onde se encontra o domínio do problema. Propõe que seja feita uma análise criteriosa desses requisitos.

Seguindo essa idéia, a metodologia avança para a modelagem de objetos que precede o modelo dinâmico e modelo funcional, direcionando o modelador a seguir os recursos que são oferecidos.

1.3.2 FUSION

- Propõe desenvolver um modelo de objetos para o

domínio do problema.

- Determinar a interface do sistema.
 - Desenvolver um modelo de interfaces. No modelo de interfaces surgem dois modelos: o Modelo de Ciclo-de-Vida e o Modelo de Operações. [Col 96]
- Quanto aos requisitos, prevê um tratamento do mundo real, mas com um pequeno discurso inicial. Com a evolução e desenvolvimento da análise, cita pontos importantes relacionados aos requisitos.

2. Identificação das classes de objetos

Através dos requisitos apresentados, faz-se uma identificação das possíveis classes de objetos sem nenhuma avaliação sobre a sua real necessidade. Em seguida descartam-se aqueles aspectos que não correspondem ao domínio do problema: as classes redundantes e as classes de implementação é o que se pode visualizar na Figura 1.

Classes:

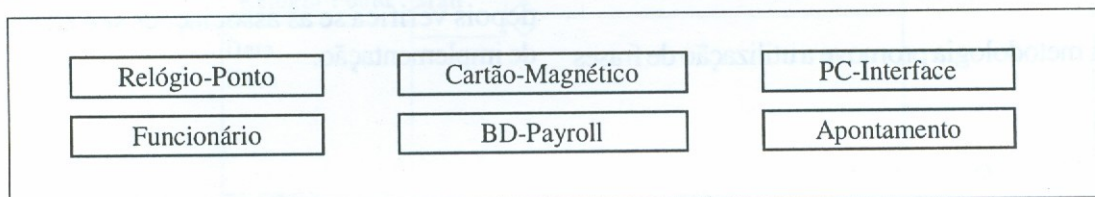


Figura 1: Classes válidas de objetos

2.1 Dicionário de dados.

A utilização do dicionário de dados descreve todas

as abrangências de cada classe dentro do contexto do sistema. A Figura 2 explica em traços gerais essa situação.

RELÓGIO-PONTO. Instalado em um ponto estratégico possibilita que os funcionários façam suas marcações de entrada e saída da empresa, através do Cartão-Magnético. Essas marcações servem de apontamento para alimentação do BD-Payroll.

CARTÃO-MAGNÉTICO. É individual e restrito a cada funcionário, contendo a matrícula como identificador individual. Quando lido o Cartão-Magnético, o apontamento de matrícula, data e hora ficarão disponíveis no PC-Interface para posteriormente ser consistido.

Figura 2: Exemplo de dicionário de dados.

2.2 FUSION

No desenvolvimento do modelo de objeto, a metodologia FUSION também faz uso de dicionário

de dados que é composto pelas classes de objetos, relacionamentos e invariantes obtidos através da lista refinada de classes e relacionamentos.

Nome	Categoria	Descrição
Funcionário	Classe	Sujeito que desenvolve alguma atividade remunerada na empresa.
Possui	Relacionamento	Funcionário possui Cartão-Magnético, que utiliza para registrar suas saídas e entradas na empresa.
Cartão-Magnético	Classe	Dispositivo que serve para iniciar processo de apontamento temporário.
Relógio-Ponto está ligado	Invariante	O Relógio-Ponto sempre deve estar ligado

Figura 3: Exemplo de dicionário de dados

3. Identificação de associações

A metodologia promove a utilização de frases

verbais, afim de reunir o maior número de associações, depois verifica se as associações são adequadas ou de implementação.

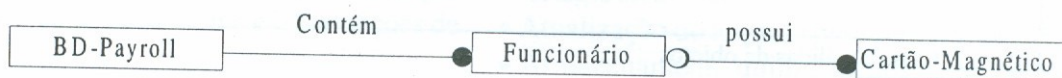


Figura 4: Exemplo de associações

4. Identificação de atributos

A metodologia aqui em questão conceitua o que vem a ser atributos, mas pede que se retratem apenas aqueles que se relacionam diretamente com o domínio do problema.

Além do mais, mostra ainda alguns critérios de eliminação de atributos desnecessários.

Nas figuras 5, 6 e 7 que seguem, encontram-se diagramas de objetos incorporando atributos e herança.

