

**GEOVANE NOGUEIRA LIMA**

**GERENCIANDO DOCUMENTOS FISCAIS ATRAVÉS  
DE UM SISTEMA DE *WORKFLOW***

**Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.**

Orientadora

**Profa. Dra. Ana Cristina Rouiller**

**LAVRAS**

**MINAS GERAIS – BRASIL**

**2003**

**GEOVANE NOGUEIRA LIMA**

**GERENCIANDO DOCUMENTOS FISCAIS ATRAVÉS  
DE UM SISTEMA DE *WORKFLOW***

**Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.**

Aprovada em 10 de Dezembro de 2003

---

**Profa. Dra. Ana Cristina Rouiller**

**LAVRAS**

**MINAS GERAIS – BRASIL**

**2003**

*A Deus e aos meus pais Bráulio e Francisca*

*Agradeço em primeiro lugar e de forma muito especial á minha querida orientadora Ana Cristina Rouiller.*

*Agradeço também aos meus colegas Breno, Leonardo, Marcio, Roberto e Blade pelo apoio e pela dedicação ao projeto. A todos colegas do UFLATEC.*

*Agradeço aos meus colegas e amigos e professores da faculdade e aos colegas do Alojamento.*

*Agradeço aos meus irmãos Luciano e Liliane pelo incentivo.*

*Enfim, agradeço a todos aqueles que de forma direta ou indireta contribuíram para o desenvolvimento deste trabalho.*

## SUMARIO

<b>1.</b>	<b>INTRODUÇÃO.....</b>	<b>1</b>
1.1	Contextualização.....	1
1.2	Motivação.....	1
1.3	Proposta.....	2
1.4	Relevância.....	2
<b>2.</b>	<b>WORKFLOW .....</b>	<b>3</b>
2.1	Introdução .....	3
2.2	Classificação dos workflows segundo McCready92.....	4
2.3	Sistema gerenciador de workflow .....	8
<b>3.</b>	<b>PROPOSTA DE DESENVOLVIMENTO DO SISTEMA SANF/WORKFLOW.13</b>	
3.1	Visão Geral.....	13
3.2	Descrição do sistema.....	14
3.3	Metodologia .....	23
<b>4.</b>	<b>O DESENVOLVIMENTO DO SISTEMA SANF/WORKFLOW .....</b>	<b>31</b>
4.1	Concepção .....	31
4.2	Elaboração.....	31
4.3	Construção.....	35
4.4	Implantação .....	35
<b>5.</b>	<b>CONCLUSÃO .....</b>	<b>36</b>
<b>6.</b>	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>37</b>

## LISTA DE FIGURAS

Figura 1 -	Relacionamento entre workflows.....	7
Figura 2 -	Arquitetura de um WFMS centralizado.....	9
Figura 3 -	Arquitetura de um WFMS distribuído.....	10
Figura 4 -	Arquitetura de um WFMS heterogêneo.....	11
Figura 5 -	Arquitetura de um WFMS multi-empresa.....	12
Figura 6 -	Arquitetura do sistema SANF.....	15
Figura 7 -	workflow: Notas Fiscais de Material Estoque.....	18
Figura 8 -	Workflow: Nota Fiscal de Material Capex.....	19
Figura 9 -	Workflow: Nota fiscal de serviço e documento de cobrança.....	20
Figura 10 -	Workflow: Nota fiscal de transferência de material.....	21
Figura 11 -	Workflow: Nota fiscal de material.....	22
Figura 12 -	Modelo Incremental.....	24
Figura 13 -	Arquitetura de sistemas em três camadas.....	32
Figura 14 -	Diagrama de subsistemas.....	33
Figura 15 -	Modelo relacional da base de dados do SANF/workflow.....	34

## **Resumo**

O presente trabalho objetiva o desenvolvimento de um sistema gerenciador de documentos fiscais através de um sistema de *workflow*, com o intuito de automatizar o processo de verificação, aprovação e pagamento dos documentos fiscais, a aplicação das técnicas da Engenharia de Software ao desenvolvimento de sistemas.

Palavras – chave: Workflow, Desenvolvimento de software.

## **Abstract**

*The present objective work the development of a gerenciador fiscal document system through a system of workflow, with intention to automatize the process of verification, approval and payment of fiscal documents, application of the techniques of the engineering of software to the development of systems.*

*Word-key: workflow, software developer.*

## 1. INTRODUÇÃO

Este capítulo apresenta uma introdução ao tema que será abordado neste trabalho, contextualizando o desenvolvimento de software, apresentando os fatores que motivaram o seu desenvolvimento e mostrando qual a proposta do mesmo.

### 1.1 Contextualização

O presente trabalho se enquadra em uma sub-área da Engenharia de Software: desenvolvimento de sistema, denominada sistemas de *workflow*. É um trabalho direcionado ao desenvolvimento de software para controle de processos baseados em *workflow*.

Este trabalho está sendo realizado no Centro de Tecnológico de informática UFLA (UFLATEC), em conjunto com uma equipe de desenvolvimento formada por: três engenheiros de software e um especialista em *workflow*. Este sistema está sendo desenvolvido para um grupo do setor de telecomunicação, este grupo é constituído de três empresas, que estão presentes em dezesseis Estados do país.

### 1.2 Motivação

Uma das atividades crítica nas empresas de médio e grande porte é o controle de pagamento de seus documentos fiscais. Quando não se tem o controle do pagamento destes documentos, muitos deles acabam sendo pagos com multa por atraso, enquanto outros são pagos antecipadamente.

A engenharia software propõe varias técnicas, modelos e processos para o desenvolvimento de sistema. O desenvolvimento de um sistema é uma boa



oportunidade para se aplicar tais técnicas e observar o comportamento das mesmas.

### **1.3 Proposta**

O principal objetivo deste trabalho é o desenvolvimento de um sistema de *workflow*, com o intuito de automatizar e monitorar o processo de pagamento de documentos fiscais, aplicando algumas das diversas técnicas da Engenharia de Software para o desenvolvimento de sistemas.

### **1.4 Relevância**

Este trabalho prevê o desenvolvimento de um software, que irá automatizar todo o processo de documentos fiscais dentro das empresas. O SANF irá garantir mais agilidade na aprovação, verificação e pagamento dos documentos fiscais.

## 2. WORKFLOW

Neste capítulo, será apresentada uma visão geral dos sistemas de *workflow*, descrevendo os conceitos básicos, as caracterizações existentes e uma arquitetura padrão para sistemas gerenciadores de *workflow*.

### 2.1 Introdução

Ainda há pouca concordância sobre o que é *workflow* e quais as características de um *workflow*. Segundo o projeto WIDE [WIDE2003], *workflow* é “um conjunto coordenado de atividades (seqüenciais ou paralelas) que são interligadas com o objetivo de alcançar uma meta comum”, sendo atividade conceituada como “uma descrição de um fragmento de trabalho que contribui para o cumprimento de um processo”. Assim, pode-se assumir que *workflow* é a divisão de um grande trabalho em várias tarefas menores, com pré-requisitos entre elas, que devem ser respeitados para o avanço da atividade. Já segundo [WFMC2003], *workflow* é: "A automação de um processo de negócios, por inteiro ou em parte, durante o qual documentos, informações e tarefas são passadas de um participante para outro por ação respeitando um conjunto de regras procedurais". Kobielus [Kobielus1997] apresenta a seguinte definição para *workflow*: “*Workflow* pode ser definido como o fluxo de informações e controle em um processo de negócio”.

No princípio, os sistemas de *workflow* limitavam-se a passar os diferentes itens de trabalho para os respectivos participantes de um processo, o qual deveriam executar tais itens como tarefas. Hoje, a tecnologia amadureceu, e o processo como um todo é automatizado, desde sua definição até o fluxo das informações de controle necessárias à execução de cada uma de suas etapas. Com isso, um sistema de *workflow* não é mais apenas um simples conjunto

coordenado de atividades de tarefas, mas é responsável pelo processo que lhe é definido [Tramontina2003]. O *workflow* se preocupa com a automação de processos onde documentos, informações ou tarefas são trocados entre os seus participantes, de acordo com um conjunto definido de regras e objetivos, esta é uma das formas atuais de se entender *workflow*.

