

IT-173

UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE MATEMÁTICA E INFORMÁTICA

TRABALHO DE LICENCIATURA

TEMA: REENGENHARIA DO SISTEMA DE INFORMAÇÃO DE PESSOAL (SIP)
DIRECÇÃO DE RECURSOS HUMANOS (DRH)—MINED

Autor: Alfredo Jaze

Supervisor: Dr. Fernando J. Rafael Comolo

Maputo, 2005

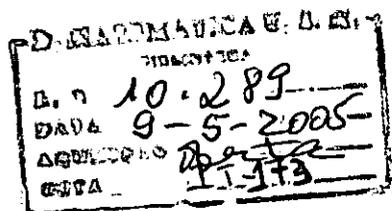
IT-173

Declaração

Declaro por minha honra que este trabalho é exclusivamente da minha própria investigação cuja finalidade é a obtenção do grau de Licenciatura em Informática, pela Universidade Eduardo Mondlane

(Alfredo Jaze)

Maputo, 2004



Agradecimentos

Durante a elaboração deste relatório, tive apoio de várias pessoas. Sem nenhuma discriminação, endereço a todos a minha gratidão.

Um agradecimento muito especial ao dr. Fernando Rafael Comolo, que aceitou supervisionar este trabalho, dando-me todo apoio necessário durante a realização deste estudo.

Outro agradecimento vai ao dr. Munguanaze que me insentivou a explorar a reengenharia de software como tema deste trabalho e forneceu-me o material básico sobre o mesmo.

O meu agradecimento é extensivo a todos docentes do Departamento de Matemática e Informática, pelo apoio que me deram durante a formação.

Quero terminar endereçando o meu agradecimento ao Sr. Sérgio Martins—Chefe do Departamento de Informação de Pessoal do MINED, que me forneceu os dados disponíveis sobre o sistema SIP.

Resumo

Este estudo tem como objectivo abordar conceitos sobre o processo de manutenção de software, especificamente no que respeita a reengenharia e engenharia reversa.

Para abordar adequadamente as técnicas de manutenção de software, deve se primeiramente considerar três conceitos dependentes: a existência de um processo de desenvolvimento de software, a presença de um sistema a ser analisado e a identificação de níveis de abstração.

Quando o sistema não é fácil de ser mantido, sendo porém de grande utilidade, ele deve ser reconstruído, partindo do sistema existente (via código fonte, interface ou ambiente), são abstraídas as suas funcionalidades e são construídos o modelo de análise e o projecto do software. Esse processo é denominado *reengenharia de software*.

A técnica tradicional, que avança progressivamente pelas fases do processo de desenvolvimento de software, é denominada *engenharia progressiva*. A execução dessa técnica consiste em partir de projectos independentes da implementação, que possuem altos níveis de abstração, indo em direcção a implementação física do sistema.

O processo inverso a engenharia progressiva, caracterizado pelas actividades retroativas do ciclo de vida, que partem de um baixo nível de abstração para um alto nível de abstração, é conhecido como *engenharia reversa*.

O processo de reengenharia de software é constituído por duas fases distintas. Na primeira, o software objecto de reconstrução é decomposto em partes, de modo a ser entendido. Na segunda, o software é reconstruído, na forma desejada, a partir do produto da primeira fase, sendo incluídos os ajustes que forem necessários.

Actualmente há uma preocupação muito grande em desenvolver sistemas utilizando tecnologias que se dispõem a complexidade, melhoram a manutenibilidade, permitem o reuso e reduzem o custo do ciclo de vida do software.

A programação orientada a objectos tem muitas vantagens porque possibilita construir sistemas altamente flexíveis, adaptáveis e extensíveis; possui uma colecção rica de mecanismos composicionais para formação de classes, instanciação de objectos, propriedades de herança, polimorfismo e ocultamento de informações, que estão presentes em linguagens de programação orientadas a objectos, as quais providênciam facilidades para criar sistemas que exibem um alto grau de reuso e facilidade de manutenção.

Objecto de reengenharia neste estudo é o Sistema de Informação de pessoal (SIP), que funciona na Direcção de Recursos Humanos (DRH) do Ministério da Educação (MINED). Esta reengenharia é realizada tendo em vista a orientação a objectos e é tomado como base a técnica de modelagem orientada a objecto—Object Modeling Technique (OMT)

Breve descrição da organização da DRH

A Direcção de Recursos Humanos, abreviadamente designada por DRH, é o órgão do Ministério da Educação (MINED) que regulamenta, planifica, coordena e monitora a execução da política de gestão, administração e controlo do pessoal do sector da Educação.

A Direcção de Recursos Humanos tem a sua estrutura geral assim definida:

- a) Director
- b) Departamento de Informação do Pessoal (DIP)
- c) Departamento de Administração de Pessoal (DAP)
- d) Departamento de Gestão e Normaçoão (DGN)
- e) Repartição de Administração Interna (RAI)

O Departamento de Informação de Pessoal (DIP) é uma unidade orgânica que procede a planificação e controlo da informação e diagnóstico da força de trabalho no MINED, estabelecendo objectivos e metas a serem alcançados e definir os meios necessários para os atingir.

Para o cumprimento das suas funções, o Departamento de Informação de Pessoal possui duas repartições:

- *Repartição de Planificação de Pessoal*
- Repartição de Estatística e arquivo de pessoal

À repartição de estatística e arquivo (REA) compete, para além doutras funções, proceder a recolha, tratamento e sistematização das informações sobre os recursos humanos existentes; desenvolver e manter actualizado um sistema de informação de pessoal que sirva de base para a tomada de decisões na área de recursos humanos do sector e fornecer dados a outras estruturas competentes.

O Departamento de Administração do Pessoal (DAP) é uma unidade estrutural que actua na área de preparação, execução e controle dos diversos actos administrativos relativos ao pessoal, na realização de concursos e respectivo provimento.

O Departamento de Administração de Pessoal compreende:

- *Repartição de Concursos e Provimento (RCP)*
- *Repartição de Previdência Social (RPS)*

A repartição de concursos e provimento (RCP) trata dos actos administrativos relativos ao pessoal no que concerne a: Ingresso no quadro do pessoal da Educação, Contratação de pessoal, Promoção, Reconversão de carreira, Progressão, Transferências, Comissão de serviço

A repartição de previdência social (RPS) prepara e controla todo o expediente relativo a reformas e falecimentos

O Departamento de Gestão e Normação (DGN) é a unidade estrutural que procede à adequação e implementação das normas gerais sobre os recursos humanos, à definição dos planos de funções, salários e quadros de pessoal e ao estudo do desenvolvimento das carreiras profissionais do sector da Educação.

O Departamento de Gestão e Normação é constituído pelas seguintes repartições:

- *Repartição de Gestão de Pessoal*
- *Repartição de Normação*

À repartição de normação (RN) compete, para além doutras funções, analisar e dar pareceres sobre o contencioso administrativo e os processos disciplinares de todos os órgãos e instituições dependents do MINED, em primeira instância

Índice

Conteúdo	página
1 INTRODUÇÃO	2
1.1 MOTIVAÇÃO	4
1.2 DESCRIÇÃO DO PROBLEMA	5
1.3 OBJECTIVOS	7
1.4 METODOLOGIA	8
2 REENGENHARIA DE SOFTWARE	9
2.1 ENGENHARIA REVERSA	11
2.2 REENGENHARIA	13
2.3 COMO REALISAR A REENGENHARIA	16
3 ORIENTAÇÃO A OBJECTOS	18
3.1 OBJECTO	18
3.2 CLASSE	18
3.3 CARACTERÍSTICAS DA ORIENTAÇÃO A OBJECTOS	19
3.4 DESENVOLVIMENTO ORIENTADO A OBJECTOS	20
4 MODELAGEM ORIENTADA A OBJECTOS USANDO A TÉCNICA OMT	21
4.1 MODELO OBJECTO	21
4.2 MODELO DINÂMICO	25
4.3 MODELO FUNCIONAL	27
4.4 RELAÇÕES ENTRE MODELOS	28
4.5 FASES DO PROCESSO DE DESENVOLVIMENTO ORIENTADO A OBJECTO	29
5 ENGENHARIA REVERSA DO SISTEMA DE INFORMAÇÃO DE PESSOAL	31
5.1 MODELO ABSTRACTO DO SISTEMA ACTUAL—SIP	31
5.2 ANÁLISE CRÍTICA DO SOFTWARE ACTUAL	35
6 MODELAGEM DO SISTEMA SIP NOVO—USANDO OMT	38
6.1 MODELO OBJECTO	38
6.2 MODELO DINÂMICO	43
6.3 MODELO FUNCIONAL	44
7 IMPACTO DA TÉCNICA OMT NA REENGENHARIA DO SIP, CONCLUSÕES E RECOMENDAÇÕES	51
7.1 O IMPACTO DO USO DE OMT NA REENGENHARIA DO SIP	51
7.2 CONCLUSÕES	52
7.3 RECOMENDAÇÕES	53
8 BILIOGRAFIA	54
ANEXO I - ORGANIGRAMA DA DRH, ABREVIATURAS, GLOSSÁRIO, NOTAÇÕES OMT	56
ANEXO II - MODELO OBJECTO DETALHADO	63
ANEXO III - MODELO DINÂMICO DETALHADO	67
ANEXO IV - MODELO FUNCIONAL DETALHADO	71
ANEXO V - TABELAS DO DEA DO SISTEMA ACTUAL	75
ANEXO VI - DICIONÁRIO DE DADOS	76

Capítulo I

1 Introdução

A variedade de problemas que envolve a manutenção de software cresce constantemente, sendo que as soluções não acompanham essa evolução. Esses problemas são resultantes de código fonte e documentação mal elaborados, além da falta de compreensão do sistema.

A partir do momento em que o sistema começa a ser usado, ele entra em um estado contínuo de mudança. Mesmo que tenha sido construído aplicando as melhores técnicas de projecto e codificação existentes, os sistemas vão-se tornando obsoletos em vista as exigências dos usuários ou novas tecnologias que são disponibilizados.

Além das correcções de erros, as mudanças mais comuns que os sistemas sofrem são as extensões em sua funcionalidade para atender as exigências dos usuários. Em geral, essas mudanças são realizadas sem que haja a preocupação com a arquitectura geral do sistema, produzindo estruturas mal projectadas, documentação desactualizada, lógica e codificação más, sendo estes, os factores que dificultam a manutenção em um sistema.

O caso de estudo neste trabalho é o Sistema de Informação de Pessoal (SIP) que a Direcção de Recursos Humanos (DRH) do Ministério da Educação (MINED) usa para fazer a gestão da informação de pessoal do sector de Educação.

O Sistema de Informação de Pessoal (SIP) é de extrema importância para a Direcção de Recursos Humanos (DRH). O sistema SIP é a base para a tomada de decisão do corpo directivo em relação ao pessoal da Educação, para tal é importante que este sistema forneça a informação completa e correcta em tempo útil.

O Sistema de Informação de pessoal (SIP) não regista toda a informação referente aos processos principais da Direcção de Recursos Humanos, havendo casos de disparidade entre a informação

existente no processo individual de pessoal (registo manual) e a existente no SIP ou, falta de informação no SIP que existe no processo individual. de pessoal.

O sistema não está documentado dificultando assim a sua compreensão e conseqüentemente também a sua manutenção.

Os problemas apresentados nos parágrafos acima são a razão da realização da reengenharia o SIP, onde se espera que este seja reescrito de forma a cobrir a maior parte dos processos principais da DRH e apresente uma documentação adequada.

Actualmente há uma preocupação muito grande em desenvolver sistemas utilizando tecnologias que se dispõem a complexidade, melhoram a manutenibilidade, permitem o reuso e reduzem o custo do ciclo de vida do software.

A orientação a objectos oferece algumas características úteis, tais como meios de abstracção bem definidos, conceitos de encapsulamento e comportamentos que efectivamente facilitam o processo de manutenção de software. Devido a estas vantagens o SIP será modelado usando OMT (*Object Modeling Technique*), uma técnica de modelação orientada a objecto.

1.1 Motivação

A existência na DRH de uma proposta de abertura de um concurso público para o desenvolvimento do sistema SIP, isto é, reconstruí-lo na forma desejada, constituiu uma grande motivação para a realização deste estudo.

Da informação colhida, através de entrevistas ao pessoal da DRH e observação do sistema, constatou-se que para além de o sistema não fornecer dados actualizados, não responder a todos os processos fundamentais da DRH, o SIP também não está documentado, dificultando assim, o seu entendimento e também a sua manutenção.

Tendo em vista os aspectos acima mencionados, é uma oportunidade realizar este estudo porque irá permitir o enriquecimento de conhecimentos adquiridos durante o curso de informática no que respeita a área de engenharia e metodologias de desenvolvimento de software especialmente as técnicas de modelação orientadas a objectos, técnicas de manutenção de software e aplicar os resultados deste estudo na resolução do problema do Sistema de Informação de Pessoal (SIP) .

A escolha do método OMT (*Object Modeling Technique*) para o desenvolvimento do novo sistema SIP deve-se ao facto de este não ter sido tratado durante o curso e haver um interesse em saber mais sobre o método, uma vez que ele cobre todo ciclo de vida do processo de desenvolvimento de software, objectivando implementações orientadas a objectos.

1.2 Descrição do problema

A Direcção de Recursos Humanos, abreviadamente designada por DRH, é o órgão do Ministério da Educação (MINED) que regulamenta, planifica, coordena e monitora a execução da política de gestão, administração e controlo do pessoal do sector da Educação.

No âmbito da informatização dos sectores do Ministério da Educação (MINED) foi desenvolvido um sistema que ajuda a DRH na gestão e utilização de recursos humanos existentes. Inicialmente o sistema funcionava com uma Base de Dado implementado em Microsoft Excel, a informação contida nesta base não era segura nem completa. Na tentativa de melhorar o sistema foi criado um sistema informático, implementado em Microsoft Access II, o sistema foi denominado *Sistema de Informação de Pessoal (SIP)*.

O sistema SIP funciona usando dois formulários principais, um para a introdução dos dados do registo de pessoal e o outro formulário serve para exibir os dados do enquadramento de pessoal e um relatório sobre o enquadramento de pessoal. O formulário para a introdução dos dados de pessoal está dividido em 4 partes: identificação de pessoal; dados do local de trabalho; dados da formação académica; dados da categoria profissional; tipo de provimento e situação em relação ao quadro. O outro formulário, que é para exibir os dados do pessoal, tem as mesmas características que o primeiro só que este, depois de receber o número do B.I. ou o nome do trabalhador, se localizado no sistema, exibe os dados do mesmo. O relatório sobre o enquadramento contém informação sobre a identificação, local de trabalho, ano de ingresso no Aparelho do Estado, categoria actual, número e data da publicação no Boletim da República e função de chefia (caso o seja).

Em casos de um trabalhador participar em um concurso para a progressão, promoção ou reconversão na carreira assim como no caso de uma transferência, a informação resultante de cada um destes processos é armazenada em arquivos existentes nos departamentos da Direcção de Recursos Humanos (DRH), estes arquivos são manuais ou bases de dados implementados em Microsoft Excel. O pessoal do Departamento de Informação de Pessoal (DIP) recolhe os dados ou processos dos arquivos para a actualização do Processo Individual (PI) e/ou do Sistema de Informação de Pessoal (SIP).

Os problemas que afligem os utilizadores do sistema SIP são:

- A disparidade entre a informação de pessoal existente nos arquivos manuais (processo individual ou registo biográfico de pessoal) e a existente no sistema SIP. Este problema deve-se ao facto de o sistema não possui rotinas nem interfaces próprios que permitem a actualização imediata do registo de pessoal no SIP, esta actualização deveria acontecer sempre que um trabalhador fosse aprovado em concursos quer para progressão, promoção ou reconversão de carreira, assim como, em casos de transferências.
- A ausência no SIP da informação sobre comissão de serviço, aposentação, falecimentos, responsabilidades disciplinares e classificação de serviço do trabalhador. Esta informação é processada manualmente e guardada em um arquivo manual que pode ser o Processo Individual e/ou Registo Biográfico ou guardada em uma base de dados implementada em Microsoft Excel.
- A ausência no SIP da informação específica sobre o trabalhador contratado como: número do contrato, data do contrato, tipo de contrato. Esta parte da informação existe só em arquivo manual.

Um outro problema é a ausência da documentação do sistema e o facto de não ter havido envolvimento maior dos utilizadores/programadores do sistema no processo de desenvolvimento do mesmo, dificultando assim o seu entendimento e conseqüentemente a sua manutenção. Não se sabe, porém, que passos terão sido tomados durante o seu desenvolvimento.

Esta situação mostra a necessidade da reengenharia do sistema SIP de modo a adaptá-lo a situação real dos processos da DRH assim como, torná-lo manutenível.

1.3 Objectivos

1.3.1 Objectivo geral

Realizar a reengenharia do Sistema de Informação de Pessoal (SIP), de forma a responder aos processos principais da DRH, nomeadamente a introdução no SIP de dados específicos de pessoal contratado; actualização imediata do registo de pessoal em caso de o trabalhador ficar aprovado em concurso de promoção, reconversão de carreira, progressão ou em caso de transferência; extensão das funcionalidades para atender processos referentes a comissão de serviço, classificação de serviço, processos disciplinares, aposentação e falecimentos que actualmente são manuais

1.3.2 Objectivos específicos

- Fazer a revisão bibliográfica a cerca do processo de manutenção de software, especificamente sobre a reengenharia e engenharia reversa de software
- Estudar e aplicar a metodologia de desenvolvimento de software orientada a objecto OMT (Object Modeling Technique) no caso de estudo
- Realizar a engenharia reversa do Sistema de Informação de Pessoal (SIP) actual
- Descrever o modelo do novo Sistema de Informação de Pessoal (SIP) usando a técnica de modelagem OMT
- Propor modelo do sistema novo à Direcção de Recursos Humano

1.4 Metodologia

1.4.1 Tipo de estudo:

Revisão dos conteúdos sobre a engenharia e reengenharia de software.

Estudo sobre metodologias orientadas a objectos.

Pesquisa na Internet e síntese dos conteúdos extraídos do material de estudo durante o curso de Informática e material disponível na biblioteca do DMI como, livros e trabalhos de licenciatura relacionados.

Realização de entrevistas ao pessoal da Direcção dos Recursos Humanos.

Observação do processo de tramitação do expediente desde a entrada, despacho e arquivo.

Consultas a documentos disponíveis (Estatutos do Funcionário do Estado, Sistema Actual das Carreiras Profissionais, Manual de procedimentos do Estatuto Geral dos Funcionários).

Análise da base de dados e dos interfaces do sistema SIP actual.

Participação em seminários e reuniões sobre a gestão de recursos humanos de modo a colher informações mais abrangentes na área de Gestão de Recursos Humanos.

Capítulo II

2 Reengenharia de Software

Para abordar adequadamente as técnicas de manutenção de software, deve se primeiramente considerar três conceitos dependentes: a existência de um processo de desenvolvimento de software, a presença de um sistema a ser analisado e a identificação de níveis de abstracção.

Qualquer que seja o processo de desenvolvimento de software espera-se que haja interacção entre seus estágios. Em um processo de desenvolvimento de software, os estágios iniciais envolvem conceitos mais gerais, independentes da implementação, enquanto que os estágios finais enfatizam os detalhes de implementação. O aumento de detalhes durante o processo de desenvolvimento conceitua os níveis de abstracção. Estágios iniciais do sistema planejam e definem requisitos de alto nível quando comparados com a própria implementação. Essa comparação é importante para deixar claro que níveis de abstracção e grau de abstracção são grandezas distintas. Enquanto que o nível de abstracção é um conceito que diferencia os estágios conceituais do projecto, o grau de abstracção é intrínseco a cada estágio. A evolução através das fases do processo de desenvolvimento de software envolve transições dos níveis mais altos de abstracção nos estágios iniciais, para níveis mais baixos nos estágios posteriores. As informações podem ser representadas em qualquer estágio de desenvolvimento, seja de forma detalhada (baixo grau de abstracção), seja de forma mais global (alto grau de abstracção).

Para que as técnicas de manutenção de software (especificamente engenharia reversa e reengenharia) sejam descritas de forma simplificada, neste estudo serão tomadas como base três fases do processo de desenvolvimento de software, com níveis de abstracção bem distintas:

Requisitos: Especificação do problema a ser resolvido, incluindo objectivos, restrições e regras de negociação;

Projecto: Especificação da solução:

Implementação: Codificação, teste e adaptação ao sistema operacional.

A técnica tradicional, que avança progressivamente pelas fases do processo de desenvolvimento de software, é denominada engenharia progressiva. A execução dessa técnica consiste em partir de projectos independentes da implementação, que possuem altos níveis de abstracção, indo em direcção a implementação física do sistema. Em outras palavras, engenharia progressiva segue a sequência de desenvolvimento estabelecida no projecto, visando a obtenção do sistema implementado.

Com excepção da engenharia progressiva, as outras transições entre as fases de desenvolvimento são tecnologias usadas na manutenção de software (CHIKOFISKY e CROSS, 1990), sendo assim definidas:

Redocumentação: como uma sub-área da engenharia reversa; é a criação ou revisão de uma representação semanticamente equivalente, dentro do mesmo nível relativo de abstracção, sendo que as formas resultantes da representação são consideradas como visões alternativas, utilizadas para uma melhor compreensão humana do sistema analisado;

Recuperação de projecto: é uma sub-área da engenharia reversa na qual o conhecimento do domínio da aplicação, informações externas e dedução são adicionadas as observações referentes ao programa, para se extraiem abstracções significativas de mais alto nível, além daquelas obtidas através da observação directa do sistema;

Restruturação: é a transformação de uma forma de representação, para outra no mesmo nível de abstracção relativo, preservando o comportamento externo do sistema (funcionalidade e semântica). Geralmente usada como uma forma de manutenção preventiva, a restruturação é aplicada em sistemas que tenham sido desenvolvidos de forma desestruturada, resultando uma representação que preserva as características do sistema, porém de forma melhor estruturada;

Engenharia reversa: é o processo de analisar um sistema com a finalidade de criar sua representação de uma forma diferente ou em um nível mais alto de abstracção que o código fonte.

