

**FACULDADES DE TAQUARA**  
**FACULDADE DE INFORMÁTICA DE TAQUARA**  
**CURSO DE SISTEMAS DE INFORMAÇÃO**

**SISTEMA DE PRÉ-VISUALIZAÇÃO E**  
**CONTROLE DE ARQUIVOS HP-GL/2**  
***PLOTCONTROL***

**ADRIANO RENATO KLEIN**

**Taquara**

**2007**

**ADRIANO RENATO KLEIN**

**SISTEMA DE PRÉ-VISUALIZAÇÃO E  
CONTROLE DE ARQUIVOS HP-GL/2  
*PLOTCONTROL***

Trabalho de Conclusão apresentado ao  
Curso de Sistemas de Informação da  
Faculdade de Informática de Taquara das  
Faculdades de Taquara, como requisito  
parcial para obtenção do grau de Bacharel  
em Sistemas de Informação, sob  
orientação do Professor Eurico Antunes.

**Taquara**

**2007**

*Dedico este trabalho à minha família, à minha  
namorada e aos amigos que me apoiaram.*

## **AGRADECIMENTOS**

*Agradeço à Deus em primeiro lugar, que me proporcionou saúde e capacidade para superar todos os desafios até este momento de minha vida.*

*A todo apoio e compreensão da minha família e pessoas do meu convívio, que me deram apoio e auxílio ao longo desta jornada.*

*Aos colegas e amigos Diego, Roger, Gregol, Tuani, Delmar, Andrea e Everton, com quem tive a alegria de conviver durante este curso. Em especial ao Vitor, Carlos e Pretto, que além de excelentes parceiros, também se tornaram grandes amigos. Aos demais colegas, que também fizeram em algum momento, parte desta caminhada.*

*Aos meus professores, que durante todo o curso transmitiram o seus conhecimentos e tiveram um papel fundamental em minha vida acadêmica, e às Faculdades de Taquara pela oportunidade oferecida, juntamente com seus funcionários e direção.*

*Um agradecimento em especial ao meu orientador, professor Eurico Antunes, que sempre me apoiou e incentivou para que este trabalho se tornasse uma realidade. Ao professor Fernando Carlos Jung, por suas constantes cobranças e me indicar um rumo neste final de curso. Ao colega Luciano Hoffmaister Ribeiro, pelo grande suporte e apoio na conclusão deste trabalho.*

*Aos meus colegas de trabalho Gustavo e Vagner, que me apoiaram e me substituíram com muita competência durante minha ausência na empresa.*

*Enfim, a todos que tornaram os momentos da minha vida universitária especiais e inesquecíveis, mas uma vez meus sinceros agradecimentos.*

## SUMÁRIO

1	<b>INTRODUÇÃO</b>	12
1.1	<b>Contextualização e motivações</b>	12
1.2	<b>Objetivos e justificativas</b>	13
1.3	<b>Organização do trabalho</b>	13
2	<b>REFERENCIAL TEÓRICO</b>	15
2.1	<b>Sistemas de <i>cad</i></b>	15
2.1.1	<i>O Autocad</i>	17
2.2	<b><i>Plotter</i></b>	18
2.3	<b>Linguagem de controle</b>	23
2.3.1	A linguagem HP-GL	24
2.3.1.1	Exemplo de instruções em HP-GL	25
2.3.2	A linguagem HP-GL/2	26
2.3.2.1	Exemplo de instruções em HP-GL/2	27
2.3.2.2	Exemplo do conteúdo de um arquivo em HP-GL/2	28
3	<b>METODOLOGIA DA ANÁLISE</b>	30
3.1	<b>O contexto atual</b>	30
3.1.1	As vantagens da utilização de arquivos HP-GL/2	33
3.1.2	Os problemas da utilização de arquivos HP-GL/2	34
3.2	<b>Análise do processo de trabalho</b>	36
3.3	<b>Análise dos sistemas de informação utilizados</b>	37
3.4	<b>Análise do sistemas a ser desenvolvido</b>	38
3.4.1	Definição dos requisitos	38
3.4.2	Diagrama de fluxo de dados (DFD)	39
3.4.3	Dicionário de dados	45
3.4.4	Especificação de processos	47
3.4.5	Diagrama de entidade relacionamento (DER)	50
4	<b>MODELAGEM E DESENVOLVIMENTO DO SISTEMAL</b>	51
4.1	<b>Definição da tecnologia de desenvolvimento</b>	51
4.2	<b>Prototipação da interface de comunicação com o usuário</b>	52
4.3	<b>Modelagem do banco de dados</b>	57