## **2.2 Classificação dos workflows segundo McCready92**

Os *workflow* podem ser classificados de diversas formas, vamos apresentar apenas a classificação proposta por McCready, pois não é objetivo deste trabalho aprofundar-se no estudo de *workflow*. Os *workflow* são agrupados em três tipos: “*workflow ad-hoc*”, “*workflow* de produção ou transacional” e “*workflow* administrativo”, [McCready92].

Geralmente, as principais diferenças entre os sistemas de *workflow* apresentados estão relacionadas à robustez dos sistemas, à sua capacidade de automatizar processos considerados críticos para a organização, o seu preço e a complexidade para instalação.

### ***Workflow ad-hoc***

Sistemas de *workflow ad-hoc* são considerados os mais simples. Sua principal característica está na capacidade de automatizar processos não estruturados. São processos onde não há um conjunto padrão para mover as informações entre as pessoas. As tarefas tipicamente envolvem coordenação, colaboração e co-decisão humana, com pequenos times de profissionais. Sistemas de *workflow ad-hoc* geralmente não possuem capacidade de segurança e tratamento de grandes volumes de dados, e por este motivo não são recomendados para automação de processos críticos. Os sistemas de *workflow*

*ad-hoc* são baseados em troca de *e-mail* para o roteamento de informações. Usualmente possuem uma base de dados proprietária para armazenar as informações compartilhadas. Como exemplos de sistemas de *workflow ad-hoc* existem: *Lotus Notes (IBM)* e *MS exchange (Microsoft Corporation)*. Outra característica deste tipo de sistema é a sua facilidade de uso e de configuração.

### ***Workflow administrativos***

Os sistemas *workflow* administrativos envolvem processos repetitivos e previsíveis com regras de coordenação de tarefas simples. A ordem e coordenação de tarefas podem estar automatizadas, mas não suportam um processo de informação complexo e não requerem acesso a sistemas de informações múltiplos usados para suportar produção e/ou serviços a clientes. Ou seja, os *workflows* administrativos manipulam o roteamento de informações simples e funções de aprovação de documentos. Este tipo de sistema já possui a capacidade de tratar um volume maior de informações além de possuir recursos mais sofisticados de segurança.. Seus usuários podem ser comunicados, por exemplo, via *e-mail* quando do início e/ou término da tarefa, sem necessidade de acessar o *workflow*. Os *workflows* que suportam *workflows* administrativos são encarregados de atribuir tarefas para o próprio executor e coletar os resultados. São recomendados para a automação de processos administrativos, por exemplo: um processo de requisição de reembolso de despesas, roteamento de relatórios e controle de documentos fiscais. Outra característica importante dos sistemas de *workflows* administrativos é o fato de estarem geralmente associados a processos de tratamento de documentos, como normas e procedimentos e construção de relatórios, por exemplo.

### ***Workflow de produção***

Os sistemas de *workflow* de produção são os mais sofisticados encontrados no mercado. Envolvem processos repetitivos e previsíveis que suportam processos de informação complexos com acesso a sistemas de informação. As ordens e coordenação das tarefas podem ser totalmente automatizadas. Os *workflow* que suportam *workflows* de produção devem fornecer facilidades para a definição de dependências entre as tarefas, bem como o controle de execução da tarefa com pouca ou nenhuma interferência humana. São frequentemente críticos e devem trabalhar com a integração e interoperabilidade de sistemas de informação. Este tipo de *workflow* envolve processos complexos e altamente estruturados, cuja execução requer um número de transações acessando diferentes sistemas de informações. Os sistemas gerenciadores de *workflow* devem permitir a definição de esquemas complexos de *workflow* e devem fornecer um mecanismo eficiente e automatizado para manipular exceções. Estes sistemas são caros e de difícil implantação. Geralmente exigem a participação de especialistas para implementação da integração aos demais sistemas da organização.

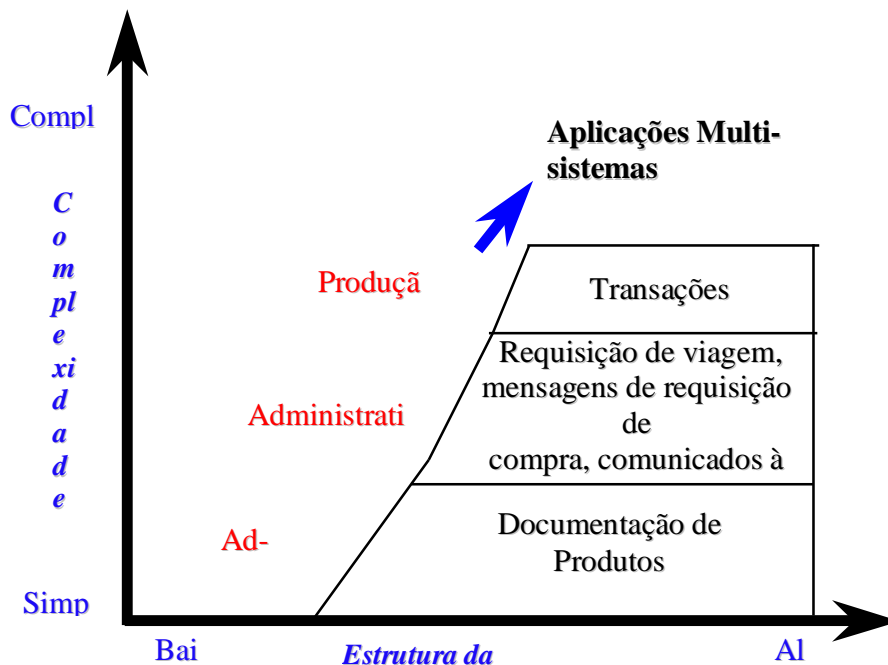


Figura 1 - Relacionamento entre workflows

A figura 1, ilustra o relacionamento entre os workflows definidos por [McCready92] usando a estrutura da tarefa versus sua complexidade. Alguns autores como Alonso [Alonso97] acrescentaram a classificação de McCready uma outra classe de workflows denominada *colaborativo*, que consideram principalmente o número de participantes envolvidos e a interação realizada entre eles. Estes workflows podem envolver diversas interações sobre a mesma tarefa até que os executores concordem com sua finalização.

### 2.3 Sistema gerenciador de workflow

Gerenciamento de *workflow* é a coordenação, controle e comunicação automatizada de trabalho para satisfazer os processos de *workflow*. Um sistema gerenciador de *workflow* (*WFMS-Workflow Management System*) abrange um conjunto de ferramentas que fornece suporte para a definição, a representação, o monitoramento e a administração dos processos de *workflow*.

Os WFMSs, via de regra, suportam a modelagem, coordenam a execução e monitoram as atividades da organização. Em geral são ferramentas colaborativas que permitem ao administrador definir as estruturas dos processos de negócio, atribuindo equipes e hierarquias de funções na organização, dentro das quais estes processos são executados.

Os WFMSs evoluíram e a arquitetura destes sistemas não ficou para trás, hoje existem vários WFMSs implementados em várias arquiteturas. As arquiteturas podem ser divididas em quatro categorias: *WFMSs de arquitetura centralizada*, *WFMS distribuído*, *WFMS heterogêneo* e *WFMS multi-empresa*.

#### ***WFMS centralizados***

Os WFMSs centralizados, como ilustrado pela *Figura 2*, possuem uma arquitetura cliente-servidor, onde clientes enviam mensagens para o servidor de *workflow* solicitando a realização de alguma determinada ação (os executores das atividades também são requisitados da mesma forma). Estes WFMSs possuem seus dados (incluindo a definição dos processos de *workflow*) centralizados em um único SGBD (Sistema Gerenciador de Banco de Dados), e promovem robustez através das propriedades transacionais desses. O processamento é realizado em um único servidor de *workflow*, não efetuando balanceamento de carga, nem especificando servidores diferentes para

compartilhar determinada tarefa. Não trabalham em ambientes heterogêneos, e solicitam sempre interferência humana no caso de situações não previstas.