Reengenharia: é a reconstrução de algo do mundo real, tendo como propósito a busca de melhorias que permitam produzir algo de qualidade melhor ou comparável ao produto inicial.

No processo de manutenção, quando se trata de reconstruir um software (ou seja, realizar sua reengenharia), é necessário, portanto, que se proceda a engenharia reversa do sistema em causa, afim de se obter os modelos de análise baseados no software existente. Esses modelos, com as devidas correcções /alterações, serão o ponto de partida para a engenharia progressiva.

2.1 Engenharia reversa

O processo inverso a engenharia progressiva, caracterizado pelas actividades retroativas do ciclo de vida, que partem de um baixo nível de abstracção para um alto nível de abstracção, é conhecido como engenharia reversa (CHIKOFFSKY e CROSS,1990).

Aplicando o conceito inicial de engenharia reversa a sistemas de software, muitas das técnicas utilizadas em hardware servem para obter uma compreensão básica do sistema e sua estrutura. Entretanto, enquanto o objectivo básico para hardware é duplicar o sistema, os objectivos mais frequentes para o software são obter uma compreensão suficiente em níveis do projecto para auxiliar a manutenção, fortalecer o crescimento do sistema, e aumentar o suporte.

2.1.1 Definições

Existem várias definições para o que se chama, engenharia reversa. Estudando a bibliografia da área, observa-se que cada autor apresenta a sua visão do que entende por esta abordagem.

A engenharia reversa de software é um processo de recuperação de projecto, consistindo em analisar um programa, na tentativa de criar uma representação do mesmo, em níveis de abstracção mais alto que o código-fonte. (Pressman 95)

Pode-se definir engenharia reversa como uma colecção de teorias, metodologias e técnicas capazes de suportar a extracção e abstracção de informações de um software existente, produzindo documentos consistentes, quer seja somente a partir do código-fonte, ou através da adição de conhecimento e experiência que não podem ser automaticamente reconstruídos a partir do código. (Chikofsky 90)

Entende-se por engenharia reversa a acção de criar um conjunto de especificações funcionais para um sistema, por alguém que não foi o projectista original, baseado na análise de um sistema existente e suas partes componentes. (Samuelson 90)

O processo de análise de um sistema para identificar os componentes de um software e seus inter-relacionamentos, e para criar uma representação do software em outra forma, provavelmente num nível mais alto de abstracção que o código-fonte, constituia a engenharia reversa. (Sommerville 95)

2.1.2 Categorias da engenharia reversa

De acordo com o nível de entendimento obtido do sistema e o propósito das informações usadas, duas categorias da engenharia reversa são definidas: a visualização de código e entendimento do programa (CHIKOFSKY e CROSS, 1990).

Visualização de código: também denominada redocumentação, a visualização de código é a criação ou revisão de representações semanticamente equivalentes num mesmo nível de abstracção. O processo de visualização de código cria as representações a partir de informações obtidas apenas da análise do código fonte, embora a apresentação dessas informações possa se diversificar. As formas das representações são consideradas visões alternativas, cujo objectivo é melhorar a compreensão do sistema global.

A forma mais simples e mais antiga de engenharia reversa é a visualização de código. A intenção é recuperar a documentação que já existiu, ou que deveria ter existido, sobre o sistema. A ênfase de facto é a criação de visões adicionais, especialmente visões gráficas, que não foram criadas durante o processo original de engenharia progressiva.

Recuperações mais ambiciosas tais como a função, os propósitos ou a essência do sistema exigem um nível de entendimento maior e são definidas como entendimento do programa.

Entendimento de programa: nesta categoria de engenharia reversa, também denominada recuperação de projecto, o conhecimento do domínio das informações externas e as deduções são adicionadas as

observações feitas sobre o sistema através do exame do mesmo, de modo a obter informações com nível mais alto de abstração.

O entendimento de programa recria abstrações do projecto a partir de uma combinação de código, documentação existente do projecto (se disponível), experiências pessoais e conhecimentos gerais sobre o problema e o domínio de aplicação. Sintetizando, deve produzir todas as informações necessárias para se entender completamente o que, como, e porque o sistema faz.

Um completo entendimento de programa busca reconstruir não somente a função do sistema, mas também o processo pelo qual o sistema foi desenvolvido. É de grande importância a recuperação de decisões de projecto tomadas durante o desenvolvimento original para uma completa estrutura de entendimento.

2.2 Reengenharia

O termo reengenharia está relacionado com a reconstrução de algo do mundo real, e independentemente da sua aplicação, o seu principal propósito é a busca por melhorias que permitam produzir algo de qualidade melhor ou, pelo menos, de qualidade comparável ao nível inicial.

2.2.1 Definições

A reengenharia, conhecida também como renovação ou reconstrução, é o exame e alteração de um sistema de um software, para reconstruí-lo em uma nova forma, e a subsequente implementação dessa nova forma. Um processo de reengenharia geralmente inclui alguma forma de engenharia reversa, seguida por uma forma de engenharia progressiva ou reestruturação. (Chikofsky e Cross 90)

A reengenharia de software é descrita como a reorganização e modificação de sistemas de software existentes, parcial ou totalmente, para torná-los mais manteníveis. (Sommerville 95)

A reengenharia tem como principal objectivo melhorar um sistema de alguma maneira, através de alterações significantes que proporcionem melhoria, porém, sem alterar suas funções.

A reengenharia pode ser dividida em duas fases principais: a Engenharia Reversa e a Engenharia Progressiva; e cada uma destas fases pode ser dividida em uma série de actividades.

O objectivo da engenharia reversa é derivar o projecto ou especificação de um sistema, partindo-se do seu código fonte. O objectivo da reengenharia é produzir um sistema novo com maior facilidade de manutenção e a engenharia reversa é usada como parte do processo de reengenharia, pois fornece o entendimento do sistema a ser reconstruído.

2.2.2 Categorias da reengenharia

Existem diversas categorias relacionadas a reengenharia, entre elas podemos destacar as seguintes:

Reengenharia de processos administrativos: que é direccionada para alterações potenciais em todos os negócios ou processos organizacionais;

Reengenharia de processos produtivos: consiste em modificar qualquer ciclo de processo padrão, que esteja em uso em uma dada organização, afim de melhor acomodar as tecnologias novas e emergentes, bem como os requisitos dos clientes para um produto ou sistema;

Reengenharia de sistemas de software ou produto: é o exame, estudo, captura e modificação de mecanismos internos ou funcionalidades de um sistema existente ou produto, visando reconstruí-lo em uma nova forma e com novas características, normalmente para tomar vantagem das novas e emergentes tecnologias, mas sem grandes alterações na funcionalidade e propósito inerentes ao sistema.

Quando se efectua a reengenharia de processos administrativos, provavelmente será necessário efectuar reengenharia do software, visto que existe uma dependência implícita entre os processos administrativos e o software (SOMMERVILLE, 1995).

2.2.3 Reengenharia do software legado

Todos os sistemas têm um tempo de vida limitado, sendo que cada alteração efectuada pode degenerar a sua estrutura, fazendo com que as manutenções se tornem cada vez mais difíceis e dispendiosas. Isso ocorre principalmente em software legado.

Um software legado pode ser informalmente definido como aquele que executa tarefas úteis para a organização e que foi desenvolvido utilizando-se técnicas actualmente considerados obsoletos. A migração e/ou alteração desse tipo de software gera desafios técnicos (por exemplo; estimar o custo de alteração) a todos os envolvidos com a manutenção de software.

Muitos programas legados são críticos para os negócios das organizações que os possuem. Eles têm embutidas informações dos negócios e procedimentos, que podem não estar documentados. O risco de remover e reescrever tais programas é grande, pois muita informação teria que ser redescoberta por tentativa e erro. Consequentemente, as organizações, de um modo geral, não “aposentam” seus sistemas legados, preferindo mantê-los em operação, com adaptações as novas necessidades. As dificuldades devem ser enfrentadas, pois qualquer software que não evoluir continuamente (i.e. sofrer manutenções) tornar-se á menos útil no mundo real.

Os maiores problemas em manter software legado são:

- Desestruturação e dificuldade de entendimento do código, muitas vezes porque o software foi desenvolvido antes da introdução dos métodos de programação estruturada;
- Programadores que não participam do desenvolvimento de um produto de software sentem dificuldades em entender e organizar a funcionalidade do código fonte;
- Documentação desactualizada, não auxiliando em nada a equipe de manutenção;
- Dificuldade de predizer as consequências de efeitos colaterais;
- Dificuldade de administrar múltiplas alterações concorrentes.

Aplicando-se a reengenharia de software o sistema pode ser redocumentado ou reestruturado, os programas podem ser traduzidos para uma linguagem de programação mais moderna e implementados em uma plataforma distribuída, bem como seus dados podem ser migrados para uma

base de dados diferente. A reengenharia de software tem como objectivo fazer sistemas flexíveis, fáceis de modificar perante as constantes mudanças das necessidades dos usuários.

Existe uma firme e crescente demanda para migrar programas legados de mainframes processados, para estações multiprocessadas, distribuídas e ligadas em rede, visando acompanhar os avanços das técnicas de programação tais como: interfaces gráficas para o usuário, comunicação interprogramas, desenvolvimento orientado a objectos, reusabilidade, etc...

2.3 Como realizar a reengenharia

O processo de reengenharia de software é constituído por duas fases distintas. Na primeira, o software objecto de reconstrução é decomposto em partes, de modo a ser entendido. Na segunda, o software é reconstruído, na forma desejada, a partir do produto da primeira fase, sendo incluídos os ajustes que forem necessários.

O processo de reengenharia pode ser traduzido como:

Reengenharia = Engenharia Reversa + Engenharia Progressiva

onde pode ser de dois tipos:

- Alterações parciais de funcionalidade (alteram parcialmente o objectivo principal do sistema): ocorrem devido a mudança nos negócios, ou necessidade do usuário;
- Alterações de implementação (alteram a forma de implementação do sistema): ocorrem devido a alterações no ambiente de operação do software e/ou linguagem de implementação (protocolos, sistema operativo, portabilidade, linguagens, etc.)

2.3.1 O processo de reengenharia

Jacobson e Lindström (1991) declaram que, para executar um processo de reengenharia de um sistema, é necessário:

Realizar a engenharia reversa: identificar como os componentes do sistema se relacionam entre eles e então criar uma descrição mais abstracta do sistema. Com este passo, obtém-se um modelo

abstracto, que mostra a funcionalidade do sistema (propósito para o qual o sistema foi construído) e um número de mapeamentos entre os diferentes níveis de abstracção. Os mapeamentos compreendem as decisões de projecto que ocorrem quando se transforma uma representação abstracta em uma representação concreta;

Decidir sobre as alterações na funcionalidade: as alterações de funcionalidade são alterações nos requisitos que o usuário determina que sejam implementados no sistema. Esse passo é executado utilizando-se as abstracções de mais alto nível, obtidas no passo anterior.

Reprojectar o sistema: parte-se das abstracções de alto nível, obtidas nos passos anteriores, para uma representação mais concreta, ou seja, executa-se a engenharia progressiva reimplementando o sistema. Neste processo, deve-se levar em consideração as alterações de técnicas de implementação. Se apenas parte do sistema for alterado, devem-se considerar questões sobre a integração/comunicação entre as partes velhas e novas do sistema.

Capítulo III

3 Orientação a Objectos

A orientação a objectos é bem mais do que um paradigma de programação, aplica-se a várias actividades relacionadas com tecnologias de informação: análise de requisitos, processos, desenho de software, construção e testes de sistemas de software. A ideia central consiste na construção de modelos de sistemas em torno de entidades que unificam dados e procedimentos: *objectos*

Podemos definir a orientação a objectos como uma forma de organizar o software como uma colecção de objectos discretos que incorporam estrutura de dados e comportamento.

As secções que seguem apresentam definições relacionadas à tecnologia de objectos que são de extrema importância à compreensão das actividades e metodologias baseadas em objectos.

3.1 Objecto

Um objecto é algo distinguível que contém *atributos* (ou propriedades) e possui um *comportamento*. Cada objecto tem uma identidade e é distinguível de outro mesmo que seus atributos sejam idênticos. O conjunto de valores associados às propriedades do objecto define o estado deste; o comportamento descreve as mudanças do estado do objecto interagindo com o seu mundo externo, através das *operações* realizadas pelo objecto.

3.2 Classe

Uma classe é o agrupamento de objectos com a mesma estrutura de dados (definida pelos *atributos* ou *propriedades*) e comportamento (*operações*). Uma classe é uma abstracção que descreve as propriedades importantes para uma aplicação e não leva em conta as outras. Cada classe descreve um conjunto possivelmente infinito de objectos individuais. Cada objecto é uma *instância* de classe. Cada instância de classe tem seu próprio valor para cada um dos atributos da classe mas compartilha os nomes e as operações com as outras instâncias de classe.

3.3 Características da orientação a objectos

A orientação a objectos caracteriza-se principalmente pela abstracção, encapsulamento, herança e polimorfismo.

3.3.1 Abstracção

A abstracção consiste em focar os aspectos mais importantes de um objecto (visão externa; o que é e o que ele faz), ignorando suas características internas (visão interna; como ele deve ser implementado).

3.3.2 Encapsulamento

O encapsulamento é o empacotamento de dados (atributos) e de operações sobre estes (métodos). No caso da orientação a objectos, os dados não podem ser acessados diretamente mas através de mensagens enviadas para as operações. A implementação de um objecto pode ser mudada sem modificar a forma de acessa-lo.

3.3.3 Herança

A herança consiste no compartilhamento de atributos e operações entre as classes numa relação hierárquica. Este mecanismo permite a uma classe ser gerada a partir de classes já existentes. Uma classe pode ser definida de forma abrangente e posteriormente refinada em termos de sub-classes e assim sucessivamente. Cada subclasse *herda* todas as propriedades e operações da sua superclasse, (que não precisam ser repetidas) adicionando apenas as suas propriedades específicas.

A relação entre classes é uma das grandes vantagens de sistemas orientados a objectos por causa da redução de trabalho resultante durante o projecto e a programação.

3.3.4 Polimorfismo

O polimorfismo significa que uma mesma operação pode se comportar de forma diferente em classes diferentes. Uma operação que tem mais de um método que a implementa é dita *polimórfica*.

Nos sistemas orientados a objectos, o suporte seleciona automaticamente o método correcto que implementa a operação, a partir do nome da operação e da classe do objecto no qual esta se operando.

3.4 Desenvolvimento orientado a objectos

O desenvolvimento orientado a objectos diz respeito aos procedimentos de concepção de sistemas a partir dos conceitos básicos anteriores. Ele corresponde às principais fases do ciclo de vida de software: análise, projecto e implementação.

Sabe-se que por vezes a programação orientada a objectos (OOP) não é uma tecnologia suficiente para o desenvolvimento das aplicações. De modo a obter os benefícios reais da tecnologia de objecto, a análise orientada a objectos (OOA) e o desenho orientado a objectos (OOD) devem vir antes da programação orientada a objectos. Ao longo dos anos muitos métodos foram descritos para a análise e desenho de sistemas orientados a objectos (OOS). A abordagem mais significativa da metodologia vem de Booch, Rumbaugh (OMT—Object Modeling Technique), OOSE Jacobson (Use Case Approach/Objectory), Schlaer-Mellor, e Wirfs-Brock (Responsibility Driven Design). Recentemente, Booch, Rumbaugh, e Jacobson uniram as suas metodologias individuais em metodologia unificada usando a linguagem de modelação unificada (UML), que consiste em um conjunto comum de modelos e notações para a análise e desenho orientados a objectos.

Capítulo IV

4 Modelagem orientada a objectos usando a Técnica OMT

Uma das mais populares técnicas de desenvolvimento orientado a objectos é OMT desenvolvido por Rumbaugh. Por causa da sua fácil notação, OMT tem se mostrado fácil entender, desenhar e usar. Este continua a ser muito sucedido em muitos domínios de aplicações: telecomunicação, transporte, compiladores, etc. (Schaufelberger 1998)

OMT define graficamente um sistema através de três tipos de modelos:

- um **modelo objecto** para representar os aspectos estáticos, estruturais, de “dados” de um sistema;
- um **modelo dinâmico** para representar os aspectos temporal, comportamental e de “controle” de um sistema;
- um **modelo funcional** para representar os aspectos transformacionais e de “função” de um sistema.

4.1 Modelo objecto

O modelo objecto descreve a estrutura de objectos no sistema: sua identidade, suas relações com os outros objectos, seus atributos e suas operações. Este modelo tenta capturar a estrutura estática dos componentes do mundo real que se pretende representar. O modelo objecto tem uma representação gráfica na forma de diagramas, contendo as classes de objetos com seus atributos e operações, e organizados segundo hierarquias e associações com os diagramas de outras classes.

4.1.1 Objetos e classes

Todos os objectos têm uma identidade e são distinguíveis. Eles são instâncias de classes de objectos que descrevem grupos de objectos com similaridade nas propriedades (atributos), comportamento (operações), relações com os outros objectos e semântica. A notação gráfica para representar objectos, classes e suas relações é composta de dois tipos de diagramas, mostrados na figura 4.1-1:

- um **diagrama de classe**, que representa classes de objectos e tem a função de ser um esquema, um modelo de dados;
- um **diagrama de instância**, utilizado para representar instâncias de classes e tem o objectivo de descrever como um conjunto particular de objectos se relaciona com outro.

O atributo é colocado na segunda parte da caixa (ver figura 4.1-2). Cada nome de atributo pode ser seguido por detalhes opcionais tais como tipo (precedido por ":") e valor default (precedido de "="). Identificadores (explícitos) de objecto não são necessários no modelo objecto pois cada objecto já tem a sua propria identidade (a partir de seus valores).

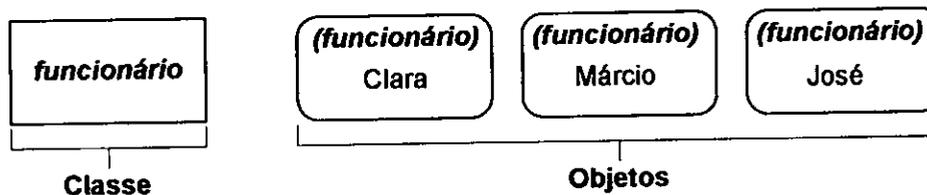


figura 4.1-1—Ilustração de Classes e Objectos.

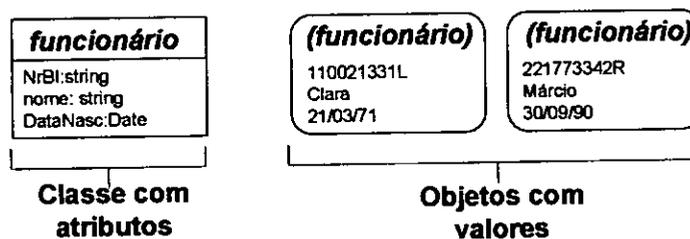


figura 4.1-2—Ilustração de Atributos e Valores.

Uma operação pode ser aplicada a ou por objectos numa classe. A mesma operação pode se aplicar a várias classes diferentes (polimorfismo). Um método é a implementação de uma operação para uma classe. Quando uma operação tem métodos em várias classes, eles devem ter a mesma assinatura (em número e tipos de argumentos). As operações se encontram na terceira parte da caixa, como ilustra a figura 4.1-3. Cada operação pode ser seguida de detalhes opcionais tais como lista de argumentos (colocada entre parênteses após o nome, separados por ",") e tipo de resultado (precedido por ":"). A notação generalizada para classes de objetos se encontra na figura 4.1-4.

Funcionário	LocalTrabalho	Contrato
NrBI Nome DataDeNasc	Identificador DescrLocal TipoDeLocal	NrContrato DataContrato DuraçãoContr
Criar Promover Exonerar	AdicionarLoc ActualizarLoc EliminarLoc	RenovarContrato RescindirContrato

figura 4.1-3—Operações com Objectos.

Nome da Classe
nome_atributo-1 : tipo_dado-1 = valor default-1 nome_atributo-2 : tipo_dado-2 = valor default-2 • • •
nome_operação-1:(lista_argumentos-1) = tipo_result-1 nome_operação-2:(lista_argumentos-2) = tipo_result-2 • • •

figura 4.1-4—Generalização da notação de modelagem de objectos.

4.1.2 Ligações e associações

Uma ligação é uma t-upla matemática correspondente a uma lista ordenada de instâncias de objectos.

Uma ligação é uma instância de uma associação.

Uma *associação* ou *association* descreve um grupo de ligações com estrutura e semântica comuns. Todas as ligações de uma associação conectam objectos das mesmas classes. O conceito de associação é similar aquele usado na área de base de dados (ver anexo I).

Associações são muitas vezes implementadas em linguagens de programação como apontadores (atributo de um objecto que contém referencia explícita a um outro) de um objecto para outro.

Multiplicidade de Associações :

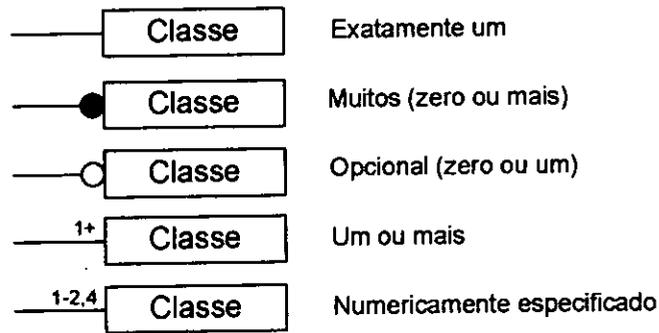


figura 4.1-5—notação para associações múltiplas

4.1.3 Agregação

A agregação é uma relação “parte-todo” ou “uma-parte-de”. A agregação é definida como uma relação entre uma classe junção e uma classe componente. A agregação é então uma forma especial de associação. A simbologia da agregação é similar a da associação (linha), exceto pelo fato de um pequeno losango indicar o lado da junção da relação, conforme a figura.