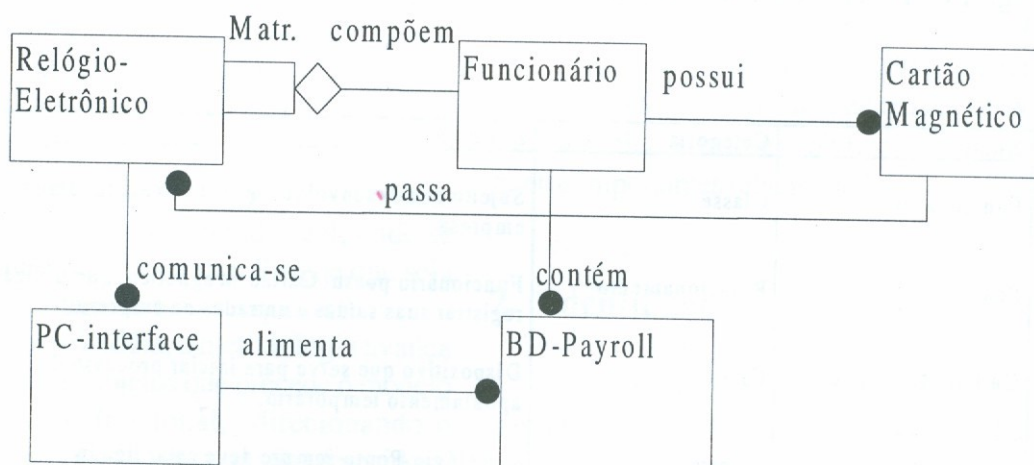


Figura 5: Diagrama inicial de objetos

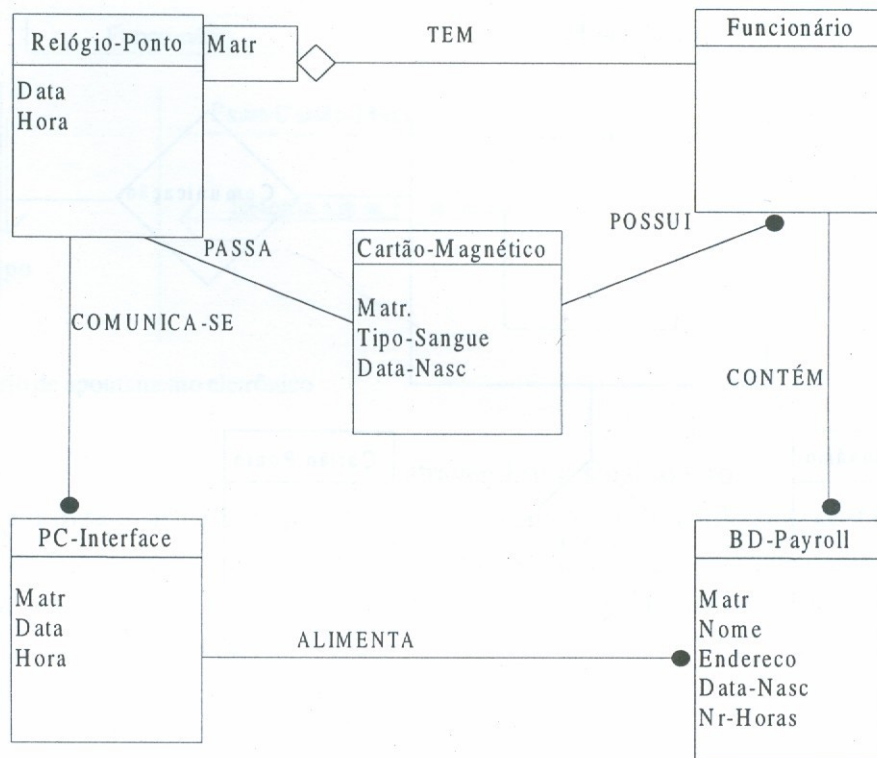


Figura 6: Diagrama de objetos com atributos

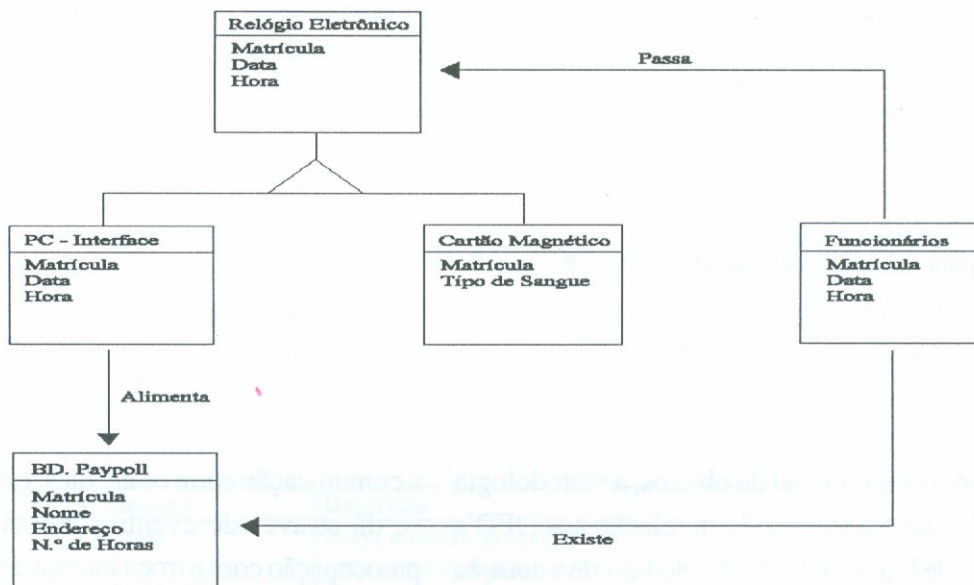


Figura 7: Diagrama de objeto com atributos e herança

4.1 FUSION

Na metodologia FUSION, os relaciona-

mentos surgem nos modelos de objetos, e esses modelos lembram (MER - Modelo Entidade Relacionamento) pela sua notação.