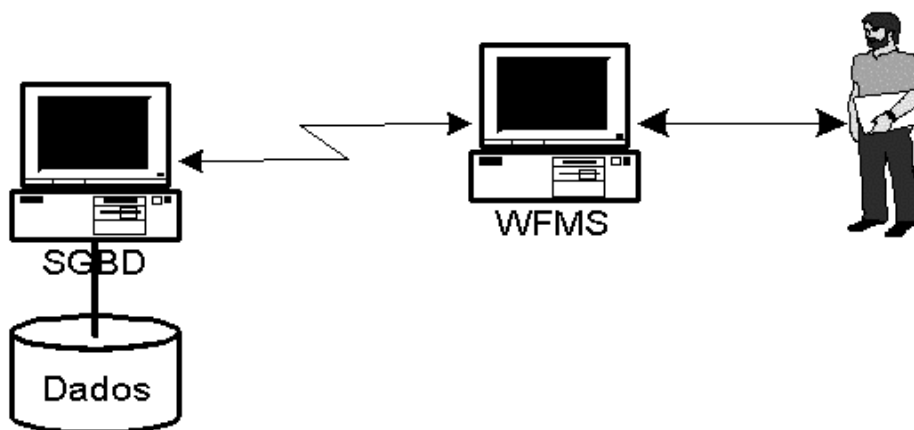


Figura 2 - Arquitetura de um WFMS centralizado

### ***WFMS distribuído***

Um WFMS distribuído, apresentado na *Figura 3 -*, permite que a execução de um processo de *workflow* seja realizada por diferentes servidores de *workflow*. Os WFMSs trocam mensagens para promover a execução de atividades e distribuírem sua carga de trabalho. Na fase de especificação do processo de *workflow* pode-se definir que partes do processo serão realizadas em quais servidores. Alguns WFMS distribuídos replicam dados e processos em diversos SGBDs e WFMSs para promover robustez e tolerância a falhas, mas não estão aptos a trabalhar em ambientes heterogêneos nem a se adaptarem a condições não previstas.

O Exotica/FMQM - *FlowMark on Message Queue Manager* [Alonso95] é um exemplo de WFMS que usa um único SGBD centralizado, o *ObjectStore*, mas distribui a execução dos processos entre os servidores de *workflow*. Outro



exemplo é dado em [Kamath95], onde servidores de *workflow* são agrupados acessando servidores de dados que possuem uma base de dados replicada em todos os SGBDs. Caso uma falha ocorra em um servidor, o processamento continua em outro que possui o processo replicado.

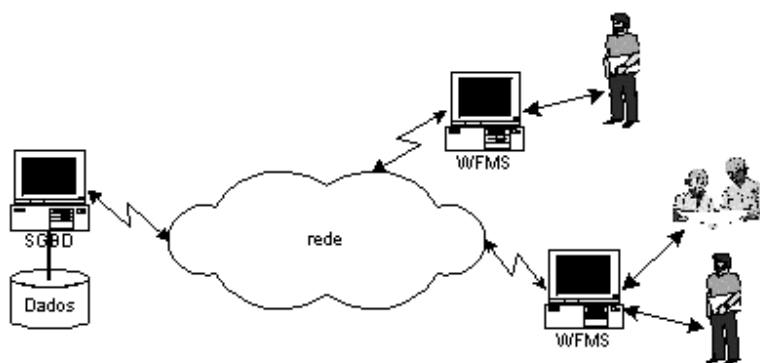


Figura 3 - Arquitetura de um WFMS distribuído

### ***WFMSs heterogêneos***

Os WFMSs heterogêneos, como mostrado na *Figura 4*, são aqueles que permitem tanto a distribuição do processamento para diversos servidores como o acesso a produtos de software distintos. Acessam sistemas legados, diversos SGBDs, entre outros softwares, através de uma interface de acesso denominada por muitos autores de camada de acesso. Um exemplo deste tipo de WFMS é o METEOR, que utiliza uma interface ORB-CORBA (*Object Request Broker - Common Object Request Broker Access*) para possibilitar um ambiente heterogêneo. Outro exemplo é o WIDE, utilizando o sistema FORO, que possui uma camada de acesso como interface de comunicação entre os componentes denominada BAL (*Basic Access Layer*). Este tipo de WFMS promove robustez através do uso das ATMs (*Advanced Transaction Management*), mas não permitem adaptação automática caso erros ou situações não previstas ocorram.

Devem conhecer a semântica da informação que manipulam, isto é, a camada que efetua a interface trata somente os aspectos sintáticos da informação.

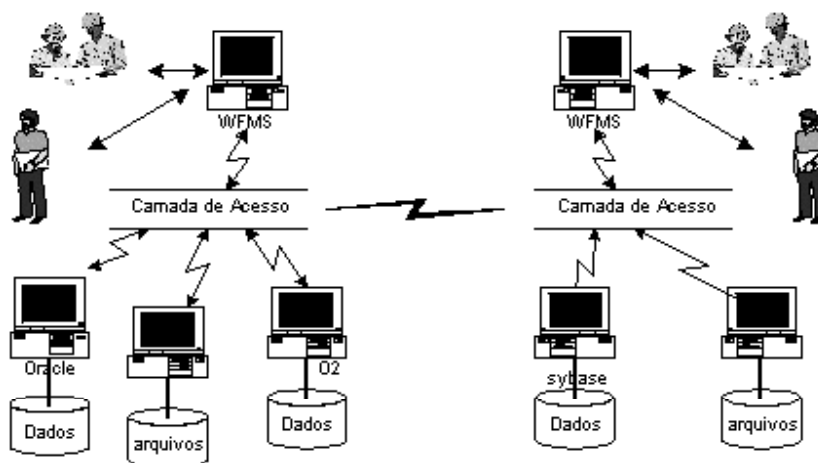


Figura 4 - Arquitetura de um WFMS heterogêneo

### ***WFMSs multi-empresa***

Os WFMSs multi-empresa, apresentado na *Figura 5*, trabalham com diversas empresas que podem possuir domínios de negócio diferentes. Um WFMS de uma empresa **A** “conversa” com o WFMS da empresa **B**, sem muitas vezes conhecer os *workflows* envolvidos no processo de negócio. Ser multi-empresa, significa poder distribuir tanto as definições como execuções dos *workflows*. Isto implica que, uma dada empresa poderá definir em seu *workflow* uma tarefa que será realizada em outras empresas, sem ter conhecimento do processo de *workflow* que realiza esta tarefa.

Como trabalham em um ambiente multi-empresarial, os WFMSs multi-empresa devem estar aptos a: negociar preços e prazos para realização de tarefas externas; decidir qual o melhor local para realização das tarefas; se adaptar a

situações não previstas; planejar execuções; aprender (sem necessidade de interferência humana) a respeito de novos procedimentos; promover atualizações estáticas e dinâmicas (durante a execução) das definições dos processos de *workflow* quando necessário; cooperar com outros WFMSs para a resolução de problemas; ter facilidades para projeto/reprojeto (promovendo o aproveitamento de definições de processos de *workflow* existentes em empresas externas e na própria empresa). Este tipo de WFMS exige menor interferência humana nas decisões e permite que os gerentes possam definir os processos com alto nível de abstração.

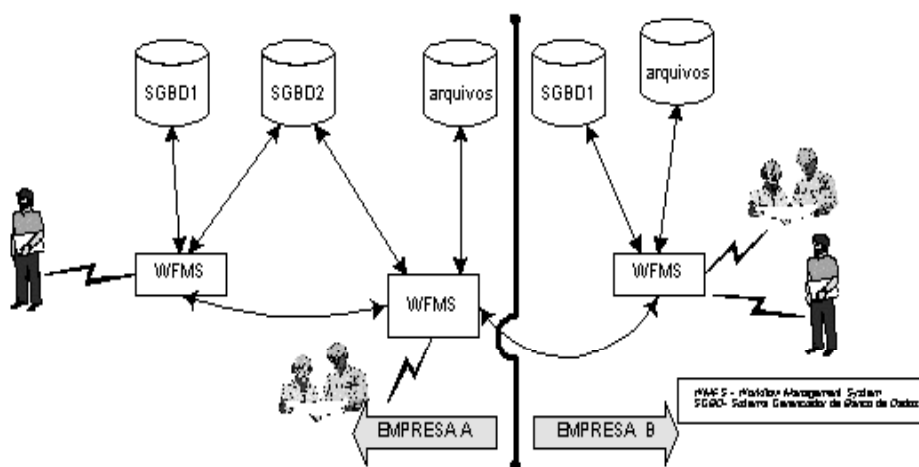


Figura 5 - Arquitetura de um WFMS multi-empresa

### **3. PROPOSTA DE DESENVOLVIMENTO DO SISTEMA SANF/WORKFLOW**

Será apresentada, neste capítulo, a proposta de desenvolvimento de um sistema de *workflow*, denominado de SANF/workflow (Sistema de Acompanhamento de Notas Fiscais). Apresentando, de forma geral, o sistema, a metodologia, o processo e o ambiente de desenvolvimento.