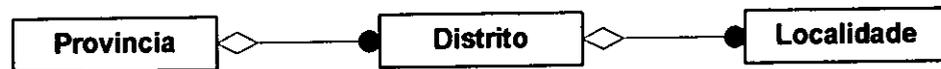


figura 4.1-6—agregação

4.1.4 Generalização e herança

Generalizações e herança são abstrações poderosas que permitem compartilhar similaridades entre as classes, apesar de preservar suas diferenças.

Generalização é uma relação do tipo “é um” entre uma classe e uma ou mais versões refinadas. A classe que está sendo refinada é uma superclasse e a classe refinada é uma subclasse. Atributos e operações comuns a um grupo de subclasses são reagrupados na superclasse e compartilhado por cada subclasse. Diz-se que cada subclasse herda as características de sua superclasse.

Utiliza-se a generalização para se referenciar à relação entre classes, enquanto a herança diz respeito ao mecanismo de compartilhar atributos e operações usando a relação de generalização.

Generalização e especialização são dois pontos de vista da mesma relação, no primeiro caso vista a partir da superclasse e no segundo da subclasse: a superclasse generaliza a subclasse e a subclasse refina ou especializa a superclasse.

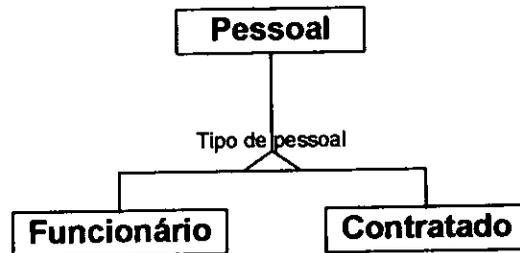


figura 4.1-7—generalização e herança

4.1.5 Módulo

Um módulo é uma construção lógica para reagrupar classes, associações e generalizações. Um módulo captura uma perspectiva ou uma vista de uma situação. Um modelo objecto consiste de um ou vários módulos. A noção de módulo força a particionar um modelo objecto em peças gerenciáveis. Uma mesma classe pode ser referenciada em módulos diferentes. De facto, referenciando a mesma classe em múltiplos módulos é o mecanismo para interligar módulos entre si.

4.2 Modelo dinâmico

O modelo dinâmico descreve os aspectos do sistema que dizem respeito ao tempo e à sequência de eventos (operações). Este modelo tenta capturar o controle, aspecto de um sistema que descreve as sequências de operações que ocorrem em resposta a estímulos externos, sem levar em conta o que as operações fazem, quem as ativa e como são implementadas. Os conceitos utilizados nesta modelagem dinâmica são os de eventos que representam os estímulos externos e de estados que representam os valores de objectos.

A representação gráfica é um diagrama de estados que representa os estados e a sequência de eventos permitidos num sistema para uma classe de objectos. Os estados e eventos podem ainda serem organizados de forma hierárquica e representados num diagrama de estados estruturado. (ver anexos I/III)

4.2.1 Eventos e estados

Cada *evento* é uma ocorrência única; entretanto é possível reagrupá-los em classes de eventos e dar a cada uma delas um nome que indica uma estrutura e um comportamento comuns. Alguns eventos são simples sinais mas muito outros tem atributos indicando a informação que eles transportam. O tempo no qual o evento ocorre é um atributo implícito de todos os eventos. Um evento transporta a informação de um objeto a outro; os valores de dados transportados por um evento são seus atributos. Os eventos incluem as condições de erro e as ocorrências normais.

Um *cenário* é uma sequência de eventos que ocorre durante uma execução particular de um sistema. Ele pode incluir todos os eventos do sistema ou apenas eventos gerados por certos objectos no sistema. A sequência de eventos e os objectos que trocam eventos podem ser mostrados juntos num diagrama de *rastro de eventos*.

Um *estado* especifica a resposta do objecto à eventos de entrada. A resposta de um objecto à um evento pode incluir uma acção ou uma mudança de estado pelo objecto.

Um estado tem uma duração; ele ocupa um intervalo de tempo entre dois eventos. Na definição de estados, pode se ignorar atributos que não afetam o comportamento do objecto. O estado é caracterizado por um nome e uma descrição contendo a sequência de eventos que leva ao estado, a condição que o caracteriza e os eventos aceites neste estado com a ação que ocorre e o estado futuro. O estado pode incluir os valores de suas ligações.

O *diagrama de estados* relaciona estados e eventos. A mudança de estado causada por um evento é chamada de *transição*. Os diagramas de estado podem representar ciclos de vida uma-vez (com um estado inicial e um estado final) que representam objectos com vida finita ou malhas contínuas. Um modelo dinâmico é uma coleção de diagramas de estado que interagem entre si através de eventos compartilhados.

Condições podem ser usadas como guardas nas transições, sendo que uma transição guardada dispara quando o evento ocorre e que a condição de guarda é verdadeira.

4.2.2 Operações

Uma descrição do comportamento de um objecto deve especificar o que o objecto faz em resposta a eventos. Operações associadas à estados ou transições são realizadas em resposta aos estados correspondentes ou a eventos.

Uma *atividade* é uma operação que leva tempo para se completar. Ela é associada a um estado. A notação “do: A” dentro de um caixa de estado indica que a atividade A inicia na entrada no estado e para na saída.

Uma acção representa uma operação cuja duração é pequena comparada com a resolução do diagrama de estados. Acções podem também representar operações de controle interno. A notação para uma acção numa transição é um “/” seguido do nome da acção, após o evento que a causa. (ver anexos I... para detalhes)

4.3 Modelo funcional

O modelo funcional descreve os aspectos do sistema que dizem respeito as transformações de valores: funções, mapeamentos, restrições e dependências funcionais. Este modelo captura *o que* o sistema faz sem levar em conta *o como* e *o quando* ele faz.

O modelo funcional é representado por vários diagramas de fluxo de dados (DFDs) que mostram as dependências entre valores e o cálculo de valores de saída a partir de valores de entrada e de funções. O modelo funcional inclui também as restrições entre valores no modelo objecto (ver anexo I).

Um DFD contém *processos* que transformam dados, *fluxos de dados* que movimentam dados, *objectos actores* que produzem e consomem dados, *objetos armazenamento de dados* que estocam os dados.

Um *processo* transforma valores de dados. Um processo é implementado como um método de uma operação de uma classe de objectos. O objecto alvo é usualmente um dos fluxos de entrada,

especialmente se a mesma classe de objecto é também um fluxo de saída. Em alguns casos, o objecto alvo é implícito.

Um *fluxo de dados* conecta a saída de um objecto ou de um processo a entrada de um outro objecto ou processo. Ele representa um valor de dados intermediário num cálculo. O valor permanece sem mudança no fluxo de dados.

Actores são ligados as entradas e saídas de um diagrama de fluxo de dados. Eles podem ser vistos como fontes e receptores de dados.

Um *arquivo de dados* é um objecto passivo do diagrama de fluxo de dados que armazena dados para um acesso futuro. Como no caso de um actor, um arquivo não gera operações sobre ele mesmo mas simplesmente responde a pedidos para armazenar e acessar dados. O acesso pode ser feito em ordem diferente do armazenamento. Destaca-se que actores e arquivos de dados são objetos que se diferenciam pelo seu comportamento e uso; actores são implementados como dispositivos externos e arquivos como armazenadores.

Um *diagrama de fluxo de dados* (DFD) é particularmente útil para mostrar a funcionalidade de alto-nível de um sistema e sua quebra em unidades funcionais menores. Um processo pode ser expandido num outro DFD no qual as entradas e saídas dos processos o são também no novo diagrama. Eventualmente, o aninhamento de diagramas termina com funções simples que devem ser especificadas como operações.

4.4 Relações entre modelos

Cada modelo descreve um aspecto do sistema mas contém referências aos outros modelos. O modelo objecto descreve a estrutura de dados sobre a qual os modelos dinâmico e funcional operam. As operações no modelo objecto correspondem aos eventos no modelo dinâmico e as funções no modelo funcional.

O modelo dinâmico descreve a estrutura de controle dos objectos. As acções no diagrama de estados correspondem as funções no diagrama funcional. Os eventos no diagrama de estados se tornam as operações no modelo objecto.

O modelo funcional descreve as funções invocadas pelas operações no modelo objecto e acções no modelo dinâmico. As funções operam sobre os valores de dados especificados pelo modelo objecto. O modelo funcional mostra ainda as restrições sobre os valores objecto

4.5 Fases do processo de desenvolvimento orientado a objecto

A figura 4.5-1 mostra as fases do processo de desenvolvimento orientado a objecto usando OMT

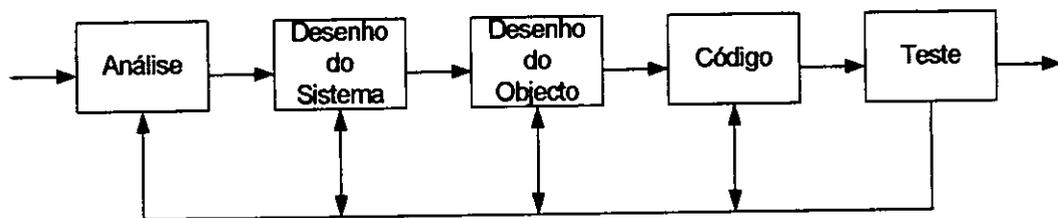


figura 4.5-1—processo OMT

Análise—entender e modelar a aplicação e o seu domínio

Desenho do sistema—definir a arquitectura geral do sistema em termos de subsistemas, tarefas concorrentes e arquivo de dados

Desenho do objecto—redefinir, depois optimizar o modelo de análise, a partir de aplicações para conceitos de computador

Codificação—implementar as classes de objectos através de uma determinada linguagem de programação

Teste—em qualquer estágio do sistema constam classes de objectos testados e não testados. A testagem é baseada em senários desenvolvidos como parte do processo do modelo dinâmico

Cada fase do processo transforma alguns inputs em outputs, partindo do alto nível de abstracção para o nível mais detalhado de abstracção que representa por fim a solução do problema (figura 4.5-2)

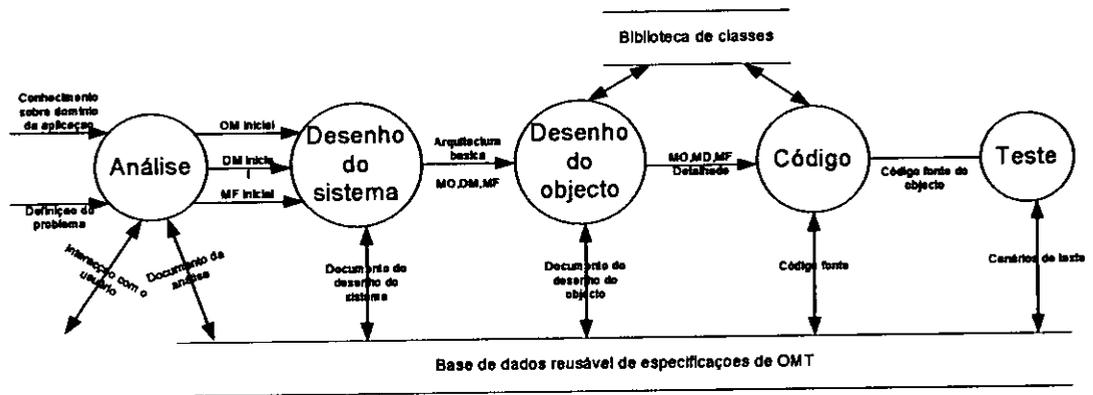


figura 4.5-2—inputs e outputs do processo de desenvolvimento orientado a objecto usando OMT

Capítulo V

5 Engenharia reversa do Sistema de Informação de Pessoal

O processo inverso a engenharia progressiva, caracterizado pelas actividades retroativas do ciclo de vida, que partem de um baixo nível de abstracção para um alto nível de abstracção, é conhecido como engenharia reversa (CHIKOFFSKY e CROSS, 1990).

5.1 Modelo abstracto do sistema actual—SIP

A informação que constitui o Registo de pessoal da Educação é produzida principalmente no Departamento de Administração de Pessoal (DAP) e através da Repartição de Administração Interna (RAI) a informação chega ao Departamento de Informação de Pessoal (DIP) onde o registo do funcionário é criado, alterado, extinto ou fazer a consulta sobre o enquadramento do funcionário.

Do estudo feito, foram identificados processos que tratam da informação sobre o pessoal da Educação que em seguida serão descritos.

A repartição de concursos e provimento (RCP) trata dos seguintes processos:

- Ingresso no quadro do pessoal da Educação
- Contratação de pessoal
- Promoção
- Reconversão de carreira
- Progressão
- Transferências
- Comissão de serviço
- Classificação de serviço

A repartição de previdência social (RPS) trata dos seguintes processos:

- Aposentação
- Subsídio por morte

A repartição de normação (RN) trata do seguinte processo:

- Responsabilidades disciplinares

A repartição de estatística e arquivo (REA) trata dos seguintes processos:

- Criar registo de pessoal
- Actualizar registo de pessoal
- Extinguir registo de pessoal
- Consultar registo de pessoal

5.1.1 Identificação de Sub-rotinas no sistema actual

Os processos “introduzir registo de pessoal” e “ver enquadramento” no sistema actual serve-se das rotinas que em seguida são listadas pelos respectivos nomes:

1. *IntPessoal()*: Sub-rotina que permite a introdução dos dados do registo do funcionário
2. *VldIntPessoal()*: Sub-rotina que permite a validação dos dados do registo introduzido
3. *GravaIntPessoal()*: Sub-rotina que permite a gravação dos dados do registo do funcionário
4. *ProcuraResgs("BI")*: Sub-rotina que localiza o registo do funcionário através do número do BI
5. *MostraInf()*: Sub-rotina que permite exibir no formulário “frmIntPessoal” a informação sobre o enquadramento do funcionário

6. *OpenReport(rptVerEnquadramento)*: Um comando que permite o carregamento do relatório para ser imprimido

7. *BtnImprimir()*: Sub-rotina que permite a impressão do relatório

O modelo abstracto do sistema actual será constituído por diagramas de fluxo de dados (DFDs) e diagrama de entidade e associação (DEA) que é, deste modo, a visualização do código do sistema actual (alto nível de abstracção). O modelo descreve o software que é, neste estudo, o objecto de reengenharia.

5.1.2 Diagrama de Fluxo de dados do processo "contratação de pessoal"

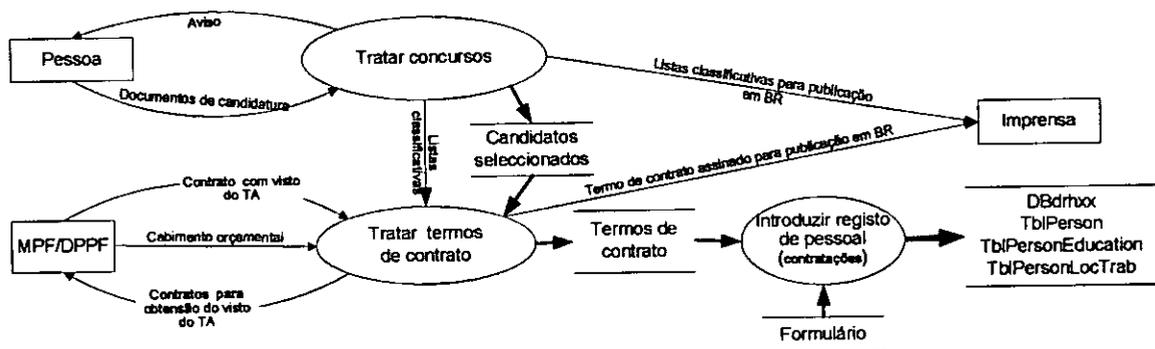


figura 5.1-1—DFD do processo de ingresso no aparelho de estado por contrato

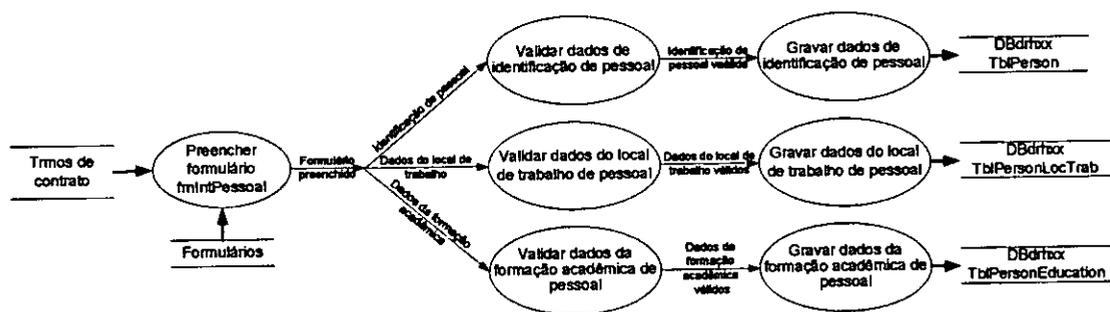


figura 5.1.1-2—DFD detalhado do processo "introduzir registo de pessoal"

5.1.3 Diagrama de Fluxo de Dados do processo "Ingresso no aparelho de estado"

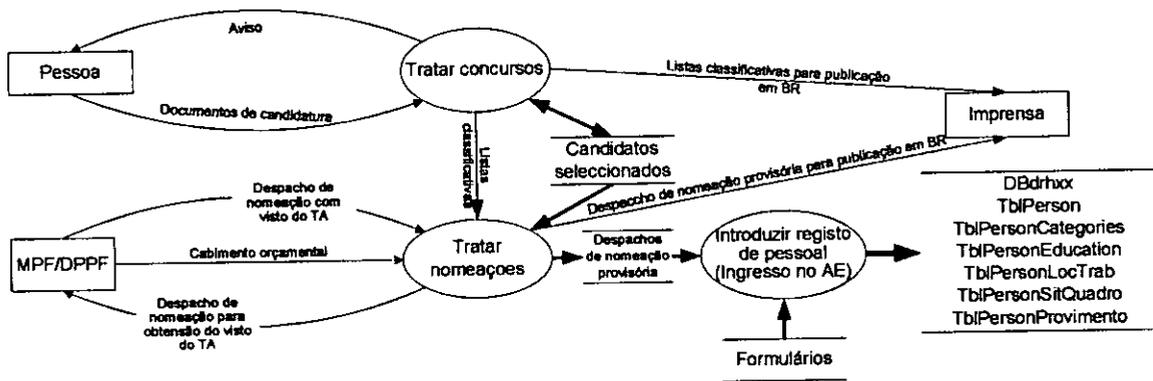


figura 5.1-2—DFD do processo de ingresso no AE

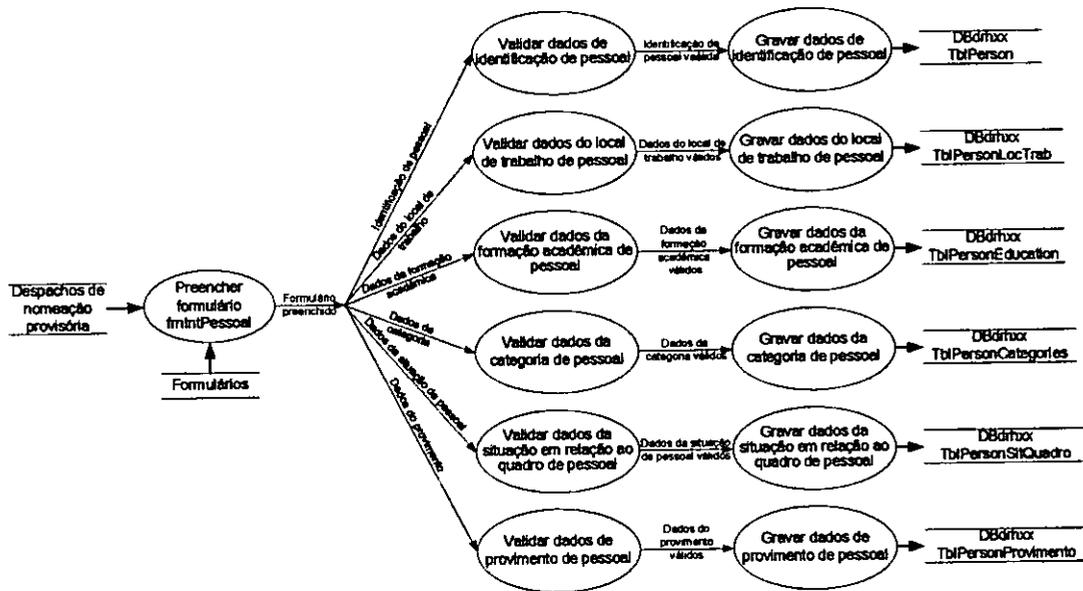


figura 5.1-3—DFD detalhado do processo "Introduzir registo de pessoal"