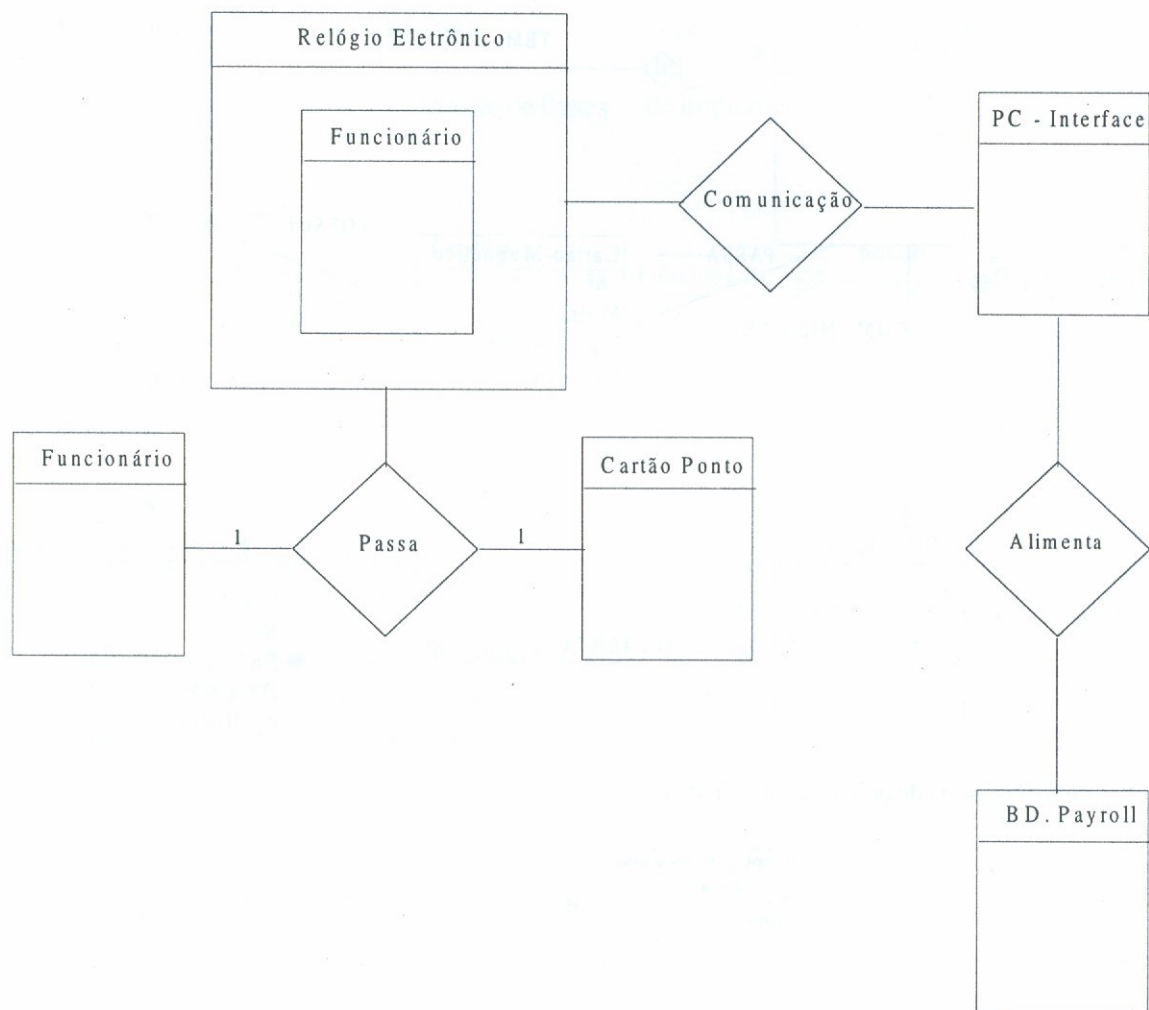


Figura 8: Modelo de objeto para classe Relógio-Ponto

No modelo inicial de objetos, a metodologia FUSION chama à atenção com relação aos DFD's (Diagrama de Fluxo de Dados) pelo fato de a notação lembrar a modelagem estruturada. A fim de evitar mal entendidos, deve-se tomar bastante cuidado para com tal interpretação.

A metodologia modela a interface do sistema através de cenários e dicionário de dados. Os cenários têm por objetivo mostrar as operações do sistema e

a comunicação entre os agentes, e essa comunicação se dá através de eventos. Nessa fase não existe preocupação com a troca interna de mensagens entre os objetos.

A figura 9 que vem na seqüência mostra um cenário de apontamento eletrônico, no qual as setas contínuas demonstram as operações do sistema e as setas tracejadas demonstram as operações dos eventos de saída do sistema.

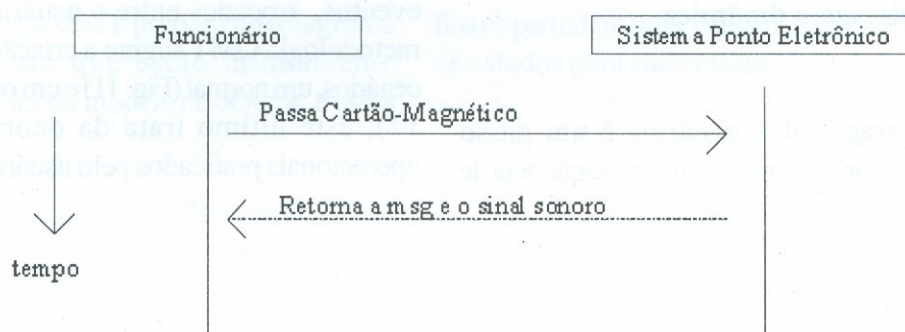


Figura 09: Cenário de apontamento eletrônico

Descrição:

O usuário que utiliza o sistema de ponto-eletrônico passa seu cartão-magnético, contendo sua matrícula. O sistema valida e armazena o apontamento, retornando ao usuário com uma informação de “OK”

através de um sinal sonoro.

A metodologia utilizada constantemente é o dicionário de dados para representar os eventos e operações entre os agentes do sistema.

Nome	Categoria	Agente	Descrição
Passar-Ponto	Evento	Funcionário	Passa o cartão-magnético para efetuar seu apontamento
Consistência dos dados de apontamento	Operação Sistema	Encarregado	Quinzenalmente faz-se a consistência dos dados que estão disponíveis no PC-Interface coletados pelo relógio-ponto
Descarregar no BD-Payroll	Operação Sistema	Encarregado	Alimentação do BD-Payroll pelos dados de apontamento consistidos

Figura 10: Dicionário de dados: Interface do sistema

Nome	Categoria	Descrição
Funcionário	Agente	Passa o cartão-magnético no relógio-eletrônico e aguarda o retorno do apontamento responsável pela consistência dos apontamentos
Encarregado	Agente	Responsável pela consistência e alimentação do BD-Payroll onde se utiliza dos recursos do sistema para efetuar as operações

Figura 11: Dicionário de dados: Agentes

4.2 OMT: Modelagem dinâmica

A preparação dos cenários é um passo importante. Os cenários constituem uma seqüência de

eventos, trocados entre o usuário e o sistema. A metodologia OMT sugere a criação de dois tipos de cenários, um normal (Fig. 11) e um outro anormal (Fig. 12); este último trata da ocorrência de erros operacionais praticados pelo usuário.

O relógio-eletrônico: Solicita ao usuário que passe o cartão-magnético

O relógio-eletrônico: Aceita o cartão-magnético e lê a matrícula

O relógio-eletrônico: Valida a matrícula através da interface com o PC-Interface e descarrega os dados coletados no PC-Interface.

O relógio-eletrônico: Emite uma mensagem de OK e um efeito sonoro.

Figura 12: Cenário normal do relógio-eletrônico

O relógio-eletrônico: Solicita ao usuário que passe o cartão-magnético

O relógio-eletrônico: Aceita o cartão-magnético e lê a matrícula.