#### **3.1 Visão Geral**

Uma das atividades crítica nas empresas de médio e grande porte é o controle de pagamento dos seus documentos fiscais. Quando não se tem o controle do pagamento destes documentos, muitos deles acabam sendo pagos com multa por atraso, enquanto outros são pagos antecipadamente. Um outro problema gravíssimo é a fraude, que na maioria vezes não é percebida.

O sistema gerenciador de *workflow* aqui proposto: SANF/workflow (Sistema de Acompanhamento de Notas Fiscais), objetiva principalmente o controle de todo o fluxo de documentos fiscais dentro da empresa, automatizando e controlando o processo de aprovação, verificação e pagamento dos documentos fiscais; permitindo à empresa otimizar e controlar o pagamento dos seus documentos fiscais, evitando assim os prejuízos causados pelo descontrole.

Pela classificação proposta por Cruz [Cruz1998], poderíamos classificar o SANF/workflow como administrativo, uma vez que apresenta quase todas as principais características de um *workflow* administrativo, como por exemplo:

- Os processos são repetitivos e previsíveis, as regras de coordenação de tarefas simples são bem definidas;

- Trata uma quantidade significativa de dados e tem uma preocupação muito grande com a segurança;
- Os usuários são avisados via e-mail, sempre que inicia ou termina o fluxo de um documento fiscal.
- Todo o processo referente ao fluxo de documento fiscal é automatizado.

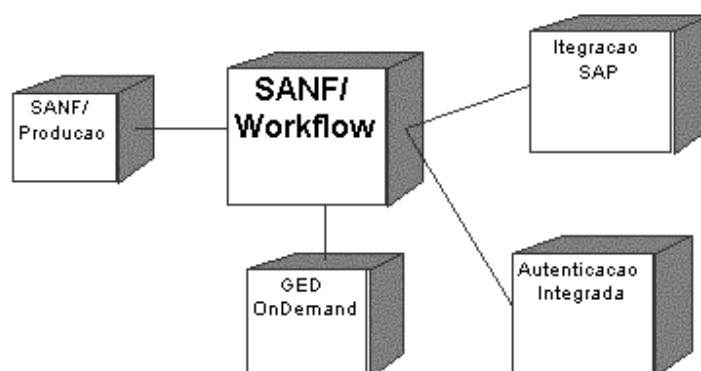
### **3.2 Descrição do sistema**

Nesta seção descreveremos, em linhas gerais, a arquitetura da aplicação, indicando seus módulos principais e a interação com o usuário e com outros sistemas.

O SANF/workflow é projetado para uma arquitetura distribuída, uma arquitetura do tipo cliente/servidor. A parte do servidor funciona em conjunto com um servidor *web*, o servidor web usado pelo SANF/workflow é IIS (*Internet Information Server*). O servidor IIS possibilita o uso da autenticação integrada. A parte do cliente pode ser acessada via *web browser*. A arquitetura cliente/servidor permite o acesso distribuído, garantindo que o sistema possa ser acessado de qualquer local da Intranet.

O sistema SANF é o conjunto de vários sistemas integrado como pode ser visto na figura 6. Fazendo integração com o sistema que controla o fluxo dos documentos fiscais na fase de produção (SANF/produção); com o sistema de GED (Gerenciador Eletrônico de Documentos); integra parte do processo com o sistema SAP/R3, que é um sistema de ERP (*Enterprise Resource Planning*) e têm a sua autenticação integrada com o *exchange* (sistema da *Microsoft* que armazena, entre outras informações, os dados dos usuários de uma rede de

computadores). E o SANF\workflow que é o sistema que controla o fluxo dos Documentos fiscais, é o SANF\workflow que faz a integração entre os outros sistemas. A interação entre os sistemas é ilustrada na *Figura 6*.



*Figura 6 - Arquitetura do sistema SANF*

A autenticação de usuário no SANF/workflow também é de forma integrada. No momento em que o usuário acessar a aplicação, o módulo de autenticação identifica o usuário que está logado no momento, em seguida verifica se o usuário que foi identificado é realmente usuário do sistema SANF/workflow. O sistema SANF/workflow trabalha com vários níveis de permissão. Os diferentes níveis de permissão estão relacionados com o perfil do usuário. Para cada perfil existe um conjunto de tarefas que são permitidas. Todo o usuário deve estar relacionado a um ou mais perfis.

A segurança do sistema é um dos pontos mais críticos. Pois, este processo está diretamente ligado à parte financeira da empresa. Toda a ação realizada no sistema ficará registrada na base do subsistema de Tranking<sup>1</sup>, desta

---

<sup>1</sup> Este subsistema é responsável pelo rastreamento do documento fiscal

forma é possível rastrear qualquer documento fiscal, bem como monitorar os usuários, medir a produtividade e eficiência dos usuários.

Vejamos uma rápida descrição do fluxo dos documentos fiscais, o fluxo inicia-se com a chegada dos documentos fiscais ao protocolo geral da empresa, onde é feita uma conferência visual dos dados a serem digitados; em seguida, o documento fiscal é encaminhado para o setor de produção da empresa onde ocorrerá a digitação e a digitalização dos mesmos. Os dados digitados irão alimentar a base do sistema SANF/workflow, enquanto as imagens, resultantes do processo de digitalização, serão armazenadas pelo sistema de GED (sistemas Gerenciadores Eletrônicos de Documentos) em seu repositório de imagens. Ao final do processo de digitalização e digitação dos dados do documento fiscal inicia-se o fluxo destes documentos no SANF/workflow. O documento deverá passar por diversas etapas do fluxo, tais como aprovação, aceitação e pagamento. O “caminho” a ser percorrido por um documento depende de vários fatores. O fluxo de um documento é finalizado quando o documento for pago, neste momento o sistema SANF/workflow registra esta informação no sistema de ERP(*Enterprise Resource Planning*) da empresa, esta primeira versão do sistema é o ERP compatível é O SAPR3.

As regras de redirecionamento dos documentos fiscais são bem definidas e praticamente todas estão vinculadas ao tipo do documento fiscal. Os documentos fiscais podem ser dos seguintes tipos:

- Nota Fiscal de Serviço;
- Nota Fiscal de Material;
- Nota Fiscal de Material Estoque;
- Nota Fiscal de Material Capex;
- Nota Fiscal de Serviço e Material;

- Nota Fiscal de Transferência de Material;
- Impostos, Taxas, Jurídicos e Contas;
- Documento de Cobrança.

As regras de redirecionamento foram definidas após estudo do tratamento que é dado aos documentos fiscais dentro das empresas. As regras foram descritas através de diagramas UML (*Unified Modeling Language*), para facilitar o seu entendimento as regras de redirecionamento variam segundo o tipo dos documentos fiscais, em função disto foi criado um diagrama para representar o fluxo de cada um dos tipos.

As regras de redirecionamento para os documentos fiscais do tipo: Nota Fiscal de Material Estoque podem ser visto na *Figura 7*

Nos diagrama os retângulos representam os pontos do fluxo, o nome na seta representa a ação realizada sob os documentos fiscais; o direcionamento da seta representa que o documento sai de um ponto para outro; o círculo preto representa o ponto de início do fluxo e o círculo duplo estado final.

A interpretação do fluxo apresentado na *Figura 7* é a seguinte:

1. O fluxo inicia e o documento é encaminhado para o SANF\produção (retângulo em laranja);
2. O sistema disponibiliza os dados para o SANF/workflow após sua validação;
3. Este tipo de documento é redirecionado para o estado “Aguardando Fechamento pelo Recebimento Físico Fiscal”;



Quando o documento estiver neste ponto ele pode ser encaminhado para outros três estados, o direcionamento do documento dependerá da ação realizada. Por exemplo: se a ação for “aceitar” o documento irá para estado que é “Incluído no SAP”.