5.1.4 Diagrama de Fluxo de Dados do processo "Consultar enquadramento)

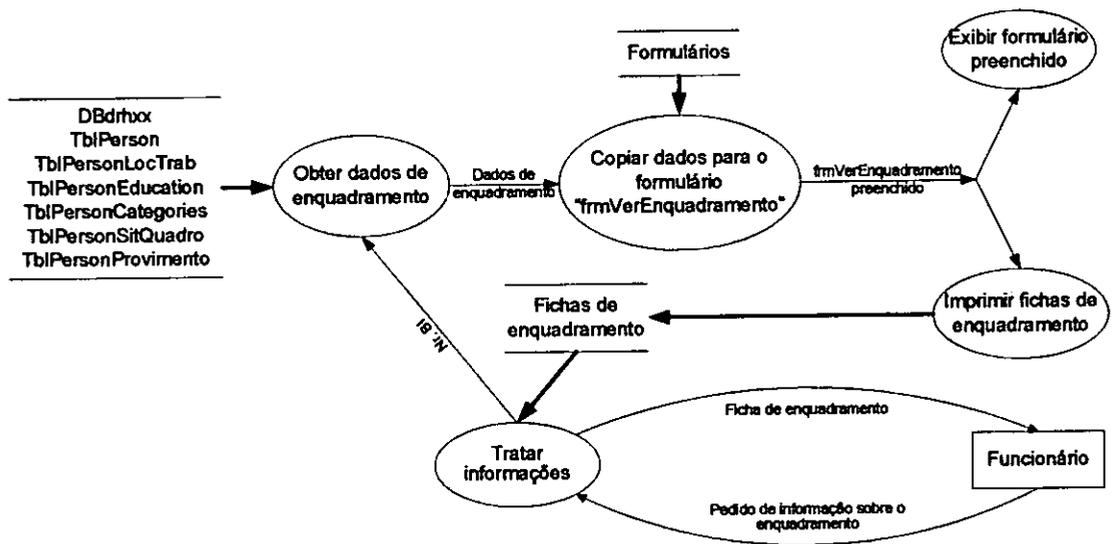


figura 5.1-4—DFD do processo "Consultar Enquadramento de pessoal" no sistema

5.1.5 Modelo entidade e associação

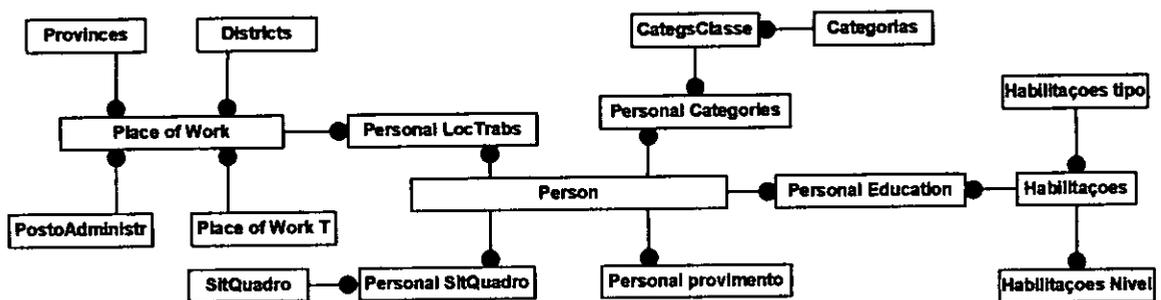


figura 5.1-5—diagrama de entidade e associação "DBdrhxx" do sistema actual

(O detalhe das tabelas do sistema actual pode ser visto no anexo V)

5.2 Análise crítica do software actual

1. O primeiro problema do Sistema de Informação de Pessoal (SIP) actual é não estar documentado, dificultando assim, a sua compreensão e consequentemente a sua manutenção

2. Não há clareza sobre o método que foi usado no desenvolvimento do SIP
3. O DAP faz o registo de informação de pessoal em bases de dados implementadas em Microsoft Excel ou em arquivos manuais, o DIP faz a recolha destas informações e as regista no Sistema de Informação de Pessoal implementado em Microsoft Access II. Estas duas bases têm estruturas diferentes, obrigando assim a duplicação da informação de pessoal, ou inexistência no Sistema de Informação de Pessoal de parte da informação contida nos arquivos do DAP
4. Não há uma clara distinção no tratamento da informação de pessoal no sistema, os dados do funcionário e os do contratado são introduzidos usando o mesmo formulário, este formulário não tem campos para a introdução dos dados específicos do contratado tais como: número do contrato, data do contrato, salário do contratado e data do visto do tribunal administrativo. Esta informação é omitida no sistema.
5. Em caso de progressão, promoção, reconverção de carreira ou transferência, o sistema não tem rotinas nem interfaces que permitem a actualização imediata do registo de pessoal, estas informações ficam guardadas em arquivos manuais ou bases em Excel usadas no DAP, esta situação faz com que haja disparidade na informação no SIP e a existente nos arquivos do DAP.
6. O sistema actual não possui processos que tratam de questões relacionadas com a participação dos funcionários em concursos, comissões de serviço, processo disciplinar, informação de serviço aposentação e falecimentos. Estas são informações contidas em arquivos manuais ou bases em Microsoft Excel existentes nas diferentes repartições da DRH. Salientar que estas informações são importantes e fazem parte do processo individual de pessoal

Estes são uma parte dos problemas identificados na análise do sistema actual, mas que justificam uma reengenharia de software

O próximo capítulo será dedicado a modelagem do novo sistema SIP, será usada a técnica de modelagem orientada a objecto (OMT) para a modelação do sistema SIP na nova forma. O novo modelo do SIP irá incluir as alterações de modo a solucionar os problemas acima mencionados.

Capítulo VI

6 Modelagem do Sistema SIP novo—usando OMT

A reengenharia, conhecida também como renovação ou reconstrução, é o exame e alteração de um sistema de um software, para reconstruí-lo em uma nova forma, e a subsequente implementação dessa nova forma. Um processo de reengenharia geralmente inclui alguma forma de engenharia reversa, seguida por uma forma de engenharia progressiva ou reestruturação.

OMT graficamente define um sistema através de três tipos de modelos: modelo objecto, modelo dinâmico e modelo funcional. Estes modelos são tratados em separado nas secções que se seguem.

6.1 Modelo objecto

O modelo objecto é o modelo mais importante. Este identifica as classes de objectos no sistema e seu relacionamento, assim como seus atributos e operações (métodos).

As secções que se seguem são dedicadas à descrição da estrutura de objecto no SIP: sua identidade, suas relações com outros objectos, seus atributos e suas operações. O modelo objecto detalhado pode ser visto no **anexo II**.

6.1.1 Cadastro de pessoal

No sistema são considerados dois tipos de pessoal da Educação; o funcionário, cuja relação com o aparelho de estado (MINED) é através do despacho de nomeação (ou provimento) e o contratado cuja relação com o aparelho de estado (MINED) é através de contrato. O contratado pode concorrer para ser funcionário do quadro se satisfazer as condições exigidas para tal.

Na modelagem, o pessoal pode ser representado como uma estrutura hierárquica de classes onde “pessoal” é a super-classe das sub-classes “funcionário” e “Contratado”, como mostra a figura 6.1-1

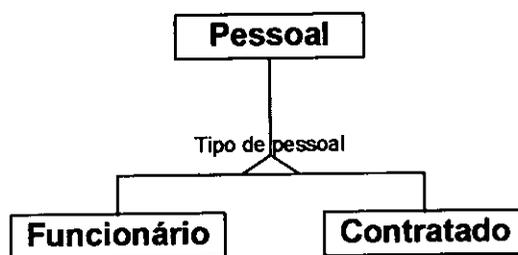


figura 6.1-1—cadastro de pessoal

6.1.2 Enquadramento do funcionário na carreira

Um funcionário da Educação pode ser enquadrado, em um determinado momento, numa carreira profissional de acordo com o nível acadêmico e o lugar a ocupar. Numa carreira podem estar enquadrados vários funcionários. A modelagem desta situação pode ser representada como mostra a figura 6.1-2

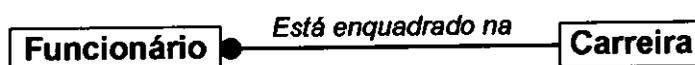


figura 6.1-2—enquadramento do funcionário na carreira profissional

6.1.3 Participação de pessoal em concursos

O pessoal da Educação pode, satisfeitas as condições exigidas, concorrer para ser promovido na carreira, mudar de carreira (se for do quadro) ou entrar para o quadro do pessoal (se for eventual).

Um trabalhador da Educação pode participar em um concurso, de cada vez, em um concurso podem participar vários trabalhadores da Educação. A associação “participa em” pode ser modelada como classe “Participação em concurso”, como mostra a figura 6.1-3



figura 6.1-3—participação de pessoal em concursos

6.1.4 Funcionário em comissão de serviço

Um funcionário pode ser designado a exercer uma função de direcção, em um determinado momento. Uma função de direcção pode ser exercida por vários funcionários da Educação. Esta associação pode ser modelada como mostra a figura 6.1-4

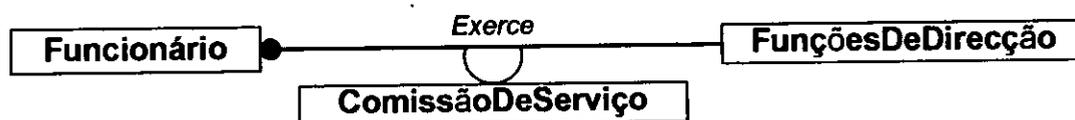


figura 6.1-4—funcionário em exercício de funções de direcção

6.1.5 Habilitações académicas do funcionário

Um funcionário da educação pode ter uma formação académica ou mais, a formação pode ser geral, profissional educacional ou profissional não educacional. Uma formação, podem ter vários trabalhadores da Educação. Esta situação pode ser modelada como mostra a figura 6.1-5

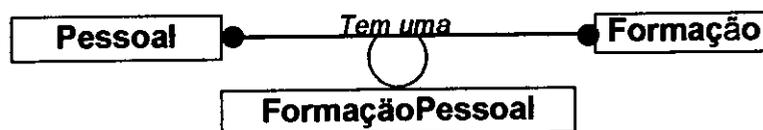


figura 6.1-5—habilitações de pessoal

6.1.6 Afectação do funcionário a um local de trabalho

A classe “transferências” irá permitir o registo da informação sobre transferências, no sistema actual, tal informação só existe em arquivo manual

Entende-se por colocação a afectação de pessoal da Educação na prestação de serviço num local determinado.

Entende-se por transferência a afectação de pessoal a tarefas em local diferente daquele em que se encontra a prestar serviço.

Todo pessoal da educação no activo está afecto em algum local de trabalho. Dependendo das necessidades de serviço ou a pedido do funcionário, este pode ser transferido.

Em um local de trabalho, podem estar afectos muitos trabalhadores.

Províncias agregam Distritos, Distritos agregam localidades (Posto administrativo).

Na modelagem a afectação de pessoal a locais de trabalho pode ser representada por uma estrutura hierarquica de classes em que “Afectação” é super-classe e “Transferência” e “Colocação” são sub-classes. O modelo pode ter a representação apresentada na figura 6.1-6

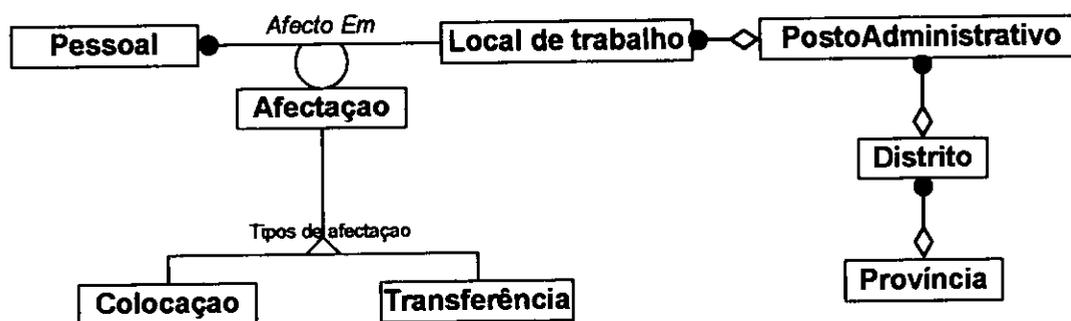


figura 6.1-6—afectação de pessoal em um local de trabalho

6.1.7 Processo disciplinar do funcionário

“Processo disciplinar” é uma classe que irá permitir o registo de informação sobre responsabilidades disciplinares de pessoal. No sistema actual esta informação é guardada em um arquivo manual

Um trabalhador da Educação pode cumprir uma pena disciplinar, de cada vez. Uma pena pode ser aplicada a vários trabalhadores. Esta situação pode ser modelada como mostra a figura 6.1-7

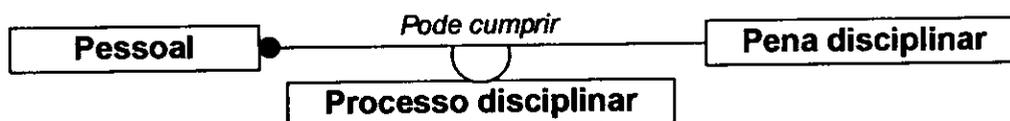


figura 6.1-7—responsabilidades disciplinares de pessoal

6.1.8 Situação do funcionário em relação ao quadro

O funcionário da Educação em relação ao quadro a que pretence, pode encontrar-se nas seguintes situações: Actividade no quadro, Actividade fora do quadro, Inactividade no quadro, Supranumerário ou Aposentado

Um funcionário pode estar, de cada vez, em uma situação. Em uma situação podem estar vários funcionários. A situação em relação ao quadro pode ser modelada como mostra a figura 6.1-8



figura 6.1-8—situação do funcionário em relação ao quadro

6.1.9 Classificação de serviço

A classe “avaliação de serviço” irá permitir o registo da classificação de serviço de pessoal. No sistema actual, a classificação de serviço é armazenado em um arquivo manual.

A classificação de serviço visa a avaliação permanente do mérito do funcionário no exercício das funções que lhe estão atribuídas,

Em relação a cada um dos indicadores constantes da folha de classificação é registado um dos seguintes valores de ponderação: Muito bom, Bom, Regular e Mau. Um trabalhador da Educação pode, durante a sua vida na Educação, ter de cada vez, uma determinada classificação de serviço. Uma classificação de serviço pode ser para vários funcionários. A associação pode ser modelada como classe “Avaliação de serviço”. A figura 6.1-9 mostra a modelagem desta situação.



figura 6.1-9—classificação de serviço

6.1.10 Aposentação

“Aposentação” é uma classe que vai permitir o registo de informações sobre a aposentação de funcionários, no sistema actual, este registo é feito em um arquivo manual. A associação é de um para um, pois as características da aposentação variam para cada funcionário. Esta situação pode ser modelada como mostra a figura 6.1-10.



figura 6.1-10—aposentação do funcionário

6.1.11 Falecimento

“Falecimento” é uma classe que irá permitir o registo de informação sobre o falecimento de funcionários, no sistema actual o registo destas informações é feito em um arquivo manual. A associação é de um para um, como mostra a figura 6.1-11

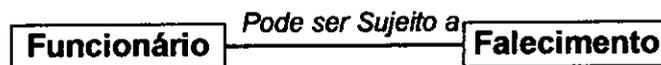


figura 6.1-11—falecimento de um funcionário

6.2 Modelo dinâmico

O modelo dinâmico indica a dinâmica de objectos e suas mudanças de estado.

Em seguida são apresentados dois diagramas de estado que correspondem aos processos de ingresso no aparelho de estado e contratação de pessoal.

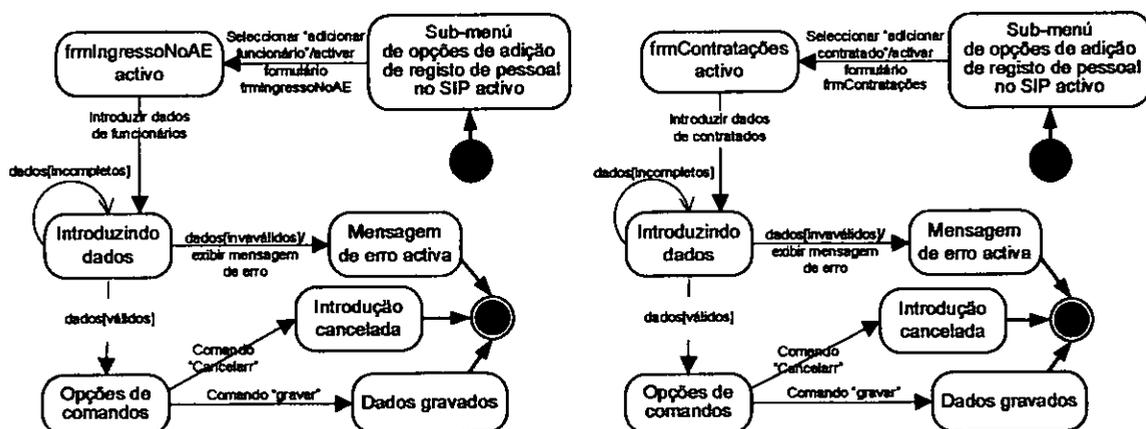


figura 6.2-1—modelo dinâmico

(Os diagramas de estado de outros processos podem ser vistos no **anexo III**)

6.3 Modelo Funcional

O sistema de gestão de informação de pessoal encontra-se dividido em três processos principais

1. Adicionar o registo de pessoal
2. Actualizar registo de pessoal
3. Consultar registo de pessoal

6.3.1 Adicionar registo de pessoal

A adição do registo de pessoal depende da execução de dois processos, que em seguida serão descritas pelos respectivos diagramas de fluxo de dados.

6.3.1.1 Ingresso no aparelho de Estado

O diagrama de fluxo de dados do processo de ingresso no aparelho de estado é apresentado na figura 6.3-1. (O DFD detalhado do processo “Adicionar registo de funcionário” pode ser visto no anexo IV)

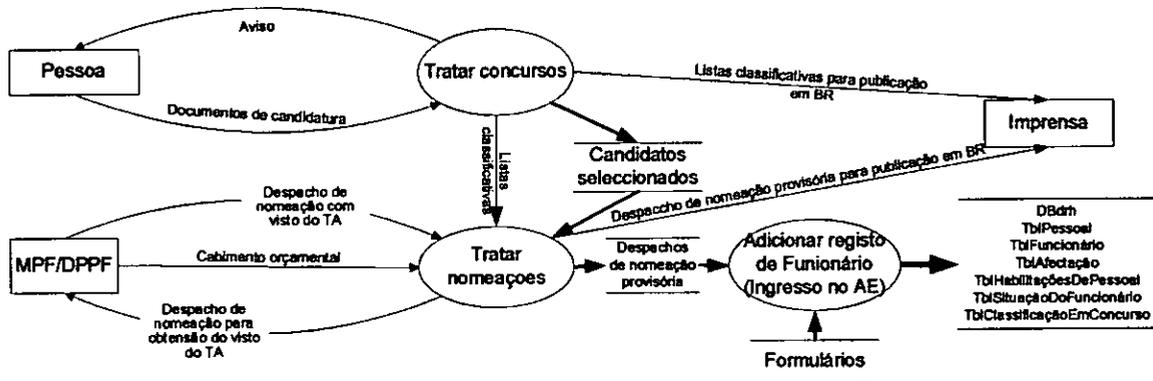


figura 6.3-1— DFD do processo de ingresso no aparelho de estado

6.3.1.2 Contratação de pessoal

O diagrama de fluxo de dados do processo de contratação de pessoal é apresentado na figura 6.3-2. (O DFD detalhado do processo “Adicionar registo de contratado” pode ser visto no anexo IV)

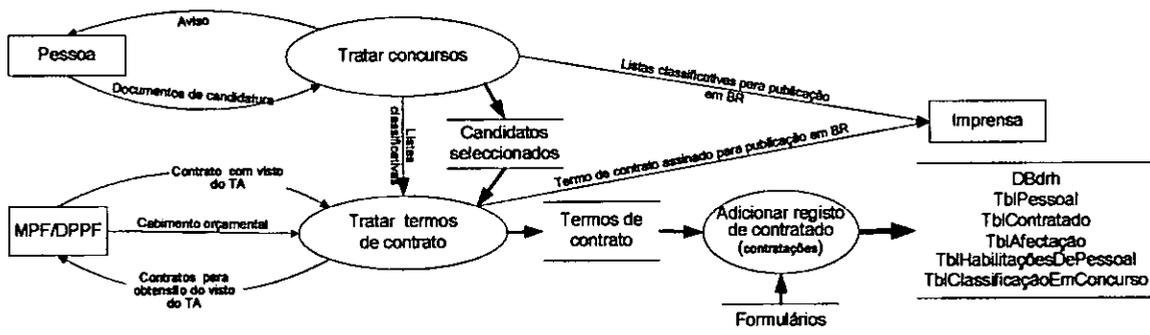


figura 6.3-2— DFD do processo de ingresso na Educação por contrato

6.3.2 Actualizar registo de pessoal

A actualização do registo de pessoal depende da execução dos processos que em seguida serão descritas através dos respectivos diagramas de fluxo de dados

6.3.2.1 Progressão do funcionário na carreira

O diagrama de fluxo de dados do processo de progressão na carreira é apresentado na figura 6.3-3. (O DFD detalhado do processo “Actualizar registo de Funcionário (progressão)” pode ser visto no anexo IV)

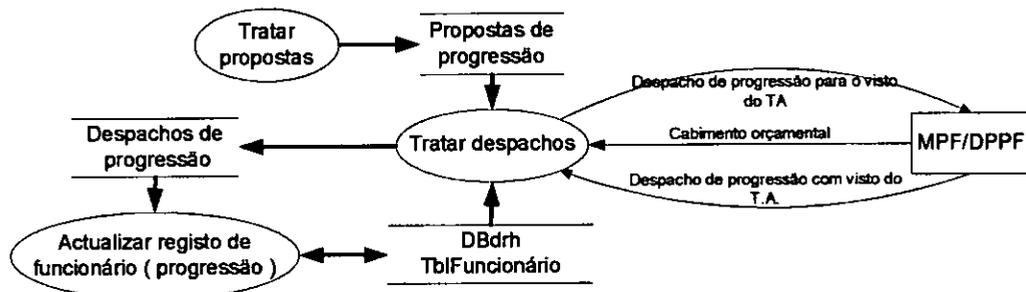


figura 6.3-3— DFD do processo de progressão na carreira

6.3.2.2 Promoção do funcionário na carreira

O diagrama de fluxo de dados do processo de promoção na carreira é apresentado na figura 6.3-4. (O DFD detalhado do processo “Actualizar registo de funcionário (promoção)” pode ser visto no anexo IV)