O relógio-eletrônico: Emite um sinal sonoro e uma mensagem dizendo que a matrícula é inválida. contactar o responsável

Figura 13: Cenário anormal do relógio-eletrônico

Com relação aos eventos para os cenários, similar à metodologia FUSION. como se pode ver na figura que segue, a notação é

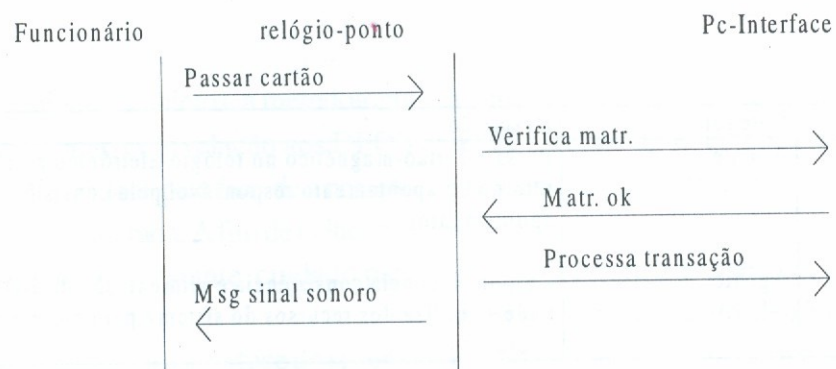


Figura 14: Diagrama de eventos para o cenário relógio-ponto

A metodologia OMT prepara um diagrama de fluxos de evento que estão intimamente relacionados às classes do modelo de objetos, e esses

fluxos permitem uma visão mais ampla dos diagramas de estados para cada classe.

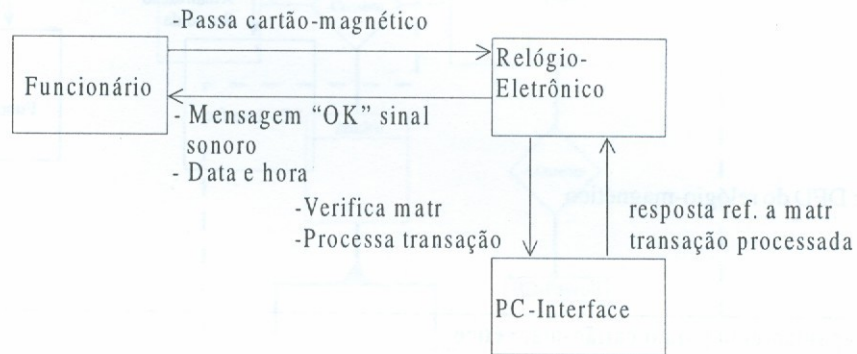


Figura 15: Diagrama de fluxo de eventos

Utiliza-se nessa instância um diagrama de fluxo de eventos que na realidade nada mais é do que o

DTE's - Diagrama de Transição de Estado.

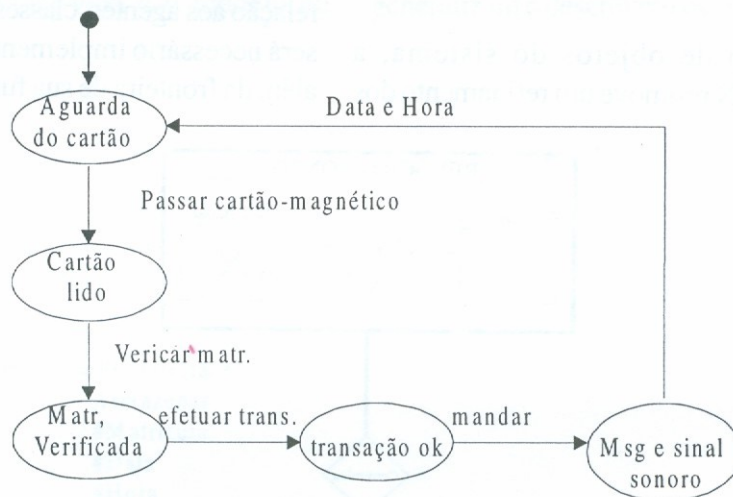


Figura 16: Diagrama de transição de estado para classe relógio-ponto

5. Modelagem funcional

No modelo funcional, a metodologia OMT aborda algumas ferramentas da análise estruturada,

como DFD - Diagrama de Fluxo de Dados, linguagem natural e até pseudocódigo para expressar as dependências de funções entre as diferentes classes de objetos.

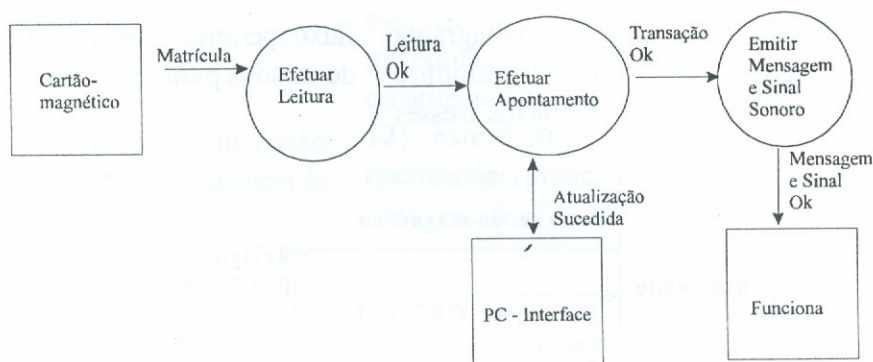


Figura 17: DFD do relógio-magnético

Fazer apontamento para o cartão-magnético

Se matr válida, atualize a base no PC-Interface com matr, data e hora

Se matr inválida, disponibilize uma mensagem "matr inválida".

Figura 18: Descrição da função efetuar apontamento

FUSION

No Modelo de objetos do sistema, a metodologia FUSION promove um refinamento dos

modelos afim de mostrar as fronteiras do sistema em relação aos agentes, classes ou relacionamentos. Não será necessário implementar aqueles que estiveram além da fronteira de sua funcionalidade [Col 96].