### Fluxo Para Documento Fiscal: Nota Fiscal de Material Estoque

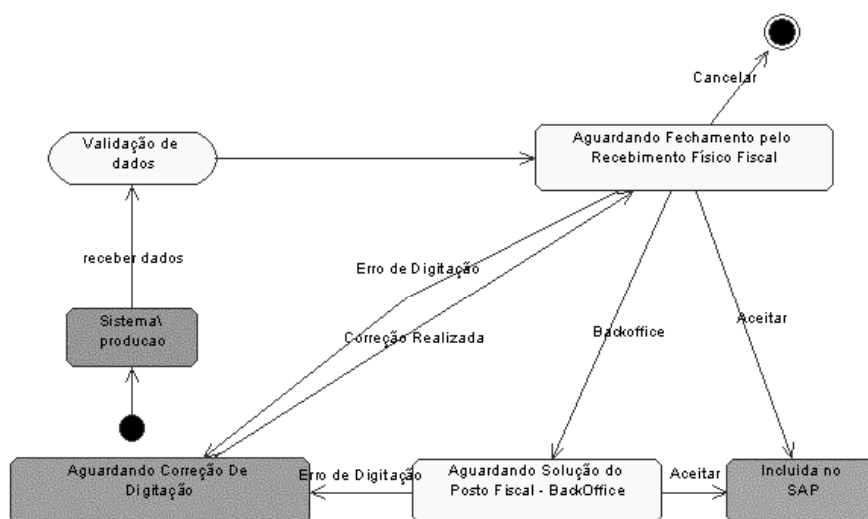
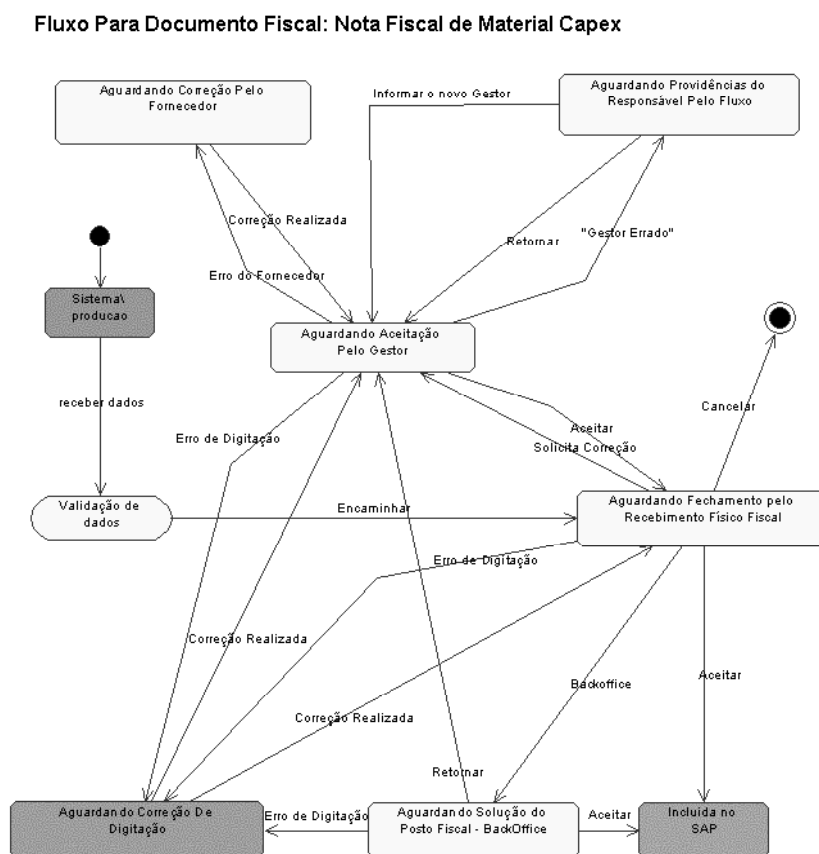


Figura 7 - workflow: Notas Fiscais de Material Estoque

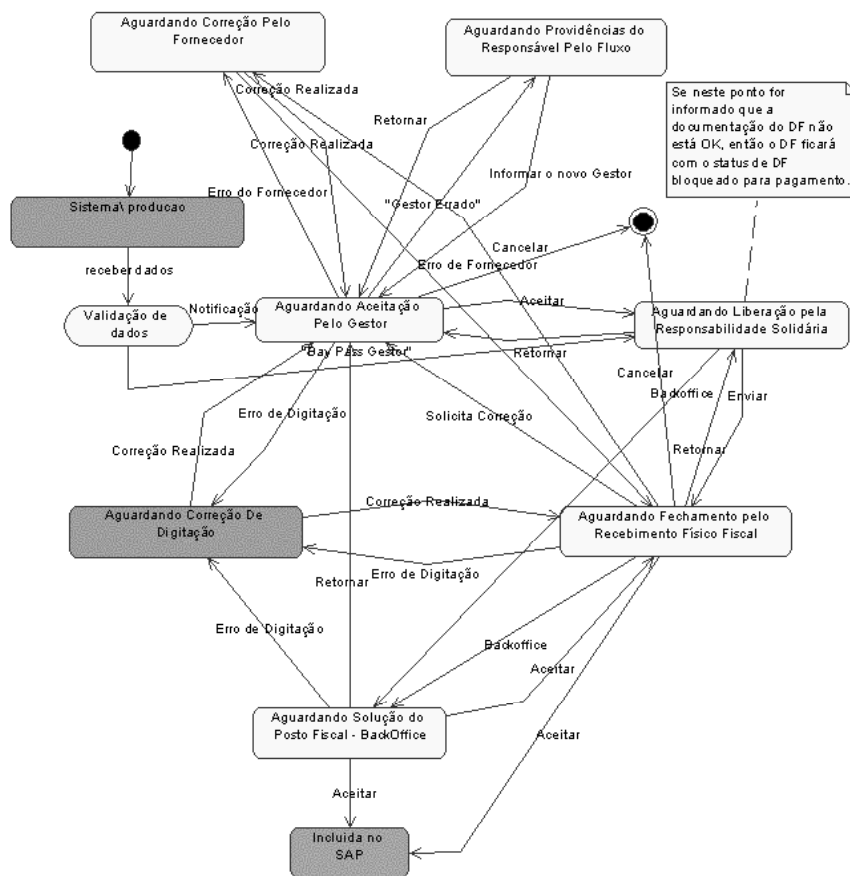
Na *Figura 8*, estão definidas as regras de direcionamento dos documentos do tipo: Nota Fiscal de Material Capex.



*Figura 8 - Workflow: Nota Fiscal de Material Capex*

A *Figura 9* apresenta o fluxo para os documentos fiscais dos tipos: Nota fiscal de serviço e documento de cobrança.

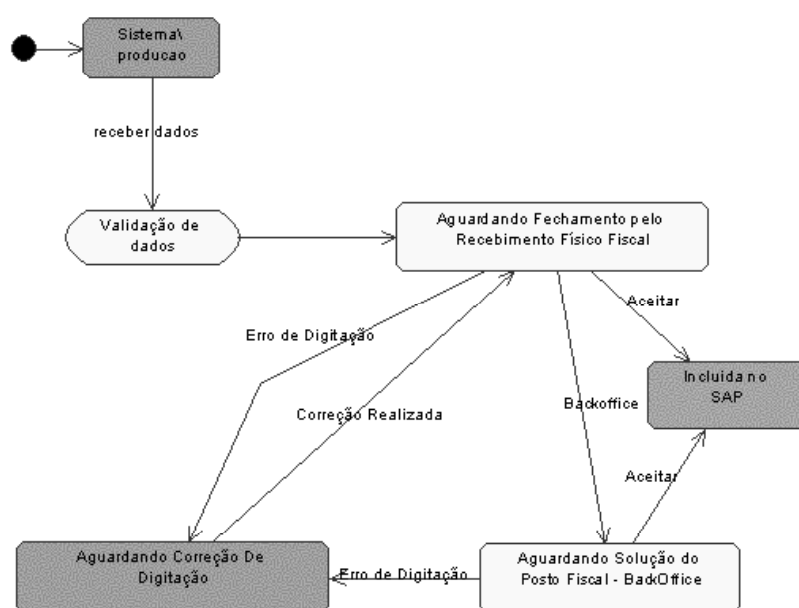
**Fluxo Para Documento Fiscal: Nota Fiscal de Serviço e Documentos de Cobrança**



*Figura 9 - Workflow: Nota fiscal de serviço e documento de cobrança*

A *Figura 10* apresenta o fluxo para os documentos fiscais do tipo: Nota fiscal de transferência de material.