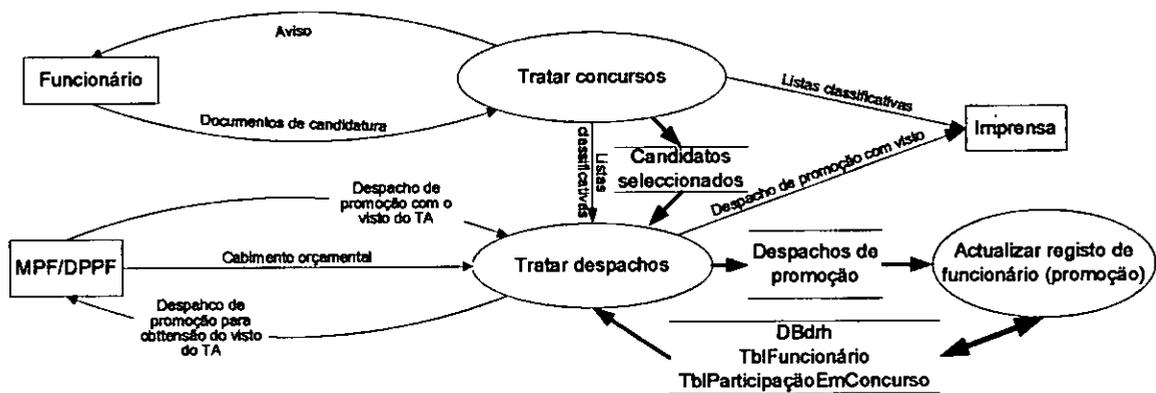


figura 6.3-4— DFD do processo de promoção na carreira

6.3.2.3 Reconversão de carreira

O diagrama de fluxo de dados do processo de reconversão de carreira é apresentado na figura 6.3-5. (O DFD detalhado do processo “Actualizar registo de funcionário (reconversão de carreira)” pode ser visto no anexo IV)

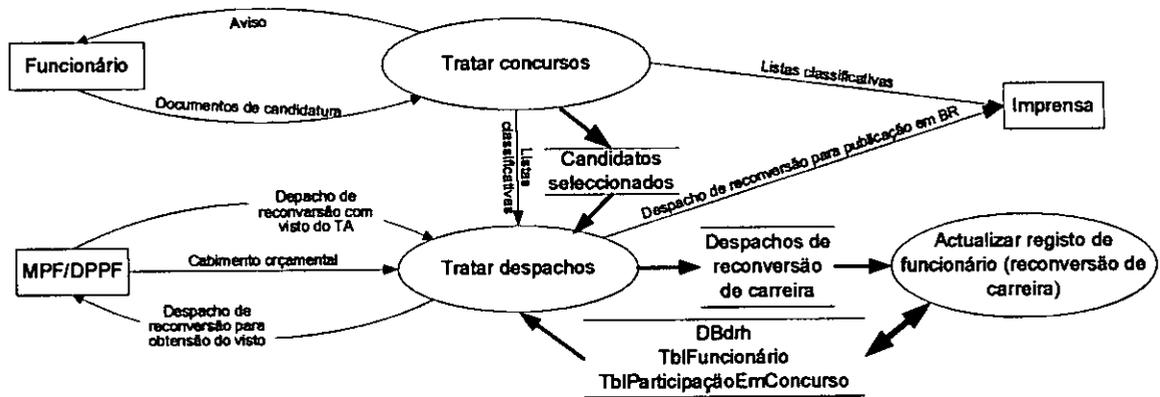


figura 6.3-5— DFD do processo de reconversão de carreira

6.3.2.4 Transferências

O diagrama de fluxo de dados do processo de transferência na carreira é apresentado na figura 6.3-6. (O DFD detalhado do processo “Actualizar registo de pessoal (Transferência)” pode ser visto no anexo IV)

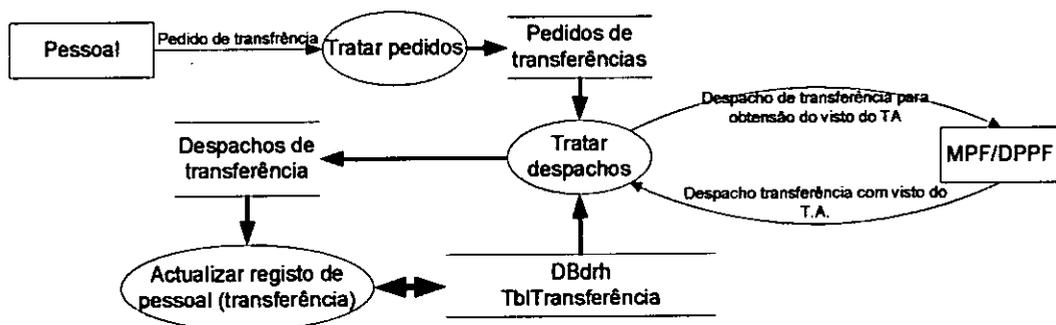


figura 6.3-6— DFD do processo de transferência de pessoal

6.3.2.5 Comissão de serviço

O diagrama de fluxo de dados do processo de comissão de serviço é apresentado na figura 6.3-7. (O DFD detalhado do processo “Actualizar registo de funcionário (Transferência)” pode ser visto no anexo IV)

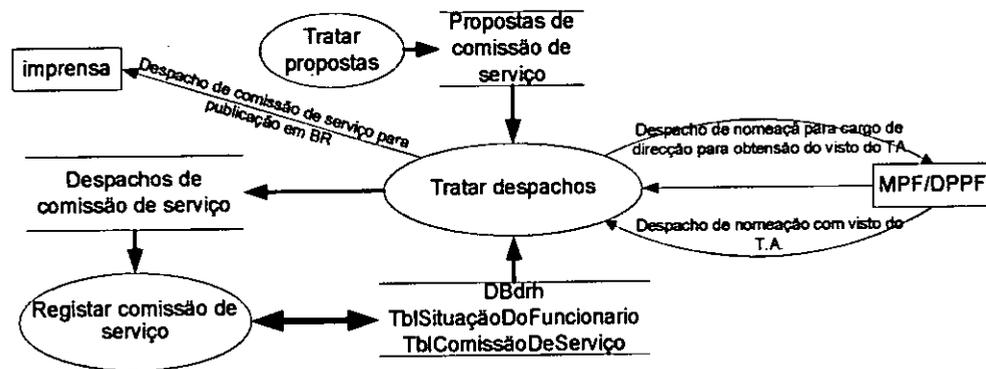


figura 6.3-7— DFD do processo de designação do funcionário para funções de chefia

6.3.2.6 Aposentação

O diagrama de fluxo de dados do processo de aposentação é apresentado na figura 6.3-8. (O DFD detalhado do processo “Registrar reformas” pode ser visto no anexo IV)



figura 6.3-8— DFD do processo de aposentação

6.3.2.7 Falecimentos

O diagrama de fluxo de dados do processo de tratar subsídio por morte é apresentado na figura 6.3-9. (O DFD detalhado do processo “Registrar falecimentos” pode ser visto no anexo IV)

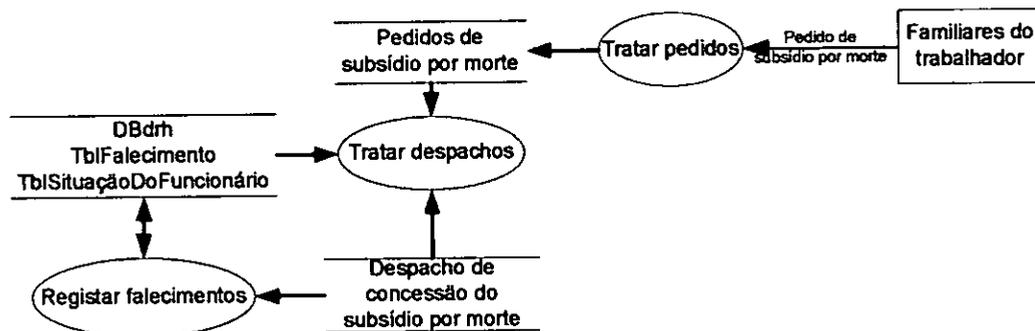


figura 6.3-9— DFD do processo de falecimentos

6.3.2.8 Responsabilidades disciplinares

O diagrama de fluxo de dados do processo que trata de responsabilidades disciplinares é apresentado na figura 6.3-10. (O DFD detalhado do processo “Registrar processo disciplinar” pode ser visto no anexo IV)

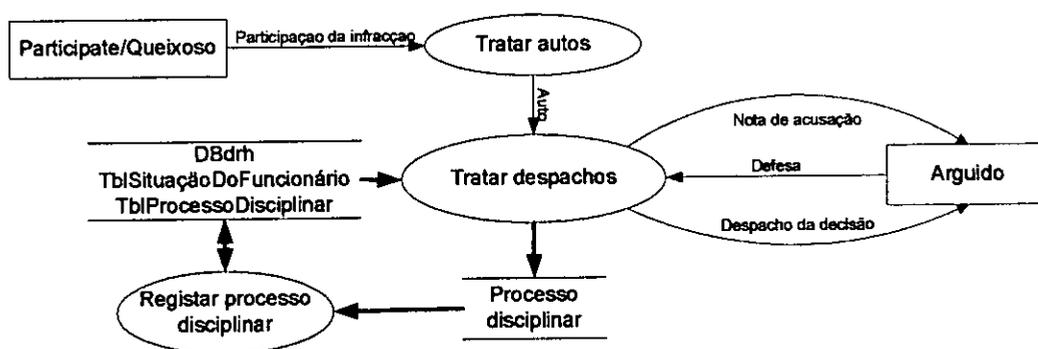


figura 6.3-10— DFD do processo disciplinar

6.3.2.9 Avaliação de serviço

O diagrama de fluxo de dados do processo que trata de Avaliação de serviço é apresentado na figura 6.3-11. (O DFD detalhado do processo “Registrar classificação de serviço” pode ser visto no anexo IV)

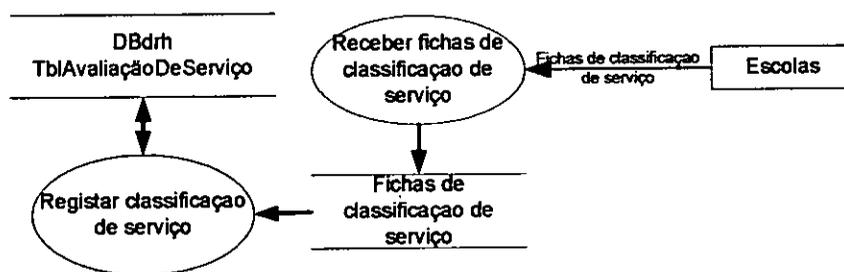


figura 6.13-11—DFD do processo de avaliação de serviço de pessoal

6.3.3 Consultar registo de pessoal

A consulta mais importante até agora é, ver o enquadramento de pessoal, pois com a participação de pessoal em concursos a situação de enquadramento muda. No futuro o sistema poderá produzir relatórios de classificação em concursos, comissão de serviço, transferências, classificação de serviço, aposentação e falecimentos.

O processo “consultar enquadramento de pessoal” é descrito na figura 6.3-12 através do diagrama de fluxo de dados

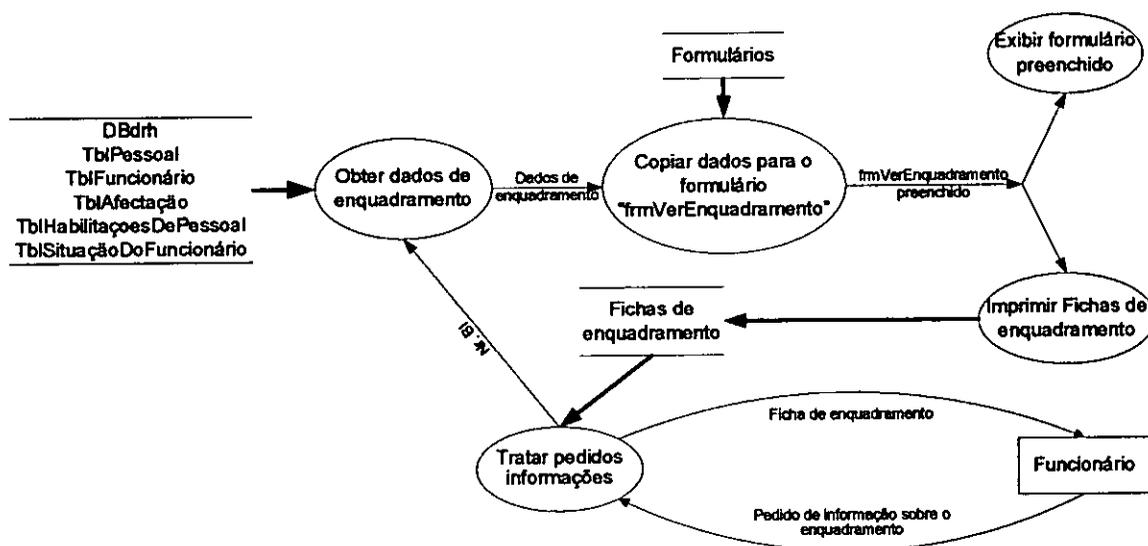


figura 6.3-12— DFD do processo de consulta de enquadramento

Capítulo VII

7 Impacto da técnica OMT na reengenharia do SIP, Conclusões e Recomendações

7.1 O impacto do uso de OMT na reengenharia do SIP

A escolha e uso da estratégia de desenvolvimento de software OMT, durante a reengenharia do SIP mostrou-se vantajosa nos seguintes aspectos:

- OMT é fácil de ser entendido e desenhar
- OMT permitiu a construção de modelos do sistema SIP em torno de entidades que unificam dados e procedimentos: **objectos**
- O uso do modelo objecto no SIP incentiva a reutilização do software, porque os objectos são componentes reusáveis devido a sua independência
- OMT cobre todo o processo de desenvolvimento de software, tal que não há necessidade de mudar de um modelo para outro

A desvantagem no uso de OMT para desenvolver o SIP relaciona-se com a dificuldade na identificação dos objectos reais do sistema.

7.2 Conclusões

Foi definido inicialmente como objectivo principal a reengenharia do Sistema de Informação de Pessoal (SIP). Para a concretização deste objectivo foi necessário realizar actividades específicas tais como: revisão bibliográfica sobre a reengenharia e engenharia reversa de software; estudo da metodologia de desenvolvimento de software OMT (Object Modeling Technique); realização da engenharia reversa do sistema SIP e; desenvolvimento do modelo novo do sistema SIP usando OMT.

A revisão bibliográfica permitiu abordar neste relatório conteúdos essenciais sobre o processo de manutenção de software, especificamente sobre a reengenharia e engenharia reversa de software. A descrição do relatório clarificou os conceitos “o que é a reengenharia”, “porque realizar a reengenharia” e “como realizar a reengenharia” de software.

Foram também tratados neste relatório conteúdos relacionados com a técnica de modelagem de software—Object Modeling Technique (OMT). O estudo e uso desta técnica no desenvolvimento do modelo do SIP novo mostrou o seu valor. OMT respeita os princípios fundamentais da orientação a objectos (abstracção, encapsulamento, herança e polimorfismo). O desenvolvimento orientado a objectos usando OMT cobre todo o processo de desenvolvimento de software, este modelo é simples de usar e fácil de desenhar e entender. Através desta técnica foi possível produzir modelos de análise do SIP que servem a engenharia progressiva.

A recuperação da documentação que deveria ter existido sobre o sistema, especialmente visões gráficas que não foram criadas durante o processo original da engenharia progressiva, foi possível através da engenharia reversa. Com essas visões foi possível compreender o software e o sistema em que o SIP está inserido, possibilitando a produção de modelos de análise que servem a reengenharia.

A reengenharia do software do SIP resultou em uma nova versão com documentação actualizada e consistente e que agregou toda a informação do software legado, mais as novas funcionalidades, satisfazendo assim as necessidades dos utilizadores (ou novas necessidades do processo em que o sistema está inserido).

A disparidade da informação existente nos arquivos da DRH e a existente no sistema SIP será reduzida, pois, o sistema novo tem processos que tratam da actualização do registo do trabalhador em casos de promoção, progressão, reconversão de carreira ou transferência e tratam também de registo de informações que antes era feito só em arquivos manuais (Processo Individual e/ou Registo Biográfico) nomeadamente a comissão de serviço, classificação de serviço, processo disciplinar, aposentação, falecimentos e a informação específica dos trabalhadores contratados. A informação da decisão de despachos de expediente será facilmente introduzida no sistema, reduzindo assim o tempo entre os despachos e a actualização dos dados no sistema, pois este sistema será único e estará disponível em cada repartição. Salienta-se que as máquinas do MINED estão em rede.

De uma maneira geral pode se concluir que com a aplicação da reengenharia, o sistema SIP foi redocumentado (ou reestruturado). As actividades da reengenharia permitiram elevar a capacidade de entendimento do software existente e o melhoramento do mesmo. Porém, neste estudo a técnica OMT não foi toda explorada por causa do limite do tempo no desenvolvimento do sistema SIP foram só consideradas as fases de análise. As outras fases ficam para o estudo posterior a este.

7.3 Recomendações:

1. Por haver ainda muita informação sobre o sistema por descobrir, é recomendável aos utilizadores/programadores do sistema, uma análise comparativa dos dois modelos de sistemas, o sistema SIP novo e o SIP actual antes da implementação.
2. Para melhor tirar vantagens da metodologia OMT usada no desenvolvimento do novo modelo de sistema, é fundamental que a equipe encarregue pela implementação do sistema em causa tenha conhecimentos sólidos sobre linguagens de programação orientadas a objectos

Capítulo VIII

8 Bibliografia

Chikofsky, E.J.; Cross, J. H. Reverse Engineering and Design Recovery: A Taxonomy. IEEE Software (1990)

Pressman, R. S. Engenharia de Software, São Paulo: Markron Books, (1995)

Sommerville, I. Software Engineering. 5ª Edição, (1995)

Uwadia, C. O. Seminar on Software re-engineering, Department of Computer Sciences, Universidade de Lagos, (2002)

Yourdon, E. Análise Estruturada Moderna, Rio de Janeiro: Editora Campus, (1990)

Rumbaugh, J.; Blaha, M.; Premerlani, W. Modelagem e Projectos Baseados em Objectos. Campus, Rio de Janeiro, RJ. (1991)

Schaufelberger, W.; Killer, D.; Kraus, F. Object-Oriented Software Development Using Object Modeling Technique (OMT). Xiaobing Qiu, Institut für Automatik, ETH Zurich (1998)

Jacobson, I.; Lindstrom, F. Re-engineering of Old Systems to an Object-Oriented architecture. SIGPLAN (1991)

Piekarski, A. E. T.; Quináia, M. A. Reengenharia de Software. Departamento de Informática, UNICENTRO 85010-990, PR (2000)

Reis, A.; Matos, A.; Costa, M. S. Estatuto Geral dos Funcionários do Estado, 2ª Edição actualizada. NBD/Paarl Print, Drukery Street, Cape Town, South Africa, nº 01961/ RLINLD (2001)

MAE, Manual de Procedimentos do Estatuto Geral dos Funcionários de Estado. Imprensa Nacional de Moçambique, Maputo (2002)

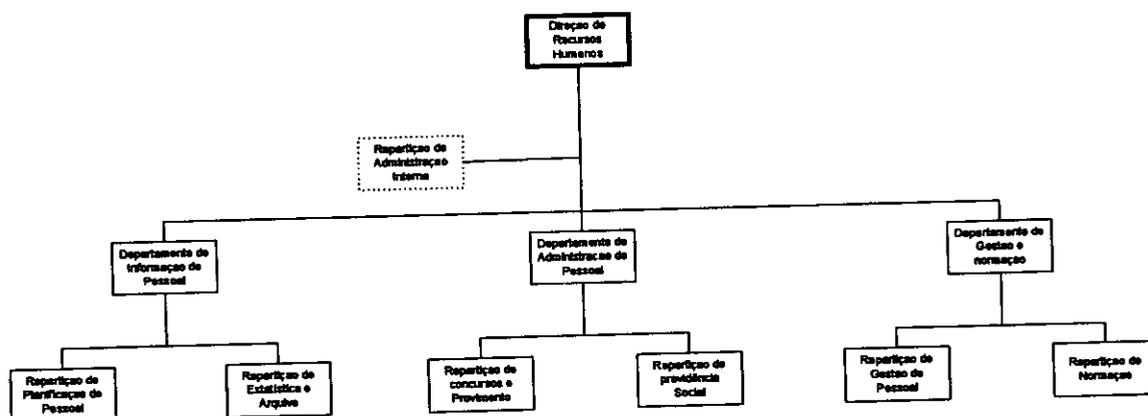
Introdução a Metodologia de Investigação, Universidade Eduardo Mondlane, Faculdade de Ciências

URL:<http://www.sei.cmu.edu/~reengineering/pubs/lsysree> (1995).

<http://www.rational.com/uml>

Anexo I - Organigrama da DRH, Abreviaturas, Glossário, Notações OMT

ORGANIGRAMA ESTRUTURAL DA DIREÇÃO DE RECURSOS HUMANOS



Abreviaturas:

- DAP—Departamento de Administração de Pessoal
- DD—Dicionário de Dado
- DEA—Diagrama de Entidade Associação
- DE—Diagrama de Estrutura
- DFD—Diagrama de Fluxo de Dados
- DGN—Departamento de Gestão e Normaço
- DIP—Departamento de Informação de Pessoal
- DM—Dynamic Model
- DNCP—Direcção Nacional de Contabilidade Púlica
- DPE—Direcção Provincial de Educação
- DPPF—Direcção Provincial do Plano e Finanças
- DRH—Direcção de Recursos Humanos
- EGFE—Estatuto Geral dos Funcionários do Estado
- FM—Functional Model
- MPF—Ministério do Plano e Finanças
- OM—Object Model
- OMT—Object Modeling Technique

OO—Object Oriented

RAI—Repartição de Administração Interna

RCP—Repartição de Concursos e Provisamento

REA—Repartição de Estatística e Arquivo

RGP—Repartição de Gestão de Pessoal

RN—Repartição de Normação

RPP—Repartição de Planificação de Pessoal

RPS—Repartição de Previdência Social

TA—Tribunal Administrativo

GLOSSÁRIO

Acção: operação instantânea associada a um evento.