4.4	<b>Codificação</b>	57
4.5	<b>Impressão de Ordem de Serviço</b>	58
4.6	<b>Teste e Validação</b>	59
4.7	<b>Instalação</b>	59
4.8	<b>Manutenção</b>	60
5	<b>CONCLUSÕES</b>	61
	<b>REFERÊNCIAS</b>	62
	<b>ANEXO</b>	64
	<b>ANEXO A - Análise do processo de trabalho</b>	65

## LISTA DE ILUSTRAÇÕES

Figura 1: Ivan Sutherland operando o computador TX-2 com o Sketchpad .....	16
Figura 2: detalhe da tela do sistema Sketchpad .....	16
Figura 3: histórico linha do tempo do <i>AutoCAD</i> .....	18
Figura 4: <i>plotter</i> Calcomp 565 de 1961 .....	20
Figura 5: <i>plotter</i> HP 9872T de 1981 .....	21
Figura 6: <i>plotter</i> HP 7580b de 1982 .....	21
Figura 7: <i>plotter</i> HP Designjet 430 de 1991 .....	25
Figura 8: exemplo de traçado numa <i>plotter</i> em HP-GL .....	25
Figura 9: exemplo de traçado numa <i>plotter</i> em HP-GL/2 .....	28
Figura 10: diagrama do fluxo de trabalho .....	36
Figura 11: diagrama de contexto ou de nível 0 .....	40
Figura 12: diagrama de fluxo de dados de nível 1 .....	41
Figura 13: diagrama de fluxo de dados de nível 2 do processo 1 .....	42
Figura 14: diagrama de fluxo de dados de nível 3 do processo 1.1 .....	43
Figura 15: diagrama de fluxo de dados de nível 2 do processo 2 .....	44
Figura 16: diagrama de fluxo de dados de nível 2 do processo 3 .....	44
Figura 17: diagrama de entidade relacionamento .....	50
Figura 18: tela de entrada no sistema .....	53
Figura 19: tela de principal do sistema .....	54
Figura 20: tela de cadastro dos usuários do sistema .....	54
Figura 21: tela de cadastro de serviços do sistema .....	55
Figura 22: tela de cadastro de clientes do sistema .....	55
Figura 23: tela de visualização de arquivos do sistema .....	56
Figura 24: tela de ordens de serviço do sistema .....	56
Figura 25: ordem de serviço impressa .....	58

## LISTA DE TABELAS E QUADROS

Quadro 1: exemplo de codificação em linguagem HP-GL .....	26
Quadro 2: exemplo de codificação em linguagem HP-GL/2 .....	28
Quadro 3: exemplo do conteúdo de um arquivo em linguagem HP-GL/2 .....	29
Quadro 4: recursos disponíveis de <i>hardware</i> .....	37
Quadro 5: recursos disponíveis de <i>software</i> .....	37
Quadro 6: dicionário de dados dos processos iniciais .....	45
Quadro 7: dicionário de dados do fluxo Usuário .....	46
Quadro 8: dicionário de dados do fluxo Cliente .....	46
Quadro 9: dicionário de dados do fluxo Serviço .....	47
Quadro 10: exemplo de teste no módulo Validar Dados .....	59