### Fluxo Para Documento Fiscal: Nota Fiscal de Transferência de Material



*Figura 10 - Workflow: Nota fiscal de transferência de material*

A Figura 11 apresenta o fluxo para os documentos fiscais do tipo: Nota fiscal de material.

**Fluxo Para Documento Fiscal: Nota Fiscal de Material**

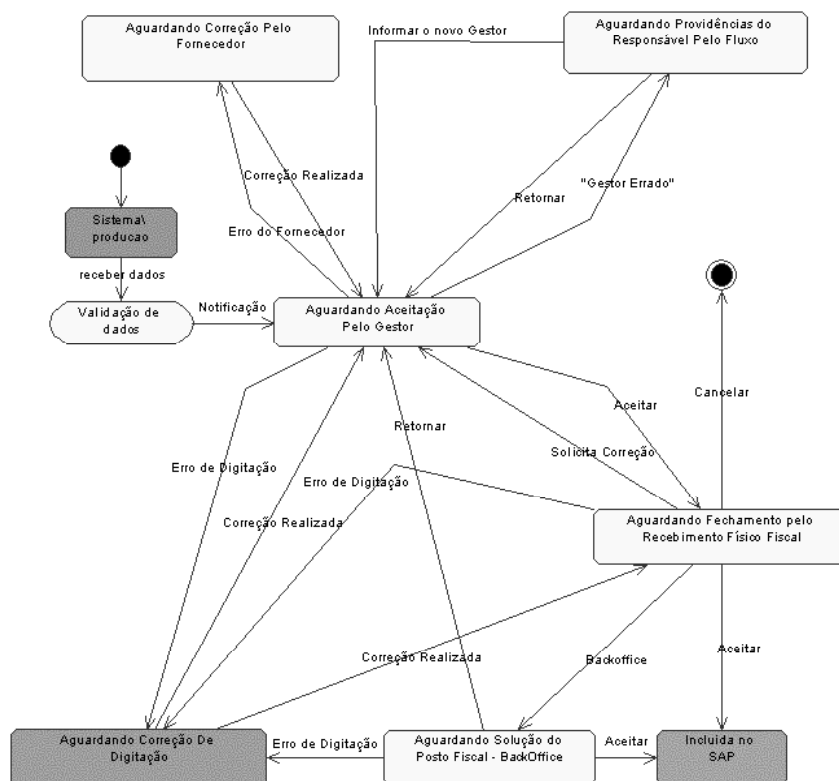


Figura 11 - Workflow: Nota fiscal de material

### 3.3 Metodologia

A fim de facilitar o entendimento do processo de desenvolvimento de software, vários modelos de ciclo de vida têm sido propostos. Estes modelos são descrições abstratas do processo de desenvolvimento, tipicamente mostrando as principais atividades e dados usados na produção e manutenção de software, bem como a ordem em que as atividades devem ser executadas. As atividades presentes nos diversos modelos de ciclo de vida de software não são um padrão; elas dependem da metodologia utilizada no desenvolvimento de um projeto de software.

Ao longo dos anos alguns modelos foram propostos [Vasconcelos2003]:

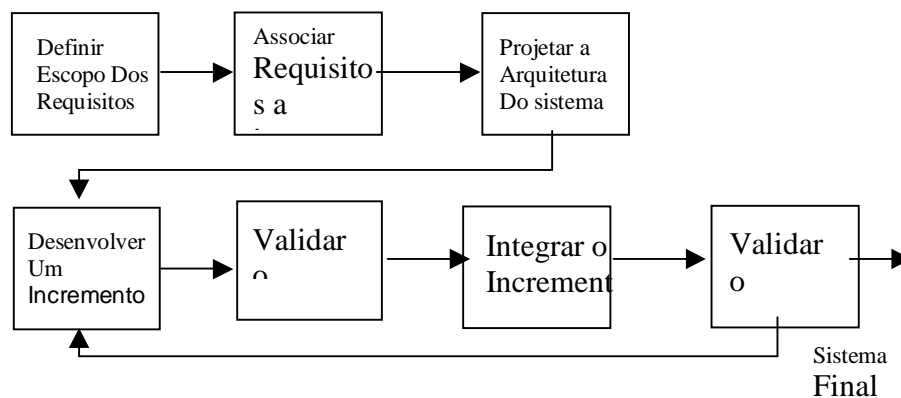
- Modelo em cascata [Royce1970];
- Modelo de desenvolvimento evolucionário [Boehm1981].
  1. Modelo de programação exploratória
  2. Modelo de prototipagem descartável
- Modelo de transformação formal,
- Modelo de desenvolvimento baseado em reuso
- Modelos iterativos [Boehm1988]:
  1. modelo de desenvolvimento espiral
  2. o modelo de desenvolvimento incremental.

O modelo de desenvolvimento mais usado hoje é o modelo interativo incremental, vamos mostrar este modelo um pouco mais detalhado.

O modelo de desenvolvimento incremental não objetiva entregar o sistema como um todo, o desenvolvimento e a entrega são divididos em incrementos, com cada incremento entregando parte da funcionalidade requerida *Figura 12*.

Os requisitos dos usuários são priorizados e os requisitos de mais alta prioridade são incluídos nas iterações iniciais. Uma vez que o desenvolvimento de um incremento é iniciado, os requisitos são "congelados", embora os requisitos possam continuar evoluindo para incrementos posteriores.

O escopo do sistema deve ser claramente entendido antes de se iniciar o desenvolvimento.



*Figura 12 - Modelo Incremental*

As principais vantagens do modelo incremental são [Vasconcelos2003]:

- A funcionalidade do sistema estará disponível mais cedo, pois ela é entregue a partir dos incrementos;
- Incrementos iniciais agem como um protótipo para ajudar a elicitar requisitos para incrementos finais;
- Diminuem-se os riscos de falhas no projeto como um todo; os serviços de prioridade mais alta do sistema tendem a receber mais testes.

O processo adotado como base para o desenvolvimento do projeto é um modelo incremental e fortemente baseado no *Rational Unified Process da Rational Corporation* [RUP2002], com a adição de algumas atividades e artefatos pertinentes às recomendações da ISO (*International Standard Organization.*), em particular as recomendações incluídas na norma [ISO15504] e ao modelo CMM (*Capability Maturity Model for Software (SW-CMM)V1.1* do SEI (*Software Engineering Institute*)).

O *Rational Unified Process* (RUP) é um processo de Engenharia de Software desenvolvido pela *Rational Software Corporation*, cujas principais características são um desenvolvimento iterativo e incremental, orientado a objetos, com foco na criação de uma arquitetura robusta, análise de riscos e utilização de casos de uso para o desenvolvimento.

Seguindo a estrutura do RUP, o projeto será desenvolvido ao longo de 4 fases – concepção, elaboração, construção e transição – cada uma com objetivos específicos:

- Na fase de concepção, deve-se estabelecer o escopo e a viabilidade econômica do projeto. A atividade principal desta fase é:



**Modelagem do Negócio** – envolve o entendimento da estrutura e dinâmica da organização do cliente, garantindo que clientes, usuários e desenvolvedores tenham a mesma visão da organização para a qual será feito o desenvolvimento.

- Na elaboração, o objetivo é eliminar os principais riscos e estabelecer uma arquitetura estável a partir da qual o sistema poderá evoluir. As principais atividades desta fase são:

**Elicitação de Requisitos** – envolve a definição dos requisitos do sistema e de como gerenciar escopo e mudanças de requisitos.

os.

**Análise e Projeto** – envolve a tradução dos requisitos numa especificação que descreve como implementar o sistema.

- Na fase de construção, um produto completo é desenvolvido de maneira iterativa até que esteja pronto para ser passado aos usuários. As atividades desta fase são:

**Implementação** – envolve o desenvolvimento de código: classes, objetos, etc., teste de unidades e integração de subsistemas.