Actor: objeto activo que conduz o diagrama de fluxo de dados produzindo ou consumindo valores.

Admissão: conjunto de actos através dos quais um indivíduo vai ocupar um lugar

Arquivo de dados: objecto passivo do diagrama de fluxo de dados que armazena dados para um acesso futuro.

Atividade: operação que leva tempo para se completar.

Carreira: é um conjunto hierarquizado de classes ou categorias de idêntico nível de conhecimentos e complexidade a que os funcionários têm acesso de acordo com o tempo de serviço e o mérito profissional.

Colocação: afectação de um funcionário do aparelho de Estado na prestação de serviço num local determinado.

Comissão de Serviço: é um regime especial de actividades destinado ao exercício de funções classificados como de “Direcção e Chefia”, inspecção ou lugares de confiança.

Concurso: é o conjunto de actos ou operações que se destinam a recrutar, a seleccionar e a classificar ou a graduar segundo critérios rigorosos e objectivos, as pessoas e os funcionários que se candidatam a lugares de ingresso ou acesso no aparelho de Estado e que preencham os requisitos previamente estabelecidos para certa categoria ou função.

Escalão: é cada uma das posições remuneratórias criadas para as carreiras horizontais ou da faixa salarial de cada classe ou categoria das carreiras mistas.

Estado: abstracção dos valores dos atributos e das ligações de um objecto.

Estrutura salarial: contém os índices salariais organizados em faixas salariais correspondentes as classes do grupo salarial em que cada carreira se integra.

Evento: algo que ocorre num instante de tempo e que não tem duração

Exoneração: termo da relação de trabalho com o Estado. Desocupação de um lugar no aparelho de Estado, pode ser a pedido do funcionário ou pela conveniência do serviço.

Fluxo de dados: conecta a saída de um objecto ou de um processo a entrada de um outro objecto ou processo.

Grupo salarial: é o conjunto de índices salariais atribuídos a uma ou mais carreiras em função da sua complexidade, responsabilidade e requisitos habilitacionais exigidos.

Índice: é um número que, multiplicado pelo valor do índice 100 da respectiva carreira e dividido o valor por 100, indica o vencimento correspondente.

Informação de serviço: classificação do serviço prestado que avalia o mérito do funcionário no exercício das funções que lhe estão atribuídas, nomeadamente as qualidades de adaptação e de direcção, a capacidade de execução, a eficácia na organização do trabalho, a eficiência e o comportamento pessoal.

Ligação ou link: conexão física ou conceitual entre instâncias de objectos

Posse: é o acto solene realizado com a presença dos funcionários do respectivo serviço, em que o indivíduo nomeado ou funcionário promovido ou designado em comissão de serviço, é investido de direitos e deveres inerentes a sua categoria ou função.

Processo: transforma valores de dados.

Progressão: é a mudança de escalão dentro da respectiva classe ou categoria.

Promoção: é a mudança para a classe ou categoria imediatamente superior da respectiva carreira.

Provimento: é um despacho de nomeação, ou contrato, sujeito a visto do Tribunal Administrativo e a publicação no Boletim da Republica que constitui a relação de trabalho no aparelho de Estado.

Reconversão de carreira: mudança de carreira do funcionário, desde que possua os requisitos habilitacionais e profissionais exigidos no respectivo qualificador profissional.

Remuneração: é constituída pelo vencimento e suplementos.

Termo de contrato: formalidade de prestação de serviços no Estado que consiste num acordo de vontade das duas partes (contratante e contratado).

Transferência: afectação de um funcionário a tarefas em local diferente daquele em que se encontra a prestar serviço.

Vencimento: constitui a retribuição que se atribui a cada funcionário de acordo com a sua categoria ou função e, como contrapartida, dos serviços que este presta ao estado.

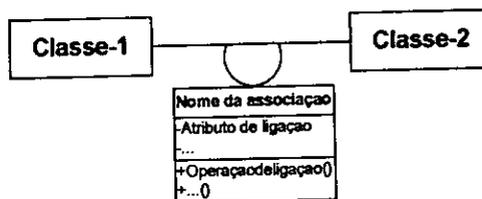
NOTAÇÕES OMT:

Notação do Modelo Objecto

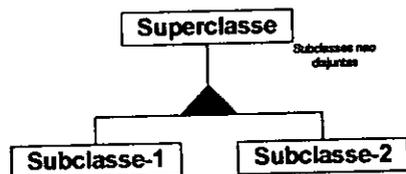
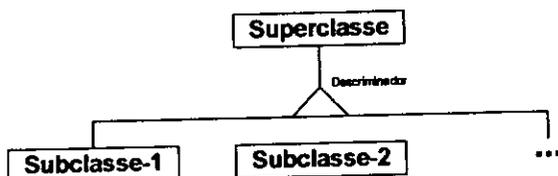
Associação:



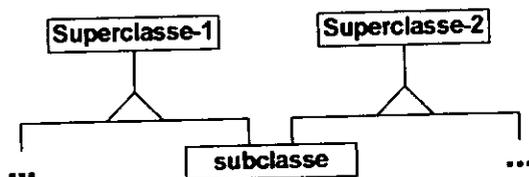
Associação como classe:



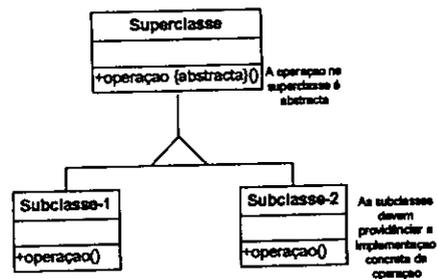
Propriedades da Generalização (Herança):



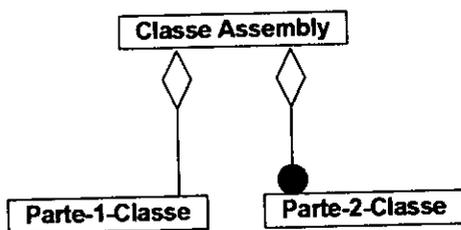
Herança Múltipla:



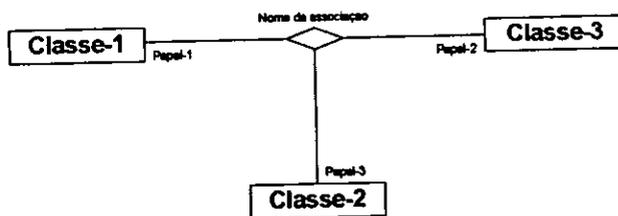
Operações abstractas:



Agregação:



Associação Ternária:

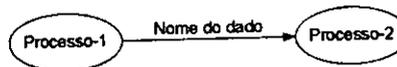


Notação do Modelo Funcional

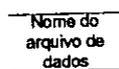
Processo :



Fluxo de dados entre processos :



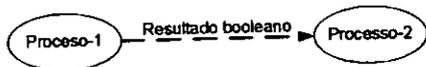
Arquivo de dados :



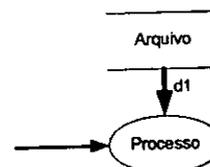
Objectos actores (como origem e destino de dados) :



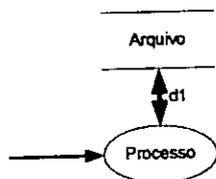
Fluxo de controle :



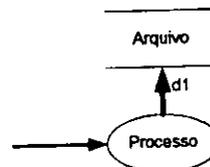
Acesso a valores no arquivo :



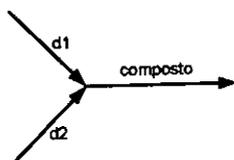
Acesso e Actualização de valores do arquivos :



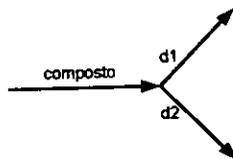
Actualização de valores no arquivo :



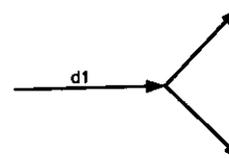
Composição do valor do dado :



Decomposição do valor do dado :



Duplicação do valor do dado :

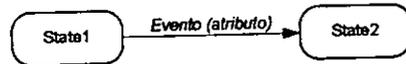


Notação do Modelo Dinâmico

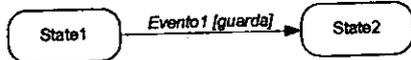
Evento causa transição entre estados:



Evento com atributo:



Transição condicionada:



Ação numa transição:



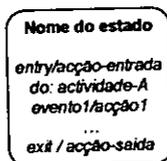
Estado inicial e final:



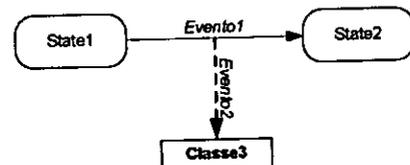
Resultado do evento numa transição:



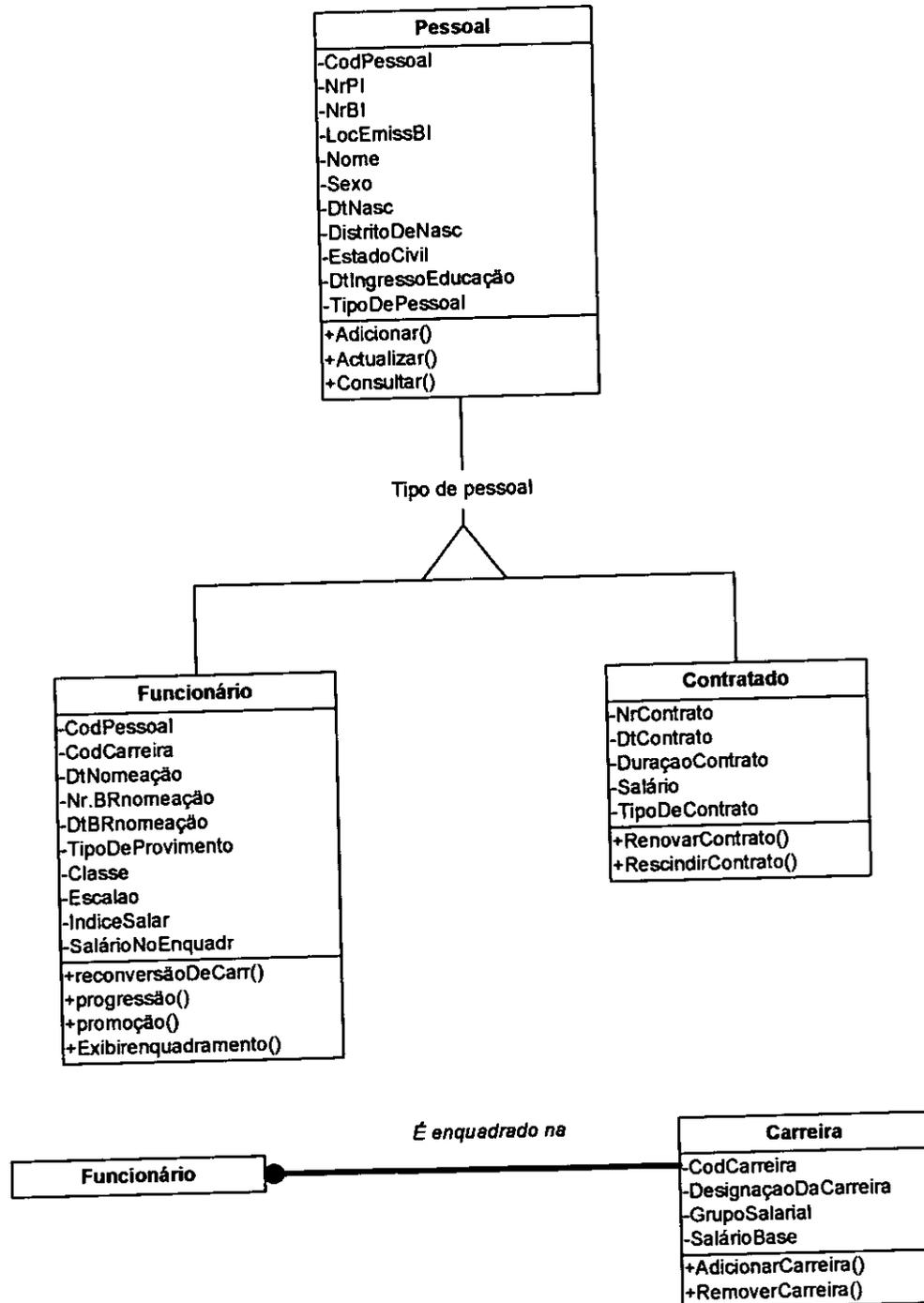
Ação e actividade durante um estado:

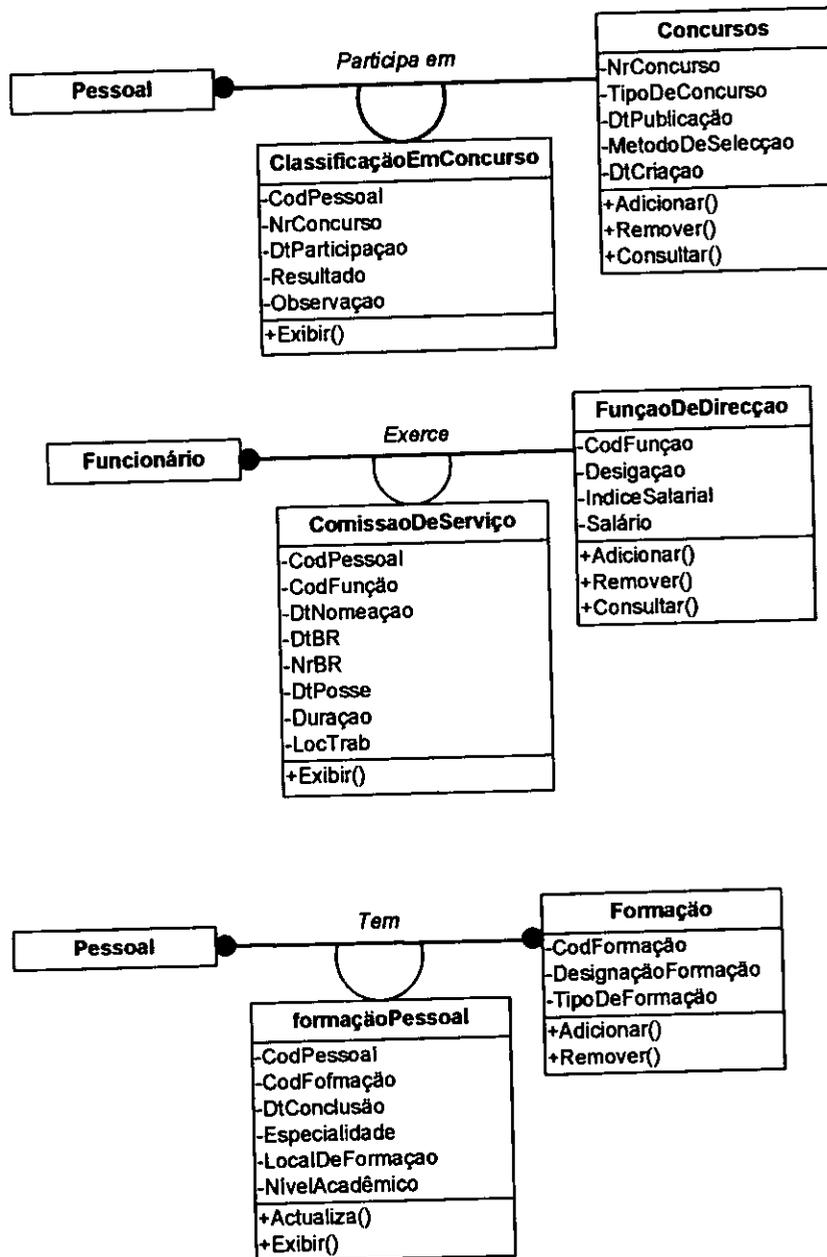


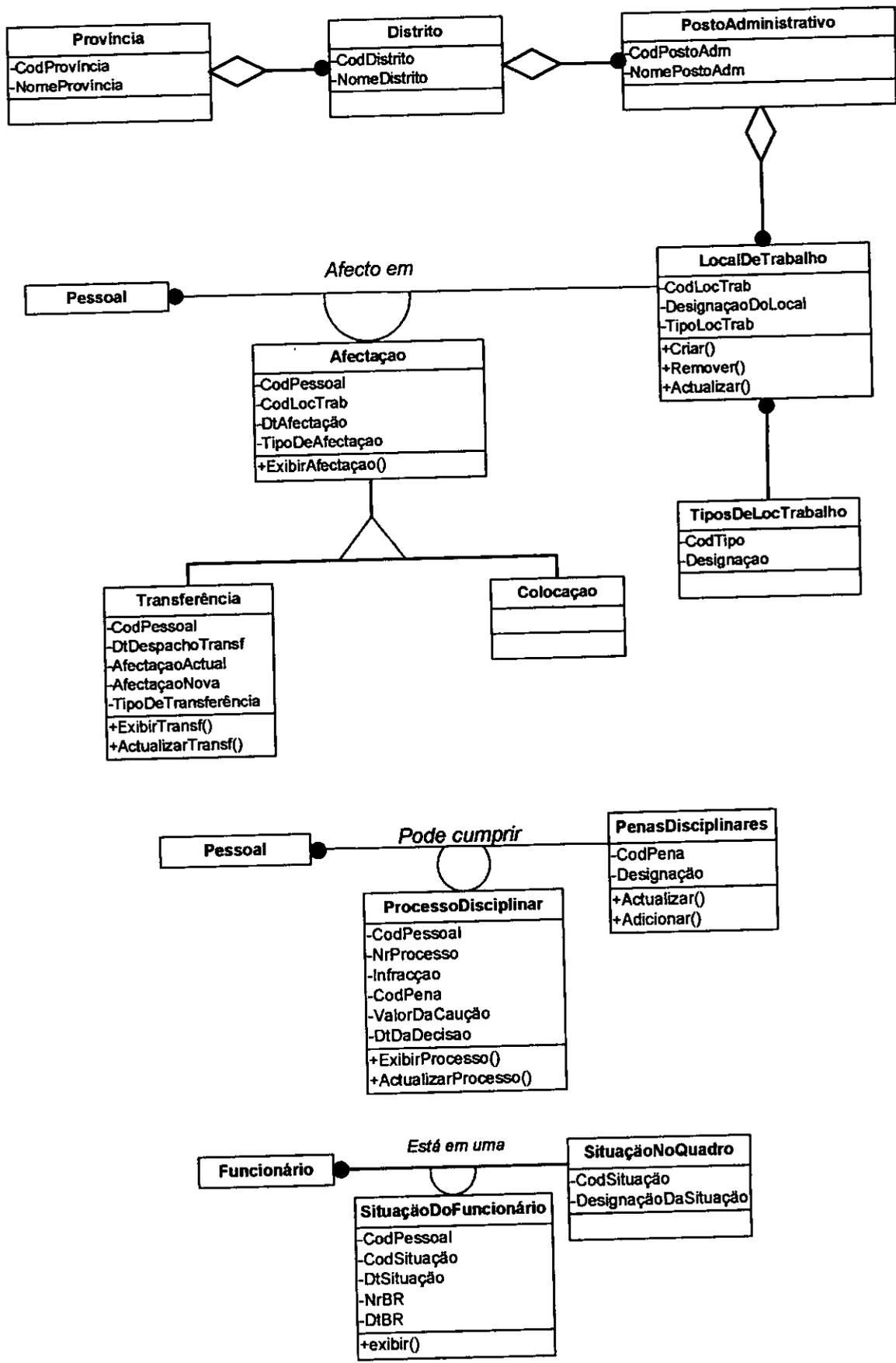
Envio de um evento a um objecto:

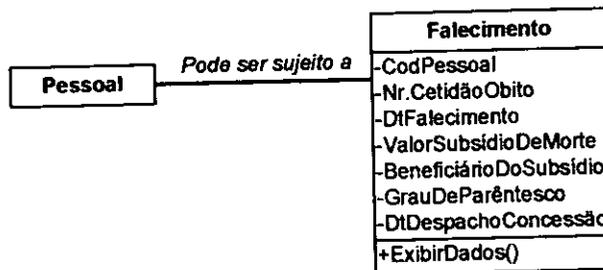
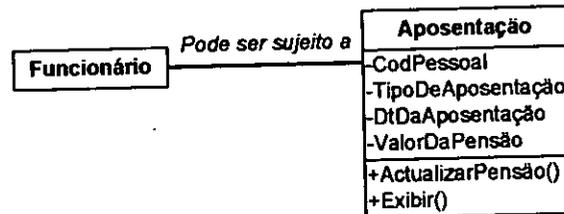
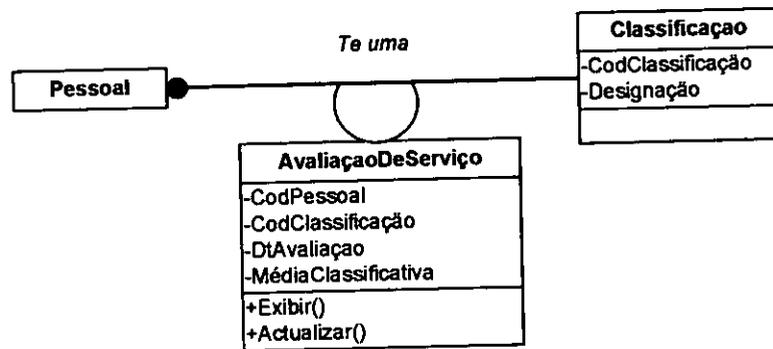