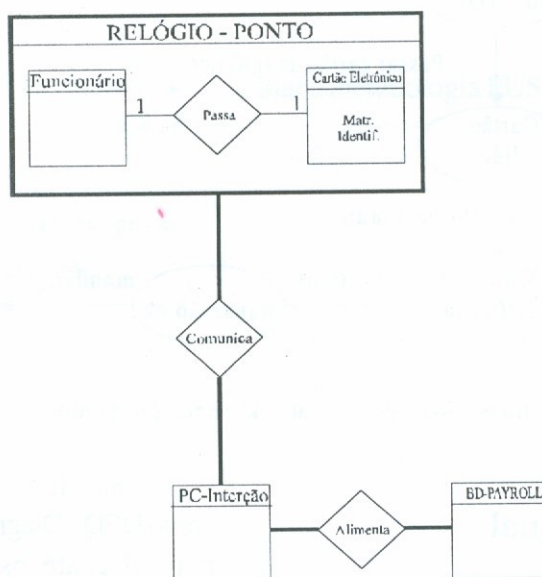


Figura 19: Modelo de objeto relógio-ponto eletrônico com identificador de matrícula do cartão-magnético.

Nesse modelo, o agente encarregado está fora da fronteira do sistema. O mesmo acontece com o relacionamento consiste.

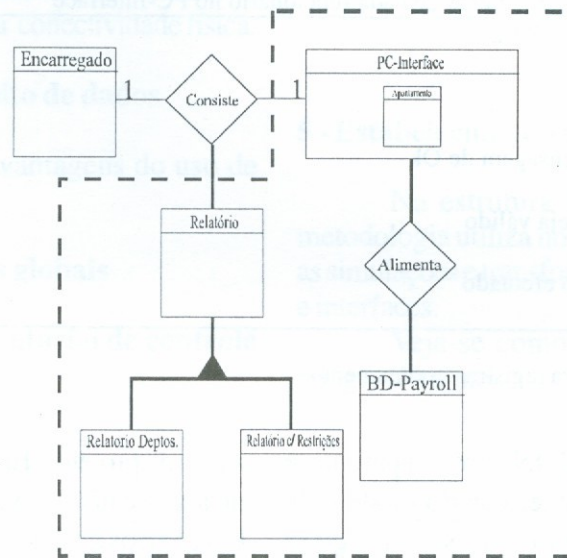


Figura 20: Modelo de objetos para consistência e alimentação do BD-Payroll

Na metodologia FUSION, encontramos dois modelos que compõem o modelo de interfaces. São eles: o modelo de ciclo-de-vida e o modelo de operações.

O modelo ciclo-de-vida utiliza expressões que generalizam e descrevem os cenários.

```
life cycle relógio-eletronico (leitura | apontamento)* || (estado)*
```

```
Leitura = Cartão-ponto.
```

```
Passar_cartão-ponto
```

```
(Verificar_matrícula #identificador_cartão-ponto)*
```

```
Encerrar_verificação
```

```
Apontamento = PC-Interface
```

```
Armazenar
```

```
#Matrícula
```

```
#Data
```

```
#Hora
```

```
Estado = (Estado_relógio-eletrônico, #msg_ok #sinal_sonoro)*
```

Figura 21: Ciclo-de-vida relógio-ponto

O modelo de operações utiliza-se de esquemas, e desenvolve cláusulas que direcionam as operações a um nível de detalhe bastante significativo,

possibilitando analisar o envolvimento e o resultado de cada operação.

Operação: Registrar-apontamento
Descrição: Registrar o apontamento de cada funcionário no PC-Interface
Leitura: Matrícula, Data e Hora
Troca:
Envio: sinal sonoro e mensagem de Ok
Assume: Cartão-ponto seja válido
Resultado: Apontamento efetuado

Figura 22: Esquema para registrar-apontamento

A metodologia FUSION abre um parêntese e sugere uma verificação criteriosa do modelo de análise antes de prosseguir para etapa seguinte que é

a fase de projetos. Entretanto considera que o modelo de análise não será perfeito, mas também não deve haver erros grotescos. Essa análise deve ser feita na intersecção de modelos.

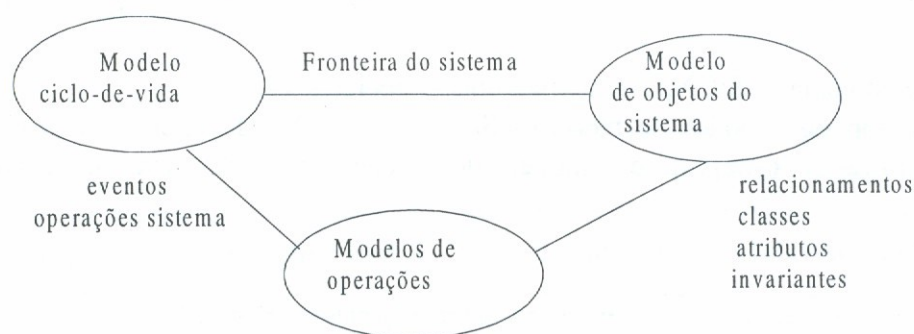


Figura 23: Verificação de intersecções entre modelos [Col 96]

6. Fase de projetos

6.1 OMT

A metodologia OMT utiliza dois sub-projetos: Projeto do sistema e projeto de objetos. No projeto de sistemas é proposto que o projetista divida o problema em diversos sub-sistemas para facilitar o enfoque e alocação de vários desenvolvedores em diferentes sistemas.

A partir daí o projeto de sistema tem um tratamento genérico, ou seja, o projeto pode ser utilizado em desenvolvimento orientado a objetos quanto em projetos convencionais.

Na divisão de sistemas em sub-sistemas, a metodologia OMT utiliza camadas para ordenar todo o pacote de aplicativos. Faz uso de partições para divisão vertical dos sistemas e sub-sistemas.

O uso de identificação de concorrência para verificar em que grau ela aparece entre os objetos e o processador(s).

A metodologia sugere algumas diretrizes. No escopo desse artigo serão apresentados apenas os pontos chave [Rum 94: 267 a 278].