**Teste** – envolve a verificação do sistema como um todo, com testes de integração e conformidade com os requisitos especificados.

- A fase de transição é quando o sistema é disponibilizado para o cliente. A principal atividade desta fase é:

**Implantação** envolve a instalação do sistema e treinamento de usuários, assim como o planejamento e condução de beta testes.

Uma das atividades importante no desenvolvimento de sistemas de software é a definição do ambiente de desenvolvimento, é neste momento que definimos qual será(ao) a(s) linguagem(s) de programação a ser(em) usada(s), quais as ferramentas serão utilizadas. De uma forma geral é a definição da tecnologia a ser usada no projeto.

Para o desenvolvimento do SANF/workflow, um dos critérios para a escolha das tecnologias em volvidas foi compatibilidade com a plataforma Microsoft. Vale lembrar que o sistema está sendo desenvolvido para “rodar” nesta plataforma. Baseado neste critério as seguintes tecnologias foram selecionadas:

- Linguagens de Programação a serem utilizadas: Visual Basic 6.0 (VB6), Active Information Pages (ASP) em conjunto com HTML, JavaScript e XML.
- A base de dados a ser usada, no primeiro momento, está sendo o SQLServer 7.0 ou Superior.

Outro ponto de decisão é a escolha da linguagem de modelagem a ser usada. Para decidir sobre a linguagem de modelagem foi feito um breve estudo das linguagens disponíveis. Sendo a *Unified Modeling Language* (UML) escolhida como a mais adequada. Será feita uma rápida apresentação da UML na próxima seção.

## *A UML*

Muitas fases do processo de desenvolvimento do sistema SANF/workflow são sustentadas na UML. Em função deste fato se faz necessário uma apresentação desta tecnologia.

Segundo o *Object Management Group*, [OMG2003], a *Unified Modeling Language* (UML) é uma linguagem unificada de modelagem que é usada para especificar, visualizar e documentar sistemas de software (e não só software), incluindo o projeto e a construção.

No processo de desenvolvimento de software o mais complicado é descrição do sistema, e o mais crítico é a comunicação entre cada um dos envolvidos no projeto e na criação do software. Nas últimas décadas, UML tem emergido como uma linguagem de modelagem visual para analistas, projetistas e programadores. Hoje já faz parte do desenvolvimento de software, a UML fornece a cada um dos envolvidos um vocabulário comum sobre o software.

A UML é uma tentativa de padronizar a modelagem orientada a objetos de uma forma que qualquer sistema, seja qual for o tipo, possa ser modelado corretamente, com consistência, fácil de se comunicar com outras aplicações, simples de ser atualizado e compreensível [OMG2003].

A UML está destinada a ser dominante, a linguagem de modelagem comum a ser usada nas indústrias. Ela está totalmente baseada em conceitos e padrões extensivamente testados provenientes das metodologias existentes anteriormente, e também é muito bem documentada com toda a especificação da semântica da linguagem representada em meta-modelos [OMG2003].

A Unified Modeling Language (UML) é dividida em cinco tipos de visões, doze tipos de diagramas e vários modelos de elementos que serão utilizados na criação dos diagramas e mecanismos gerais que em conjunto especificam e exemplificam as definições do sistema, tanto a definição no que diz respeito à funcionalidade estática e dinâmica do desenvolvimento de um sistema.

Uma breve descrição dos componentes UML:

- **Visões:** As Visões mostram diferentes aspectos do sistema que está sendo modelado. A visão não é um gráfico, mas uma abstração consistindo em uma série de diagramas. Definindo um número de visões, cada uma mostrará aspectos particulares do sistema, dando enfoque a ângulos e níveis de abstrações diferentes e uma figura completa do sistema poderá ser construída.
- **Modelos de Elementos:** Os conceitos usados nos diagramas são modelos de elementos que representam definições comuns da orientação a objetos como as classes, objetos, mensagem, relacionamentos entre classes incluindo associações, dependências e heranças.
- **Mecanismos Gerais:** Os mecanismos gerais provêm comentários suplementares, informações, ou semântica sobre os elementos que compõem os modelos; eles provêm também mecanismos de extensão para adaptar ou estender a UML para um método/processo, organização ou usuário específico.
- **Diagramas:** Os diagramas são os gráficos que descrevem o conteúdo em uma visão. UML possui nove tipos de diagramas

que são usados em combinação para prover todas as visões do sistema.

## **4. O DESENVOLVIMENTO DO SISTEMA SANF/WORKFLOW**

Neste capítulo, é realizada uma descrição do desenvolvimento do sistema SANF/workflow apresentando, brevemente, cada uma das 4 fases do seu desenvolvimento.

### **4.1 Conceção**

Nesta fase, se deu a definição do escopo e a viabilidade econômica do desenvolvimento do sistema SANF/workflow.

Foi feito um estudo juntamente com uma grande empresa do setor de telecomunicações do país, a qual terá a identidade preservada por questões legais. O principal objetivo desta atividade é garantir que toda a equipe envolvida no projeto tenham a mesma visão do que está sendo proposto.

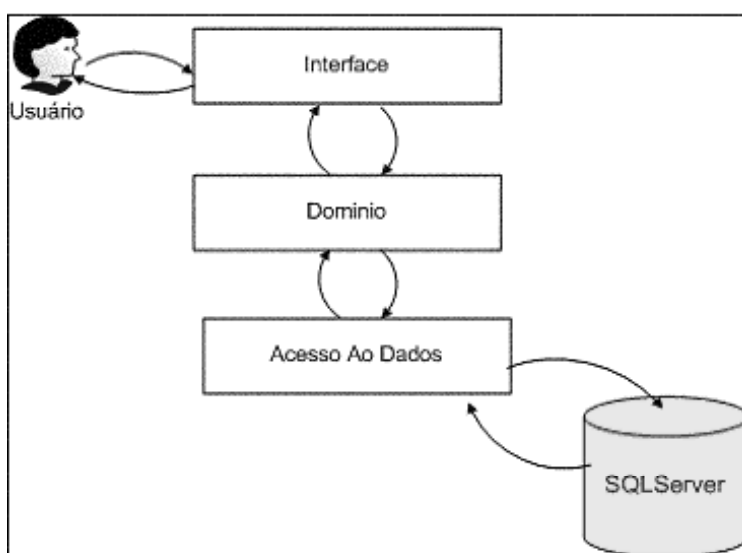
### **4.2 Elaboração**

É nesta etapa que se defini exatamente o que será o sistema, a atividade de especificação dos requisitos se passou em conjunto com a mesma empresa, em que se passou a fase de concepção. Para realizar a elicitação de requisitos, foram selecionadas algumas das várias técnicas de elicitação de requisitos, propostas pela Engenharia de Software. Resultando na confecção de um documento de requisitos que especifica, em linguagem natural, as funcionalidades do sistema.

Uma outra atividade, ainda nesta fase é a análise e projeto que envolve a tradução dos requisitos, obtidos na atividade de elicitação, numa especificação

formal que descreve como implementar o sistema. A UML é a linguagem de modelagem usada para especificar o como fazer.

Uma das primeiras especificações da fase de análise e projeto constitui-se da definição da arquitetura do sistema. O SANF/workflow será desenvolvido de forma orientada a objetos e em uma arquitetura de três camadas. Esta arquitetura inclui a camada do domínio (contém as regras do negócio), do acesso aos dados (persistência), dos serviços de alto nível e da interface do usuário. A *Figura 13* ilustra a arquitetura do sistema.



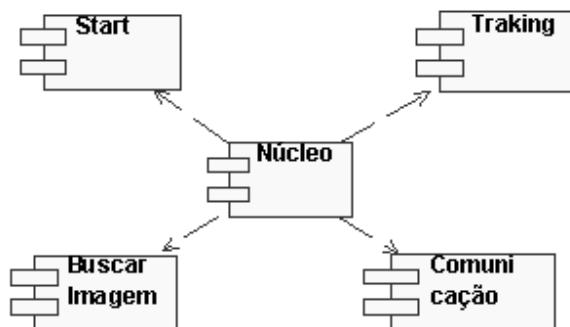
*Figura 13 - Arquitetura de sistemas em três camadas*

O sistema SANF/workflow é dividido em alguns subsistemas. Esses encontram-se na *Figura 14*.