Anexo II - Modelo objecto detalhado

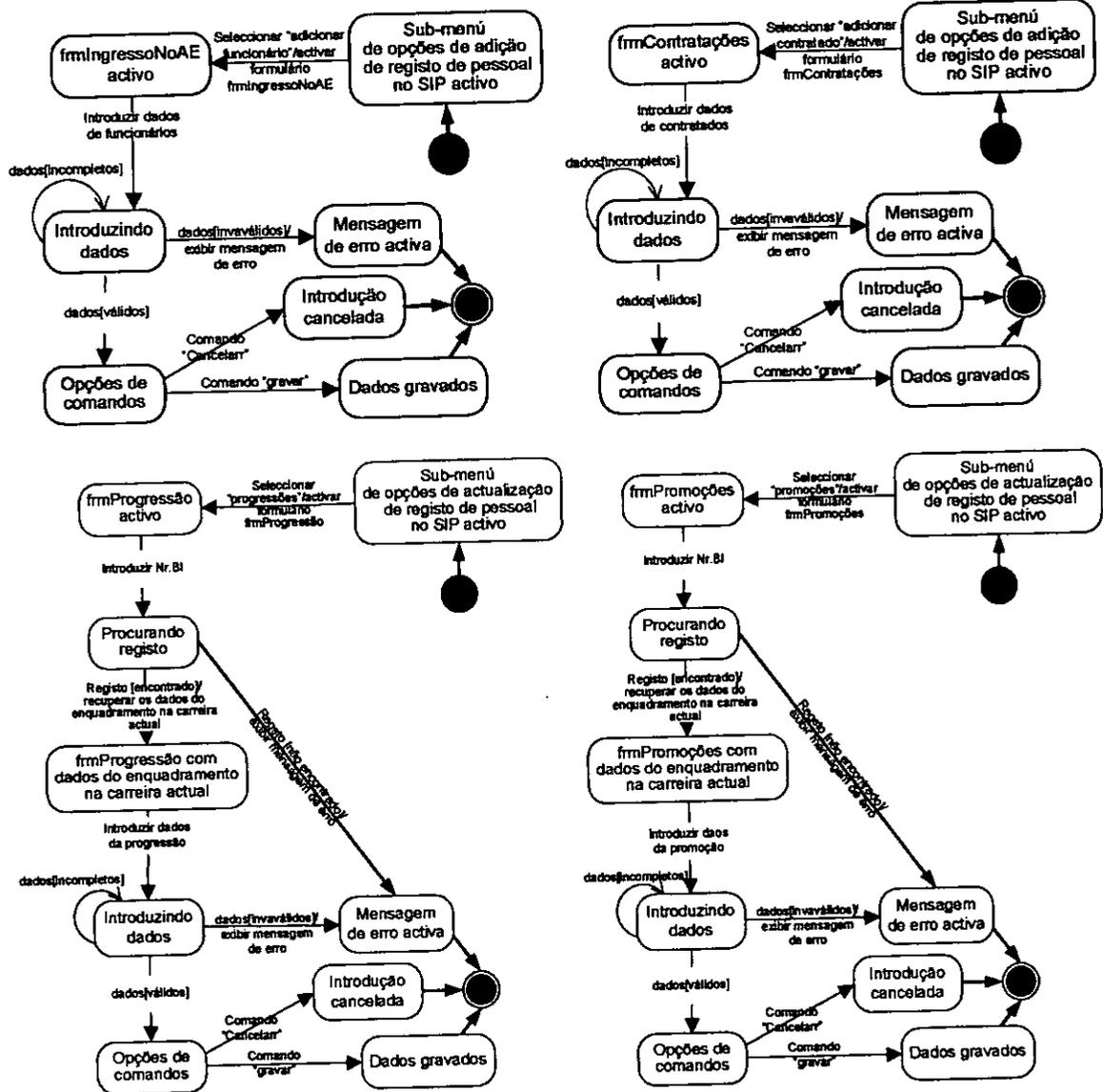


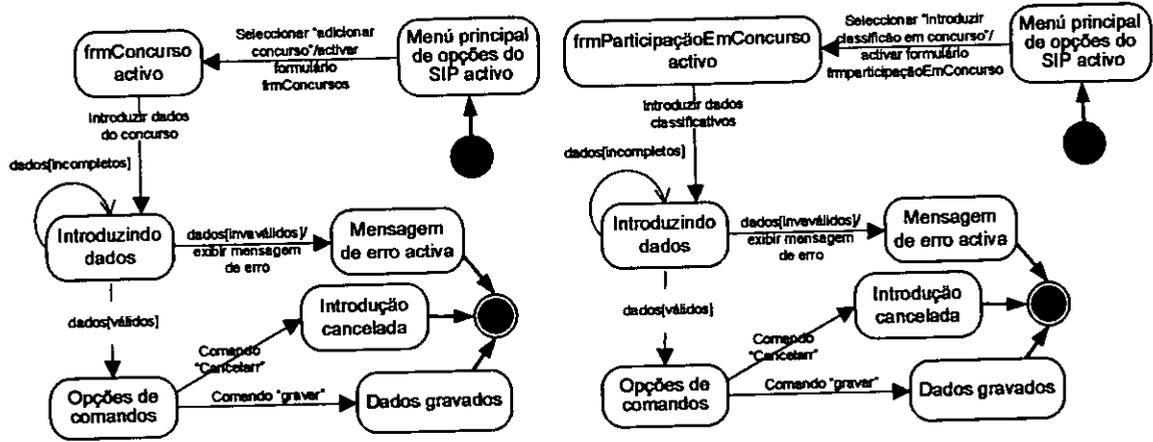
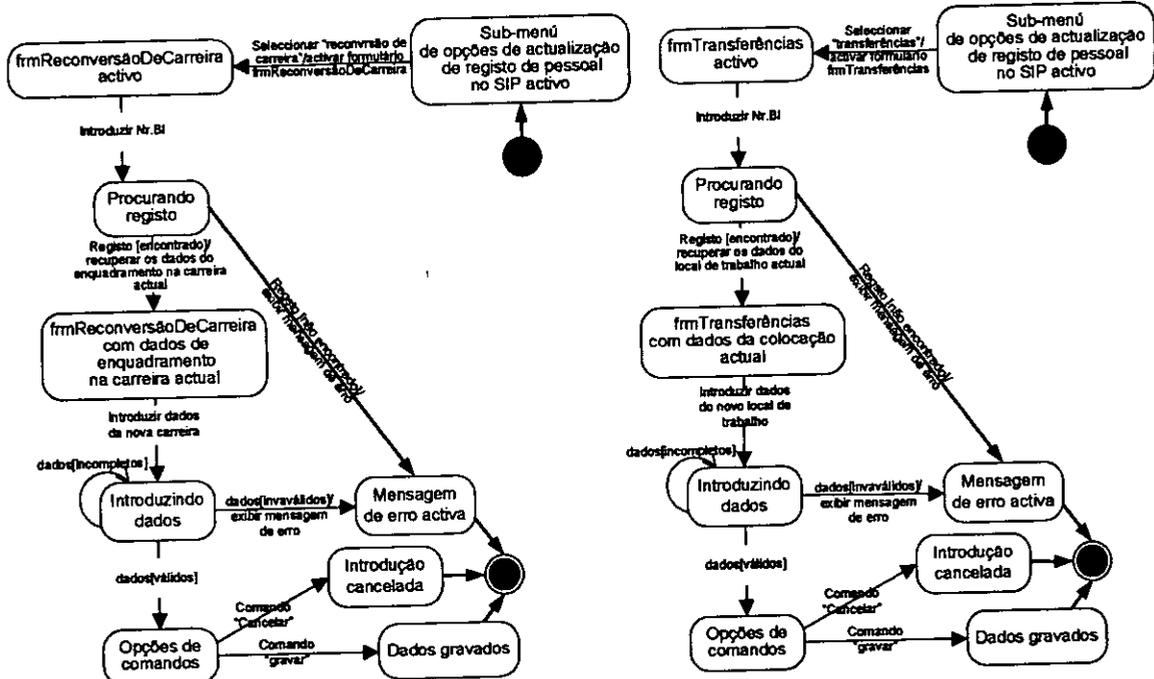


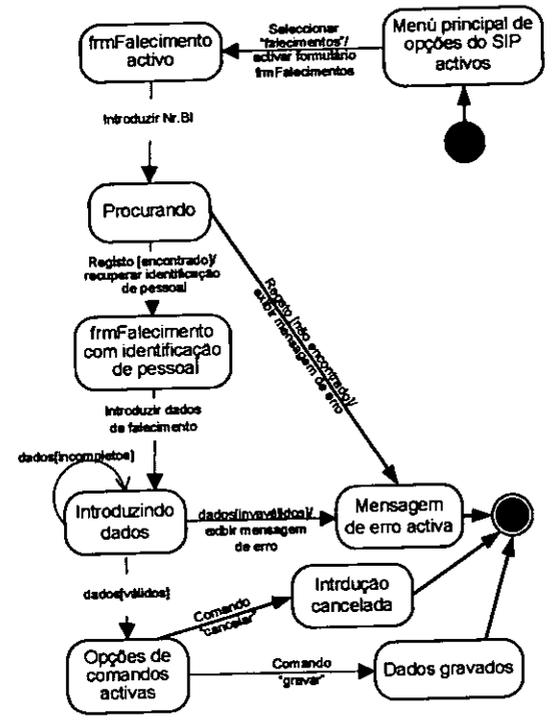
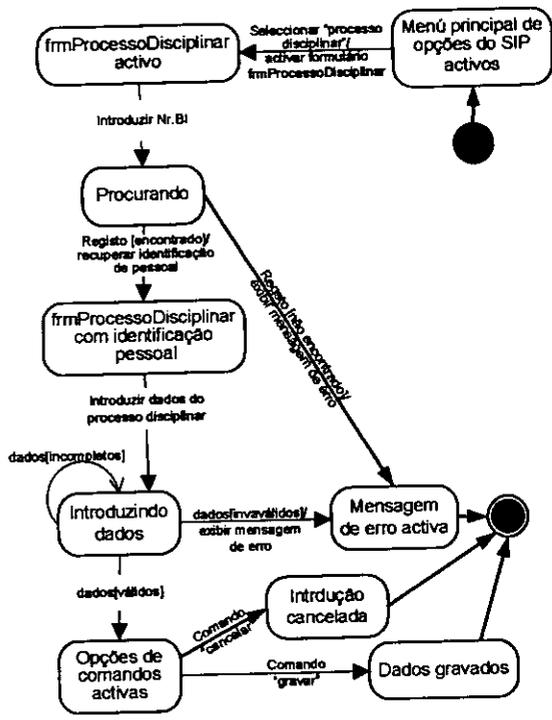
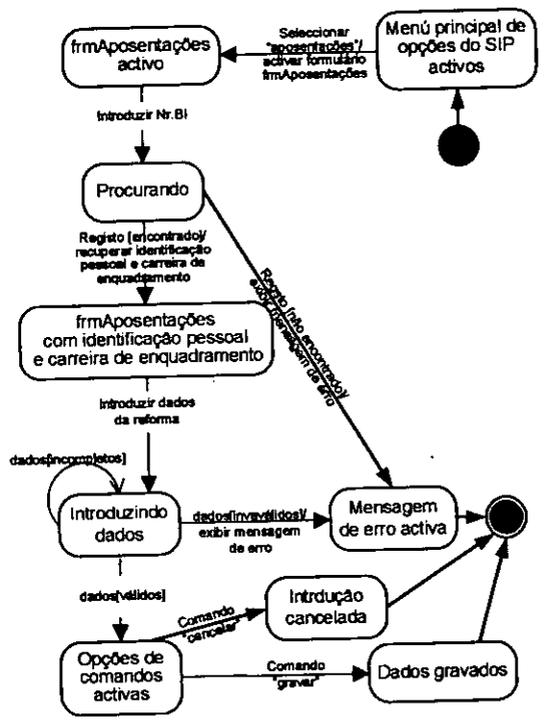
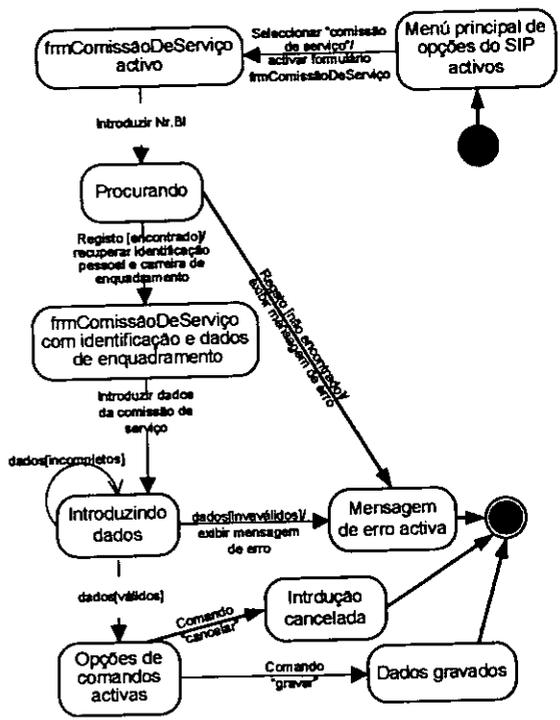


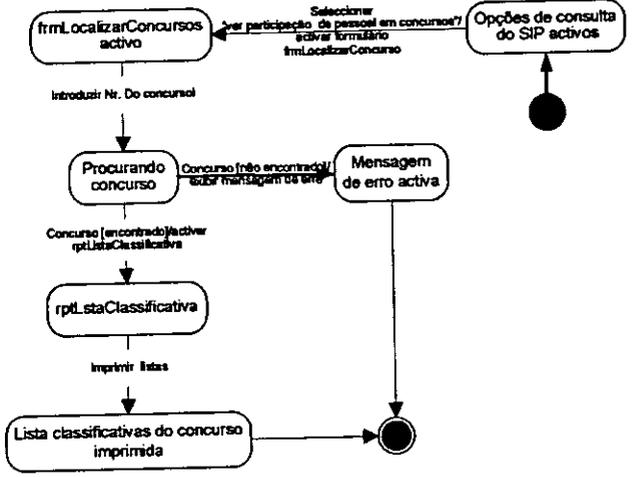
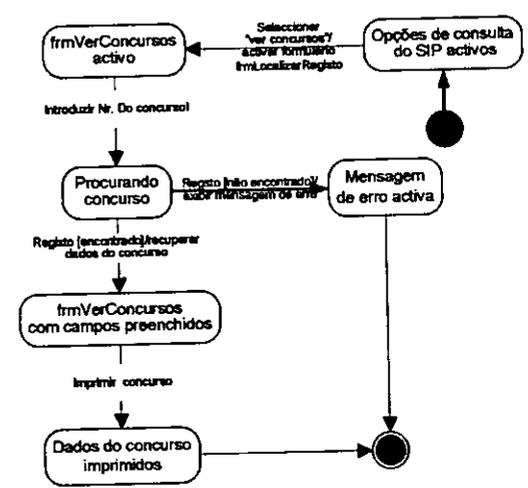
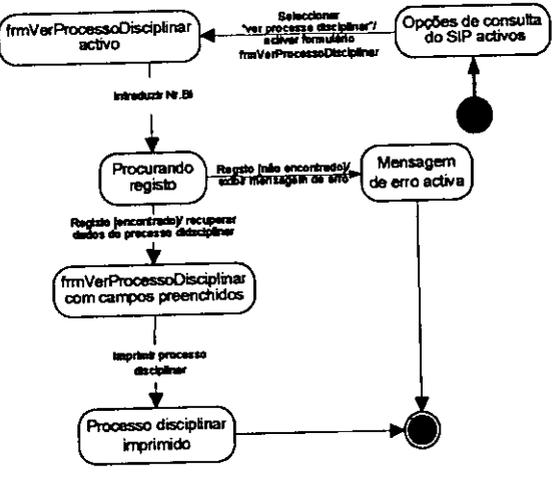
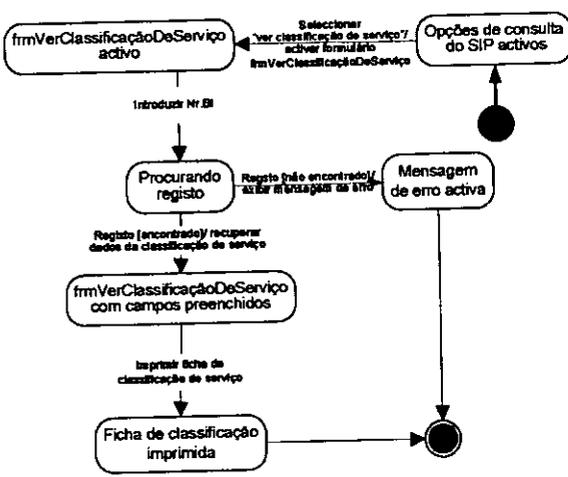
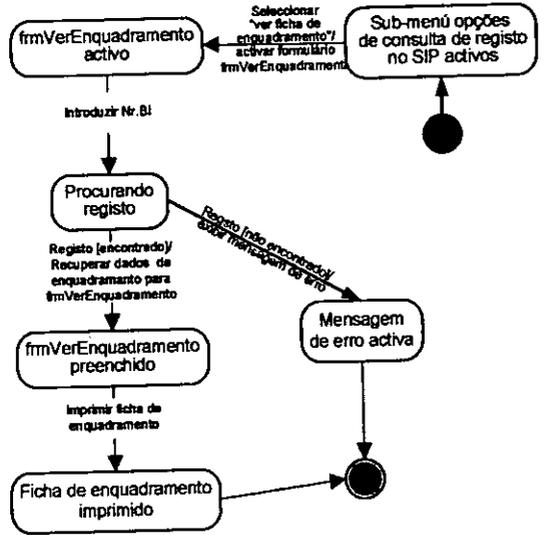
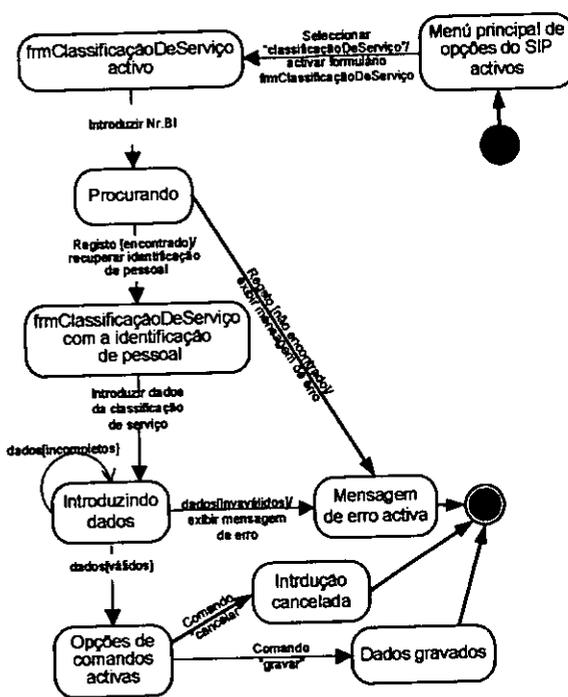


Anexo III - Modelo dinâmico detalhado



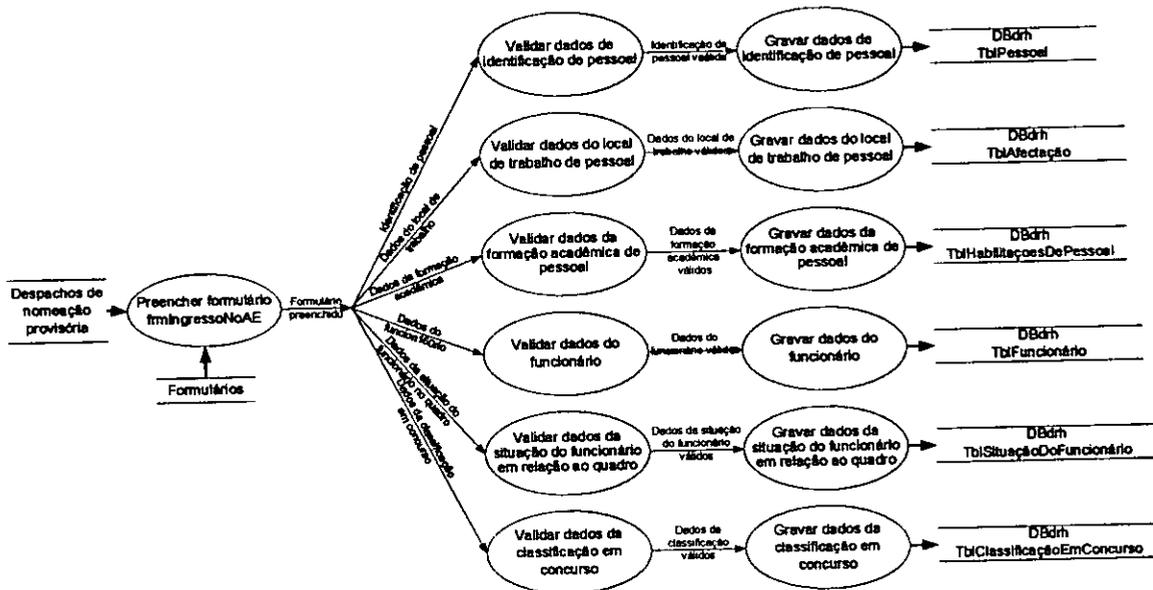




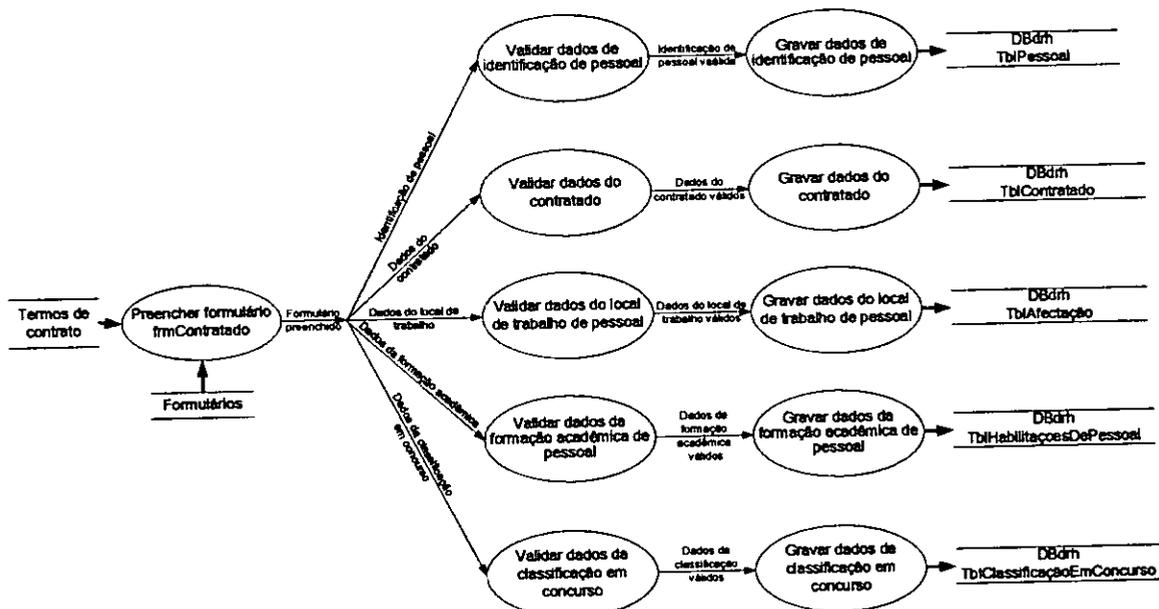


Anexo IV - Modelo funcional detalhado

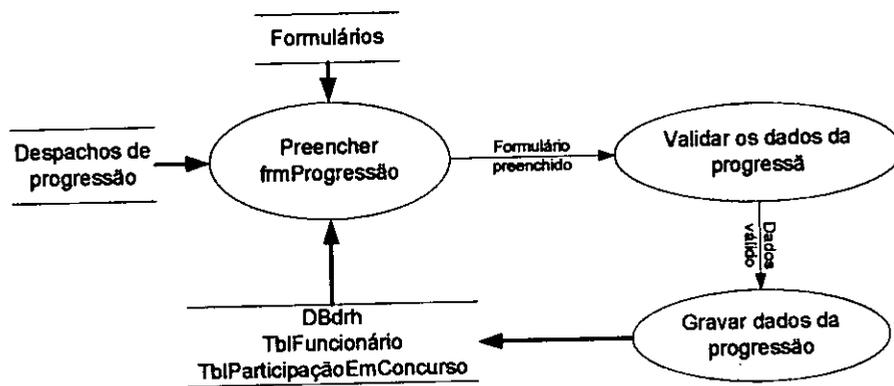
DFD detalhado do processo "Adicionar registo de funcionário":