1- A alocação de sub-sistemas a processadores e tarefas.

1.1 - Estimativa das necessidades de recursos

de *Hardware*.

- 1.2 - Equilíbrio *Hardware-Software*
- 1.3 - Alocação de tarefas ao processador
- 1.4 - A determinação da conectividade física.

2 - Gerenciamento de depósito de dados

2.1 - Vantagens e desvantagens do uso de banco de dados

3 - Manipulação de recursos globais

4 - Escolha de uma implementação de controle de Software

- 4.1 - Sistemas baseados em procedimentos;
- 4.2 - Sistemas baseados em eventos;
- 4.3 - Sistemas concorrentes;
- 4.4 - Controle interno.

5 - Estabelecimento de prioridades

Na estrutura arquitetônica do sistema, a metodologia utiliza novamente DFD's para modelar as simulações e transformações das estrutura de dados e interfaces.

Veja-se como seria a estrutura física do relógio eletrônico.

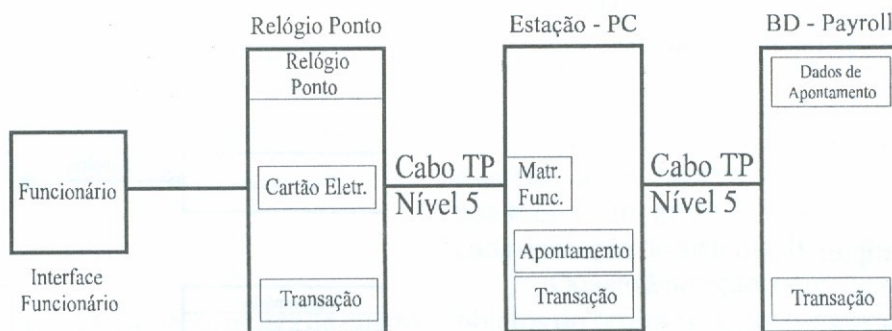


Figura 24: Arquitetura do sistema relógio-eletrônico.

7. Projeto de objetos

A OMT centraliza as atenções nos três modelos concebidos na fase de análise:

- Modelo de objetos;
- Modelo funcional;
- Modelo dinâmico.

Com isso estabelece as fases a serem executadas durante o projeto de objetos.

Combina os três modelos para obter operações sobre classes, onde descreve os processos executados pelos DFD's - Diagrama de Fluxo de Dados.

No projeto de algoritmos a especificação acontece com base no modelo funcional e fornece alguns conceitos ao projetista como:

- Escolher algoritmos que minimizem o custo

da implementação das operações;

- Selecionar estrutura de dados apropriadas aos algoritmos;
- Definir novas classes internas e operações quando necessárias;
- Atribuir responsabilidade por operações às classes apropriadas. [Rum 94]

A otimização do projeto prevê redundância de associações, replanejamento do processamento e armazenagem de atributos derivados.

A implementação do controle faz referência ao mundo dinâmico; para implementação, surgem três abordagens básicas:

- Sistemas baseados em procedimentos;
- Sistemas baseados em eventos;
- Utilização de tarefas concorrentes.

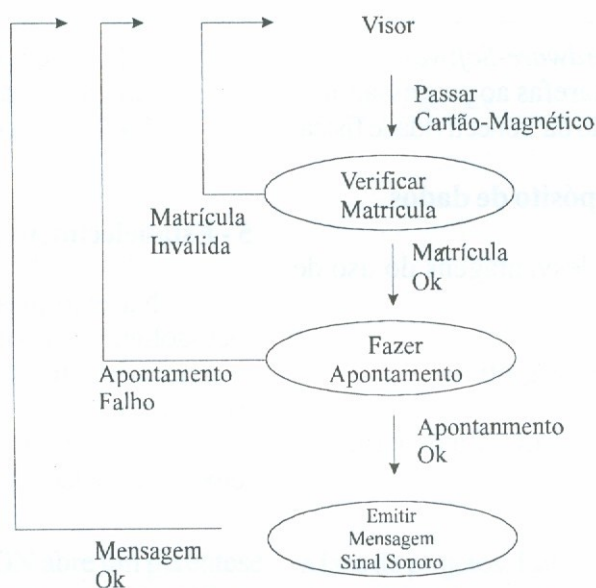


Figura 25: Controle do relógio ponto.

Pseudocódigo

Fazer sempre:

Exibir monitor;
 Ler cartão-magnético;
 Repetir;
 Verificar Matrícula;
 Até que Matrícula esteja ok;
 Repetir;
 Fazer apontamento;
 Até que apontamento ficar ok;
 Enviar msg e sinal sonoro.

No ajuste da herança, pede-se para reorganizar e ajustar as classes de objetos e de operações, abstraindo o comportamento comum dos grupos de classes de objetos e utilizando a delegação para compartilhar o comportamento do processo executado, toda vez que a semântica se mostrar inválida.

O projeto de associações é composto pelas associações de sentido e de dupla direção.

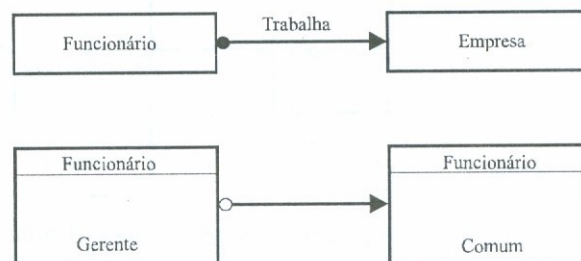


Figura 26: Implementação de associações de sentido único com utilização de ponteiros.

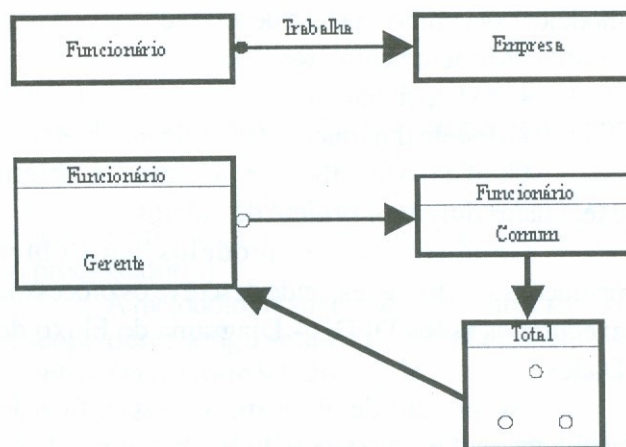


Figura 27: Implementação de associações de dupla direção com a utilização de ponteiros.