- O subsistema **Start**: é responsável pela interface entre os sistemas SANF/produção e o SANF/workflow.

- O subsistema **Tracking**: é um subsistema que nos permite rastrear documento dentro do fluxo.
- O subsistema **Busca Imagem**: é o subsistema que captura a imagem no sistema GED e à apresenta para o usuário.
- O subsistema **Comunicação**: é responsável pelas notificações realizadas no sistema, via e-mail, para os usuários. E, por fim, o
- O **Núcleo do Sistema**: é no núcleo que estão definidas as regras de redirecionamento dos documentos fiscais no fluxo, o núcleo também é quem controla os demais subsistemas.

O subsistema **Autenticação**: que não está definido na *Figura 14*, entretanto existe, é responsável pela autenticação do usuário.



*Figura 14 - Diagrama de subsistemas*

Outro produto desta etapa é o projeto da base de dados. Na *Figura 15* apresentamos o modelo reduzido, contendo apenas as entidades e os relacionamentos, do banco de dados do SANF/workflow.



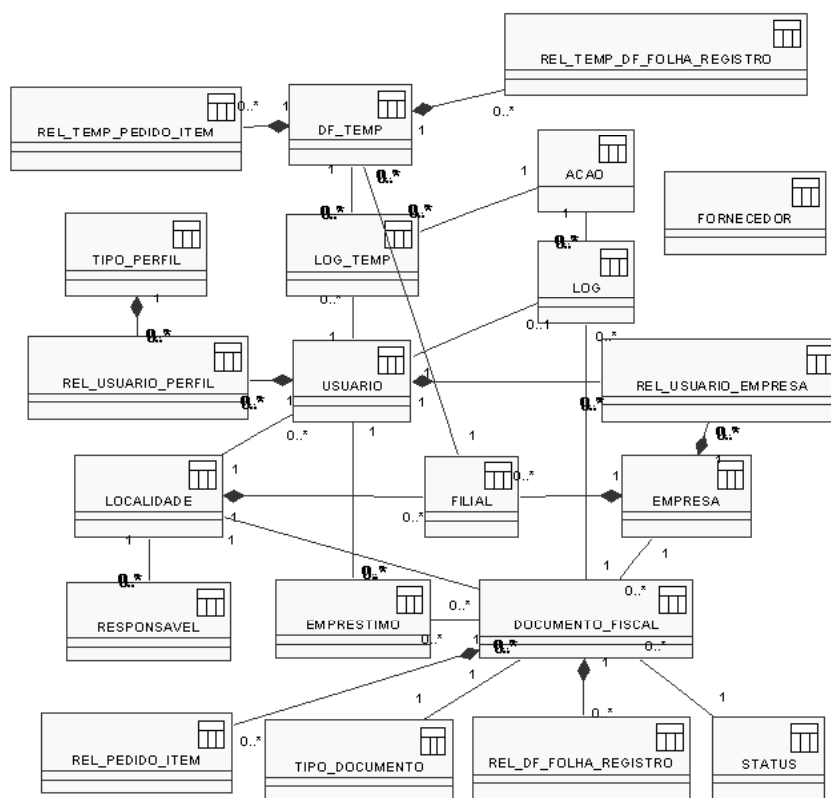


Figura 15 - Modelo relacional da base de dados do SANF/workflow

A UML foi utilizada para o analisar e projetar o sistema SANF/workflow, sendo gerados vários diagramas UML, tais como: diagrama de classe, diagrama de caso de uso e diagrama de colaboração e diagramas de componentes. Será disponibilizado, apenas alguns, diagramas, com o intuito de ilustrar e auxiliar no entendimento do sistema. Entretanto, por questões legais, não estará disponível toda a modelagem do sistema.

### **4.3 Construção**

O produto completo é gerado na fase de construção, é nesta fase onde o código fonte do sistema é gerado. Na medida em que os fontes vão sendo gerados, os teste começam a serem realizados. Como resultado desta etapa tem os fontes do sistema e ,conseqüentemente, o sistema propriamente dito.

Assim como, a modelagem, os fontes do sistema não estarão disponíveis, apenas por uma questão legal.

### **4.4 Implantação**

A Implantação envolve a instalação do sistema e treinamento de usuários, assim como o planejamento e condução de beta testes. Até o presente momento o sistema foi implantado em uma empresa, esta implantação ocorreu à dois meses atrás, durante este tempo o sistema tem sido monitorado e o que se tem percebido é uma comportamento satisfatório.

## 5. CONCLUSÕES

A aplicação das técnicas da Engenharia de Software tais como metodologias de desenvolvimento e o processo de desenvolvimento, foram de suma importância no desenvolvimento do sistema SANF/workflow, garantindo que o sistema seja completamente implementado e que atenda os requisitos propostos.

O desenvolvimento do sistema SANF/workflow foi totalmente concluído. E quando confrontamos os requisitos do sistema com suas funcionalidade, podemos observar que o sistema cobre, de forma satisfatória, uma grande quantidade dos requisitos propostos.

O SANF/workflow foi implantado em uma grande empresa do setor de telecomunicações, há mais de dois meses. O sistema tem atendido as principais necessidades da empresa. Entretanto já foram levantadas possíveis melhorias e que já estão sendo projetadas para a nova versão do sistema. As novas vêm automatizar ainda mais o fluxo de documentos fiscais.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

[Alonso95] Alonso, G. and Agrawal, D. and Abbadi, A. and Mohan, C. *Functionality and Limitations of Current Workflow Management Systems*. IBM Almaden Research Center 650 Harry Road (K55-B1), San Jose, CA 95120-6099, USA, 1995.

[Alonso97] Alonso, G. Agrawal, D. and El Abbadi, A . and Mohan, C. *Functionalities an Limations of Current Workflow Management Systems*. Technical report, IBM Almaden Research Center, 1997. To apper in IEEE Expert.

[Boehm1988] Boehm, B. *A Spiral Model for Software Development and Enhancement*, Computer, vol. 21, n. 5, maio 1988.

[Casati96] Casati, F. and Grefen, P. and Pozzi, G. and Sánchez, G. *WIDE Workflow model and architecture*. Politecnico di Milano, Italy. University of Twente, The Netherlands, Sema Group sae, Spain. Project of the European Commission (Project N. 20280).

[Cruz1998] Cruz T.: *Workflow. A Tecnologia que vai Revolucionar* . São Paulo, Atlas. 1998.

[ISO15504] ISO/IEC TR 15504, Parts 1-9: Information Technology – Software Process Assessment, 1998.

[Kamath95] Kamath, M. & Rmamritham, K. *Modeling, Correctness & System Issues in Supportin Advanced Database Applications using Workflow Management Systems*. University of Massachusetts, 1995. (Tecnical Report 95-50), 1995.

[kobielus1997] Kobielus, James G.: *Workflow Strategies; IDG Books – Foster City – CA*, 1997.

[McCready92] McCready, S. *There is more than one kind of workflow software*. Computerword, november, 1992.

[OMG2003] *Object Management Group*. [www.omg.org](http://www.omg.org) site acessado em junho de 2003.

[Royce1970] Royce, W. *Managing the Development of Large Software Systems*, 1970.

[RUP2002] *Rational Unified Process*, 2002 A.

[SEI2003] *Carnegie Mellon Software Engineering Institute*, [www.sei.cmu.edu](http://www.sei.cmu.edu) site acessado em junho de 2003.

[Tramontina2003] Tramontina, Gregório B.: *O Estado da Arte da Tecnologia de Workflow*, Universidade Estadual de Londrina, 2002.

[Vasconcelos2003] Vasconcelos, M. L & Maciel, T. M. M. *Introdução à Engenharia de Software e aos princípios de qualidade*, 2003.

[WIDE2003] The WIDE Workflow Modeling Language. Available at: <http://dis.sema.es/projects/WIDE/Documents/4080-2.pdf>. site acessado em novembro de 2003.

[WFMC2003] *Workflow Management Coalition. The Workflow Reference Model*. <http://www.wfmc.org>. site acessado em novembro 20030