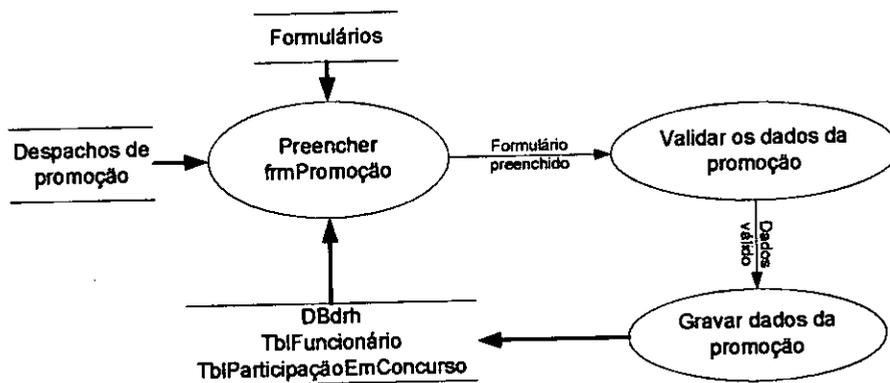
DFD detalhado do processo "Adicionar registo de contratado":



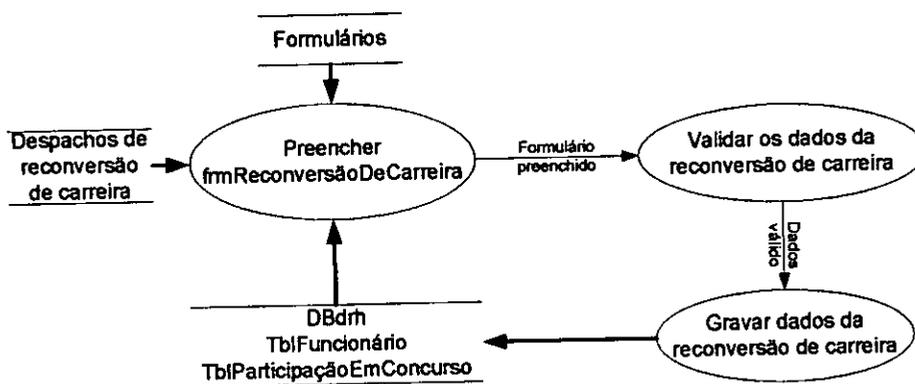
DFD detalhado do processo "Actualizar registo de funcionário (progressão)":



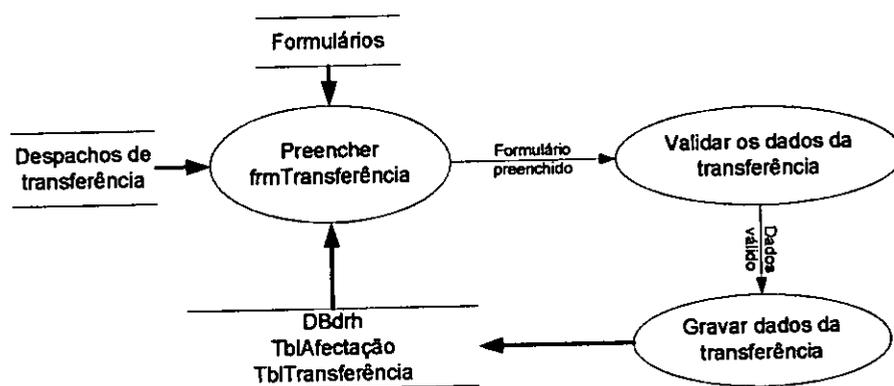
DFD detalhado do processo "Actualizar registo de funcionário (promoção)":



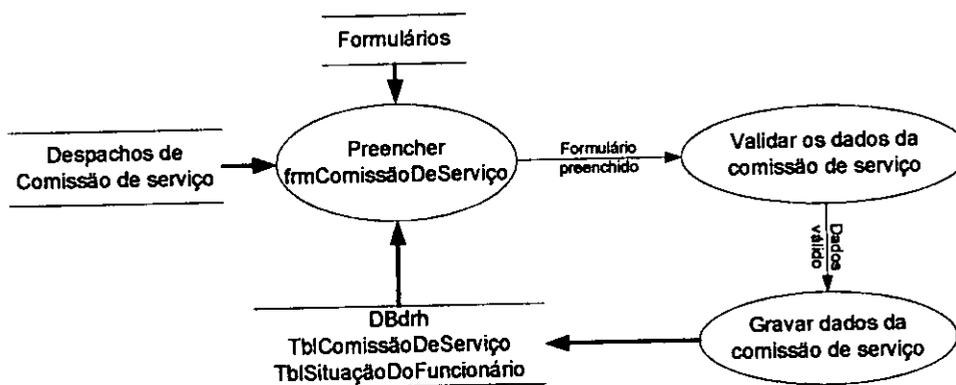
DFD detalhado do processo "Actualizar registo de funcionário (reconversão de carreira)":



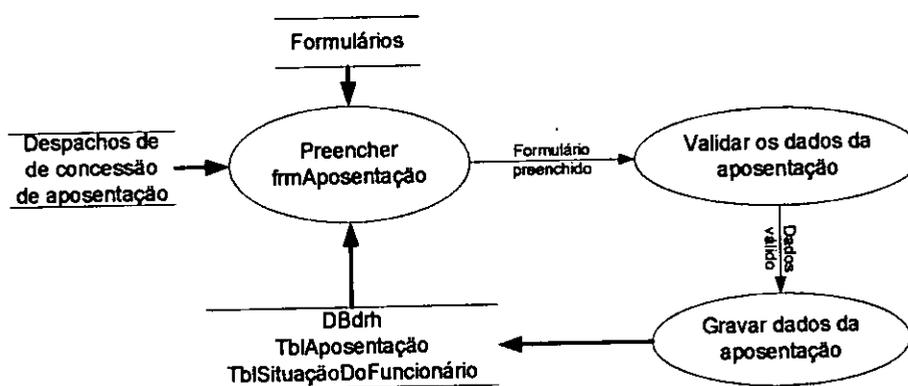
DFD detalhado do processo "Actualizar registo de funcionário (transferência)":



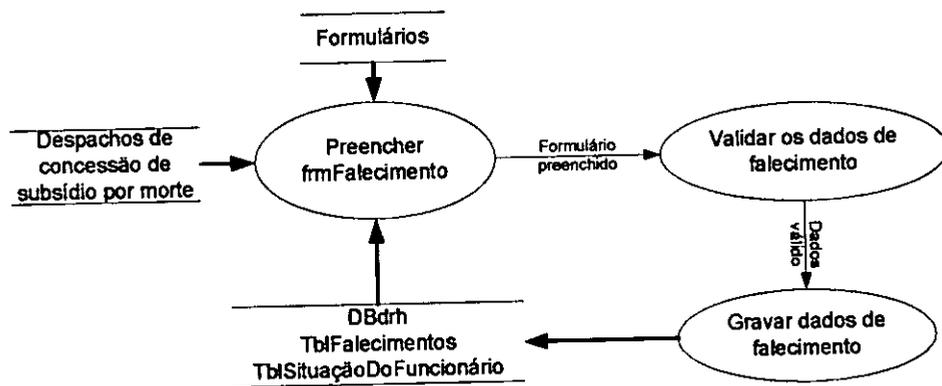
DFD detalhado do processo "Registar comissão de serviço":



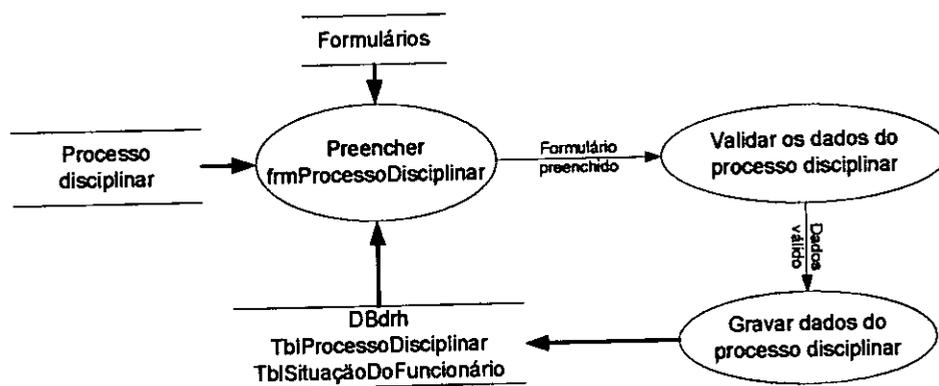
DFD detalhado do processo "Registar reforma":



DFD detalhado do processo "Registrar falecimento":



DFD detalhado do processo "Registrar processo disciplinar":



Categories
CdCat
NmCat
Desconhecida
TemClasses
NvLReq
CdSubCat
Duvida

CategsClasse
CdCateg
NmCateg
NmClasse
LetraSalarial
CdCat
NvLReq
ClasseIngresso
Topo
CdCatSeg
CdAntigo
QuadroGeral

Personal Categories
PersID
Prof_ID
BR_Nr
DataIni
DataPosse

Personal Education
Pers_ID
CdEduc
EducaAnoFim
Educ_comment

SitQuadro
StatusID
StatusName
TemBR
SituaçãoDeChefia
PodeConcorrer

PostoAdministr
CdPostoAdm
NmPostoAdm
CdBonusGrupo
NmSedePostoA
CdDistrito
CdProv
SedePrincipal
SedeDistrital

Person
PersID
FieldID
BI
BI_emissão
sex
Nome_completo
Nasc_data
Nasc_Dist
Marital_status
IngrAEData
IngrAEBR_Nr
TipoFunc

Habilitações
CdEduc
NmEduc
RamoCurso
NivelEduc
TipoEduc
Duração

Personal SitQuadro
PersonID
StatusID
Status_data_inic
Status_data_fim
BR_nr

PersonalLocTrabs
PersID
Data_ini
LocTrab_tipo
LocTrab_Dist
LocTrab_cod
BR_Nr

Districts
DistrictID
District_Nom
Province_ID

Habiltações_Nivel
CdNivelEduc
NmNvlEduc

Habiltações_Tipos
CdTipoEduc
NmTipoEduc

Place of Work
Provice_ID
District_ID
CdPostoAdm
Localidade_ID
Loctrab
Loctrab_Name
Loctrab_Tipo
Loctrab_genero
NmLocTrab

Place of Work T
Loctrab_tipo
LocTrab_tipo_de

Provinces
Provice_ID
Provice_Nom

Personal Provimento
PersID
CdProvimento
DataProvimento
BRProvimento

Anexo VI – Dicionário de dados

Processos:

Anexo VI – Dicionário de dados

Processos:

Nome do processo: Tratar concursos	
Descrição:	<ol style="list-style-type: none">1. Elaborar e aprovar a proposta de abertura de concurso de ingresso2. Actualizar o arquivo "concursos"3. Elaborar o aviso e publicar na imprensa4. Nomear o júri que vai acompanhar o processo5. Receber as candidaturas6. Proceder a selecção dos candidatos segundo critérios estabelecidos7. Publicar as listas classificativas (pautas) na imprensa8. Enviar as pautas publicadas ao sector que trata de nomeações9. Organizar o processo do candidato seleccionado

Nome do processo: Tratar despachos (Ingresso no AE)	
Descrição:	<ol style="list-style-type: none">1. Receber as listas classificativas (pautas)2. Verificar o cabimento orçamental3. Elaborar a proposta de nomeação provisória4. Submeter a proposta à despacho do Director dos recursos humanos5. Enviar o despacho ao MPF/DPPF para obtenção do visto do tribunal administrativo, a DAF para efeito de processamento de salário6. Publicação do despacho em BR7. Actualizar o arquivo "Despacho de nomeação provisória"

Nome do processo: Tratar termos de contrato	
Descrição:	<ol style="list-style-type: none">1. Receber as pautas2. Verificar o cabimento orçamental3. Elaborar os termos individuais de contrato4. Proceder a assinatura de contrato5. Enviar os contratos ao MPF/DPPF para obtenção do visto do tribunal administrativo, a DAF para efeito de processamento de salário6. Publicação dos contratos em BR7. Actualizar o arquivo "Termos de contrato"

Nome do processo: Tratar despacho de progressão	
Descrição:	<ol style="list-style-type: none">1. Seleccionar os os funcionários que satisfazem as condições para a progressão2. Verificar o cabimento orçamental3. Elaborar propostas individuais de progressão4. Submeter as propostas à despacho do Director dos recursos humanos5. Enviar os despachos ao MPF/DPPF para obtenção do visto do tribunal administrativo, a DAF para efeito de actualização de salário6. Publicar os despachos em BR7. Actualizar o arquivo "Despacho de progressões"

Nome do processo: Tratar despacho de promoção

- Descrição:**
1. Receber as pautas
 2. Verificar o cabimento orçamental
 3. Elaborar a proposta de promoção
 4. Submeter a proposta à despacho do Director dos recursos humanos
 5. Enviar o despacho ao MPF/DPPF para obtenção do visto do tribunal administrativo, a DAF para efeito de actualização de salário
 6. Publicar o despacho em BR
 7. Actualizar o arquivo "Despachos promoção"

Nome do processo: Tratar despacho de reconversão de carreira

- Descrição:**
1. Receber as pautas
 2. Verificar o cabimento orçamental
 3. Elaborar a proposta de reenquadramento na carreira
 4. Submeter a proposta à despacho do Director dos recursos humanos
 5. Enviar o despacho ao MPF/DPPF para obtenção do visto do tribunal administrativo, a DAF para efeito de actualização de salário
 6. Publicar o despacho em BR
 7. Actualizar o arquivo "Reconversão de carreiras"

Nome do processo: Tratar despacho de transferência

- Descrição:**
1. Receber pedidos de transferência
 2. Analisar os pedidos
 3. Elaborar proposta de transferência
 4. Submeter a proposta à despacho do Director dos recursos humanos
 5. Enviar os despachos ao MPF/DPPF para obtenção do visto do tribunal administrativo, a DAF para efeito de transferência de salário
 6. Actualizar o arquivo "Transferências"

Nome do processo: Tratar despacho de de comissão de serviço

- Descrição:**
1. Elaborar proposta de nomeação de pessoal para exercício de funções de direcção
 2. Submeter a proposta à despacho do Director dos recursos humanos
 3. Enviar o despacho ao MPF/DPPF para obtenção do visto do tribunal administrativo, a DAF para efeito de actualização de salário
 4. Publicar o despacho em BR
 5. Actualizar o arquivo "comissões de serviço"

Nome do processo: Tratar despacho de concessão de aposentação	
Descrição:	<ol style="list-style-type: none"> 1. Receber documentos para efeitos de aposentação 2. Organizar o processo de aposentação 3. Remeter o processo a despacho da concessão da aposentação 4. Publicação do despacho e contagem de tempo em BR 5. Elaborar nota de envio para fixação da pensão 6. Enviar a nota a MPF/DPPF 7. Atualização do arquivo "Despacho de concessão de aposentações"

Nome do processo: Tratar despacho de concessão de subsídio por morte	
Descrição:	<ol style="list-style-type: none"> 1. Receber os pedidos de subsídio, com o anexo de certidão de óbito do trabalhador falecido 2. Submeter os pedidos à despacho da concessão do subsídio 3. Elaborar a nota de envio do processo para o MPF/DPPF 4. Actualizar o arquivo "Despacho de concessão de subsídio por morte"

Nome do processo: Tratar despacho de processo disciplinar	
Descrição:	<ol style="list-style-type: none"> 1. Receber a participação da infração 2. Formular os autos 3. Designar o instructor do processo disciplinar 4. Instruir o processo disciplinar 5. Submeter o processo disciplinar a despacho da decisão 6. Actualizar o arquivo "processos disciplinares"

Nome do processo: Tratar despacho de exoneração	
Descrição:	<ol style="list-style-type: none"> 1. Receber pedidos de exoneração 2. Analisar os pedidos 3. Elaborar proposta de exoneração 4. Submeter a proposta à despacho do Director dos recursos humanos 5. Puplicar os despachos em BR 6. Actualizar o arquivo " Despachos de exonerações"

Arquivos:

Nome do arquivo: Despacho de nomeação provisória
Aliases: Nenhum
Composição: nome do funcionário + carreira + classe + escalão + unidade orgânica onde vai ser colocado + Nr. Do concurso + classificação no concurso + lugar criado e não provido + data do despacho + Nr. Do BR + data do BR + forma de provimento + data do visto do TA + cópia do BI+ certificado de habilitações

Nome do arquivo: Termos de contrato
Aliases: Nenhum
Composição: nome do funcionário + naturalidade + Nr. BI + data do BI + local de emissão + actividade que exerce + unidade orgânica onde vai prestar serviço + Nr. Do concurso + classificação no concurso + data do contrato + Nr. Do contrato + salário mensal + validade do contrato + habilitações literárias + data do visto do TA + cópia do BI+ certificado de habilitações

Nome do arquivo: Despacho de progressão
Alíases: Nenhum
Composição: nome do funcionário + escalão actual + classe + carreira + graduação na avaliação de potencial + escalão para que transita + data do despacho + artigo do Decreto + Nr. BR + data do BR + data do visto do TA

Nome do arquivo: Despacho de promoção
Alíases: Nenhum
Composição: nome do funcionário + Nr. Do concurso + classificação em concurso + carreira + classe actual + classe para que transita + escalão + data do despacho + Nr. Do BR + data do BR + data do visto do TA

Nome do arquivo: Despacho de reconversão de carreira
Alíases: Despacho de nomeação para nova carreira
Composição: nome do funcionário + carreira actual + carreira para que transita + classe + escalão + data do despacho + Nr. Do BR + data do BR + data do visto do TA

Nome do arquivo: Despacho de transferência
Alíases: Nenhum
Composição: nome do funcionário + carreira + local onde está colocado + tipo de transferência + local para onde é transferido + data do despacho + data do visto do TA

Nome do arquivo: Despacho de nomeação em comissão de serviço
Alíases: Nenhum
Composição: nome do funcionário + carreira + função para que é nomeado + data do despacho + data da posse + Nr. Do BR + data do BR + data do visto do TA

Nome do arquivo: Processo disciplinar
Alíases: Nenhum
Composição: número do processo + nome do trabalhador + carreira + infração + pena aplicada + data da decisão

Nome do arquivo: Despacho da concessão de aposentação
Alíases: Nenhum
Composição: nome do funcionário + carreira ou função + local de trabalho + tipo de aposentação + artigo que lhe faculta a aposentação + tempo de serviço + data do despacho + vencimento + Nr. Do BR da contagem do tempo + data do BR + data do visto do TA

Nome do arquivo: Despacho de concessão do subsídio por morte
Alíases: Nenhum
Composição: Nome do beneficiário + grau de parentesco com o falecido + nome do funcionário + data da morte + carreira + unidade orgânica onde estava colocado + lugar que ocupava + valor do subsídio + data do despacho + Nr. Da certidão de óbito + certidão de casamento ou certidão de nascimento do requerente

Nome do arquivo: classificação de serviço
Alíases: Avaliação de serviço
Composição: nome do funcionário + nota classificativa + classificação qualitativa + data da avaliação