Notação de representação de objetos.



Figura 28: Representações alternativas para um atributo.

A metodologia OMT prevê o empacotamento físico envolvendo as seguintes questões:

- Ocultamento de informações internas para visão externa;
- Coerência de entidades;
- Construção de módulos físicos. [Rum 94].

À medida em que cada modelo construído na fase de análise é implementado na fase de projeto, pede-se uma documentação principalmente com relação às tomadas de decisões.

FUSION

A metodologia FUSION se utiliza de quatro modelos para desenvolver a fase de projetos. São eles: Grafo de interação; grafo de visibilidade;

descrição de classes e grafos de herança e ainda apresenta uma heurística de como desenvolver um bom projeto orientado a objetos.

8. Grafos de interação de objetos

Essa fase do projeto da qual a metodologia FUSION se utiliza, e tem por objetivo descrever a comunicação entre os objetos (troca de mensagens) de forma estruturada, afim de satisfazer as especificações funcionais do sistema, para isso constrói-se um grafo de interação de objetos para cada operação do sistema. [Col 96]

Quanto à notação, os retângulos representam objetos do projeto e as setas representam trocas de mensagens. Apenas o controlador terá uma seta chegando de qualquer lugar, menos de outro retângulo.

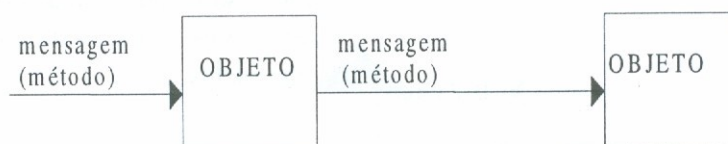


Figura 29: Representação da notação referente à troca de mensagens entre objetos.

O grafo de objetos mostra um seqüenciamento de mensagens através de números decimais de Dewey

e isso permite com que as mensagens sejam ativadas dentro de uma lógica seqüencial.

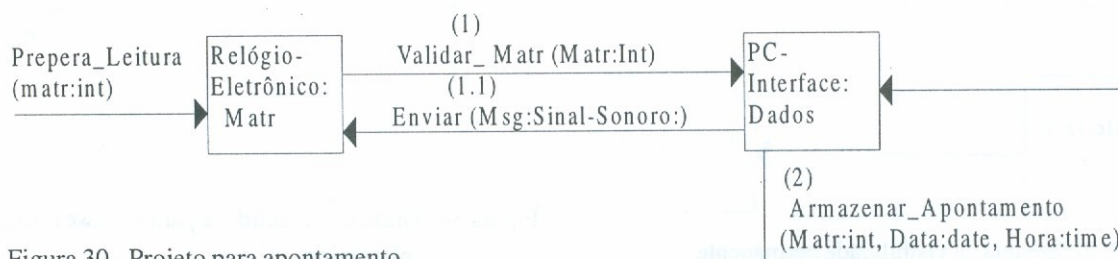


Figura 30 - Projeto para apontamento

8.1 Descrição:

OPERAÇÃO relógio-eletrônico: Prepara_leitura (Matrícula:int) (1)
 // prepara o relógio-eletrônico para receber cartão-magnético, a
 matrícula de estar ativa
 Se Matr. ativa retornar mensagem de Ok e sinal sonoro
 Caso contrário retornar mensagem de não OK
 OPERAÇÃO PC-Interface enviar (Msg ok:Char) (1.1)
 Retorna através de mensagens de consistência da matrícula
 MÉTODO PC-Interface: armazenar apontamento (Matr:Int, Hora:Time,Data:Date) (2)

8.2 Grafos visibilidade

Vem a completar a proposta dos grafos de interação, afim de decidir como será realizada a comunicação entre os objetos. A notação utiliza retângulos para demonstrar clientes e servidores, e setas para visibilidade. Embora de acordo com as características da visibilidade, os grafos sofrem pequenas alterações.

A visibilidade dinâmica implica em processar vários arquivos que estão no BD-Payroll; então o BD-Payroll será a referência. A visibilidade dinâmica acontece quando houver persistência nas chamadas de métodos.

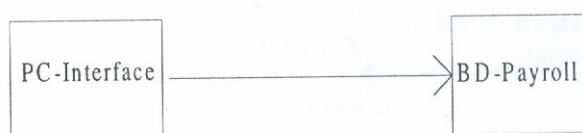


Figura 31: Referência de visibilidade dinâmica

Na questão de exclusividade permanente, os dados do Pc-Interface serão acessados exclusivamente pelo relógio-eletrônico.

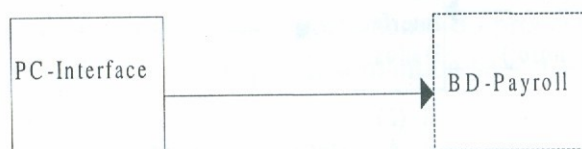


Figura32: Referência de visibilidade permanente

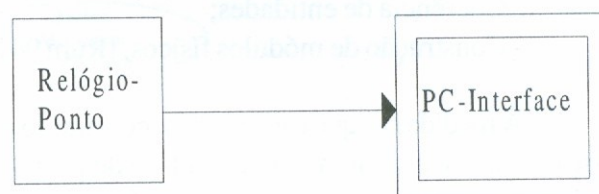


Figura 33: Referência de visibilidade exclusiva para um objeto

A questão cliente/servidor entre objetos acontece através da situação, o objeto servidor que recebe a mensagem de um objeto cliente, e essa mensagem pode ser exclusiva ou compartilhada.

A Metodologia FUSION, com grafos mais abrangentes, mostra as visibilidades entre clientes, servidores, setas, vinculações e mutabilidade das referências.

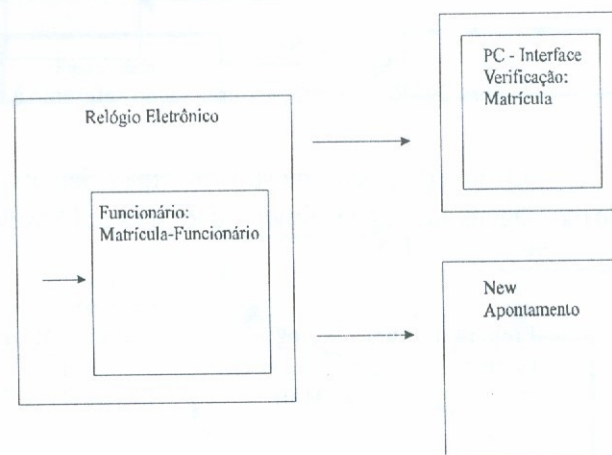


Figura 34: Gráfico de visibilidade para a classe relógio-eletrônico

9. Descrição de classes

Para se criar descrição de classes é necessária a utilização dos grafos de interação; grafos de visibilidade e o modelo de objeto do sistema. E nesse passo são estabelecidos para cada classe: os métodos, atributos de dados e atributos de objetos. Quanto à notação, veja a sintaxe [Col 96]

Class <nome-da-classe> [is a <nomes-de-superclasses>]

Aplicação da descrição de classes na classe PC-Interface.

Classe PC-Interface

Attribute Matr:Int;

Attribute Data:Int;

Attribute Hora:time;

Method Validar-Matr(): Bolol;

Method Emitir-Msg(): Msg:String;

Method Armazenar():Matr:Int,

Data:Date,Horas:Time

End Class

10. Grafo de herança

Ao se tratar de desenvolvimento orientado a objetos, a aplicação de grafos de herança é bastante útil, onde uma classe pode ser especialização de outra classe.

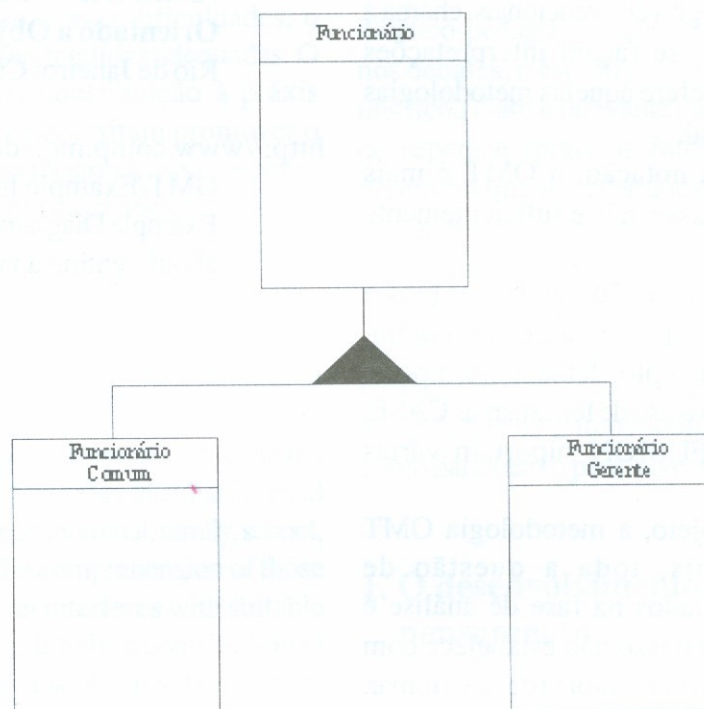


Figura 35: Grafo de herança para funcionário.

Conclusão

Diante dos passos realizados através do estudo de caso, o desenvolvimento acontece em duas fases: análise e projeto.

Na fase de análise, como se pode observar ao longo desse estudo, a metodologia OMT mostrou-se bastante desenvolvida, com abrangência significativa e apresentando um tipo de discurso que se mostra claro e objetivo, embora com muita frequência faça analogias com metodologias convencionais.

A metodologia FUSION tem um perfil mais depurado, enxuto e direto. Utiliza-se frequentemente de dicionário de dados e parte do pressuposto que o analista tenha uma noção do que vem ser desenvolvimento orientado a objetos.

Ao contrário da metodologia OMT, que faz analogias com metodologias convencionais, chama à atenção para que não se façam interpretações precipitadas no que se refere àquelas metodologias de modelagem tradicional.

Com relação à notação, a OMT é mais detalhada. Em função disso, não é suficientemente clara.

Já a FUSION não é tão detalhada, mas o entendimento que ele proporciona é claro. Porém, como todo desenvolvimento demanda tempo, é aconselhável que se faça o uso de ferramentas CASE. Nesse caso, OMT e FUSION suportam várias interpretações.

Na fase de projeto, a metodologia OMT deixa a desejar, pois, toda a questão de procedimentos cadenciados na fase de análise é deixada para trás, e com isso não estabelece com clareza os procedimentos que o projetista deve tomar, e novamente aborda alguns conceitos do desenvolvimento convencional. No entanto, apresenta heurísticas que podem ser úteis.

A metodologia FUSION mantém a mesma linha da fase de análise, no que diz respeito à organização do desenvolvimento do projeto, e utiliza

algumas notações importantes da fase anterior. Uma de suas características mais importantes é o fato de documentar os passos através de grafos, descrições e heranças.

Bibliografia

01. RUMBAUGH, James; BLAHA, Michall; PREMERLANI, William; EDDY, Frederick; LORENSEN, William. **Modelagem e Projetos Baseados em Objetos**. Rio de Janeiro: Campus, 1994, 652p
02. COLEMAN, Derek; PATRICK, Arnold; BODOFF, Stephanie; DOLLIN, Chris; GILCHRIS, Helena; HAYES, Fiona; JEREMAES, Paul. **Desenvolvimento Orientado a Objetos O Método Fusion**. Rio de Janeiro: Campus, 1996, 389p

<http://www.comp.mq.edu.au/courses/comp331/OMT/Example.html> OMT Object-Model Example Diagram. This example diagram is about renting a video at a video store.