## LISTA DE SIGLAS

*CAD: Computer Aided Design* ou Desenho Auxiliado por Computador

*E-mail: eletronic mail* ou correio eletrônico

DWG: tipo de arquivo do *AutoCAD*, fabricado pela empresa americana Autodesk

HP-GL/2: *Hewlett Packard Graphics Language* versão 2

FTP: *File Transfer Protocol* ou Protocolo de Transferência de Arquivos

Dpi: *dot per inch* ou ponto por polegada, usada para medir resolução gráfica

ROM: *Read Only Memory* ou Memória de Somente Leitura

SDI: *Simple Document Interface* ou Interface Simples de Documentos

CD-R: *Compact Disc Recordable* ou Disco Compacto Gravável

*Case: Computer-Aided Software Engineering*

## RESUMO

O objetivo deste trabalho foi utilizar a dinâmica e os conceitos do método de trabalho de uma empresa prestadora de serviços de impressão em grandes formatos (plotagem), para o desenvolvimento de um programa de computador que permita visualizar, controlar, imprimir e contabilizar arquivos em formato *HP-GL/2*, auxiliando assim nas tarefas rotineiras de orçamento e impressão de projetos, e também no controle de clientes, arquivos e serviços com o finalidade de torná-las mais simplificadas, ágeis e organizadas. Para isto foi realizado um levantamento dos dados junto a uma empresa do ramo de impressão em grandes formatos, afim de obter informações necessárias sobre o processo atual de trabalho, verificar detalhadamente os sistemas de informação atualmente utilizados na empresa e desenvolver um protótipo do *software*, que permitiu ter uma visão antecipada das tarefas que o sistema implementa.

Palavras-Chave: *hp-gl/2*, *plotter*, plotagem, visualizador.

## **ABSTRACT**

*The objective of this work was to use the dynamics and the concepts of the functioning in a small business company, that provides print services in wide format, for a software development which allows to view, to control, to plot and to audit HP-GL/2 files, aiding so in routinist tasks like budget e projects print, and also in the management of customers, files and services with the purpose to become it more simplified, agile and organized. For this a survey of the data inside a company of print services in wide format was carried through, to get necessary information about the current work process, verify the systems of information currently used in this company and to develop a software prototype, that allowed to have an anticipated vision of the tasks that the system implements.*

*Key-Words: hp-gl/2, plotter, plot, viewer.*

## 1 INTRODUÇÃO

Com uma situação de mercado cada vez mais competitiva em todas as áreas, um dos diferenciais de mercado para as empresas que atuam em um mesmo segmento passa a ser os seus recursos tecnológicos e como estes são utilizados.

Neste capítulo apresenta-se uma rápida introdução do trabalho proposto, juntamente com as motivações, o objetivos e justificativas para a sua realização.

### 1.1 Contextualização e motivações

Pensando em aprimorar o uso do computador com eficiência e eficácia dentro de uma empresa especializada em prestação de serviços na área de impressão de projetos, tendo como base para o seu trabalho a cópia de arquivos de sistemas de *CAD*, detectou-se uma necessidade para o desenvolvimento de um *software* para auxiliar nas tarefas rotineiras, melhorando assim o processo de trabalho de empresas que prestam serviços na área de plotagem<sup>1</sup>, e oferecendo uma alternativa específica para este ramo de negócio.

A agilidade e contabilização do processo de impressão influencia diretamente no fluxo do trabalho, sendo um fator determinante para obtenção de vantagem competitiva neste tipo de empresa.

Atualmente existem vários programas que auxiliam na tarefa de impressão de projetos, mas nenhum fornecendo uma solução completa que represente a realidade do negócio especificamente. Por este motivo, criou-se uma grande expectativa e motivação para o desenvolvimento deste software.

---

<sup>1</sup> Plotagem: impressão de grandes formatos

## **1.2 Objetivos e justificativas**

Este trabalho tem como o seu foco principal o desenvolvimento de um software que permitirá visualizar, controlar, imprimir e contabilizar arquivos em formato HP-GL/2, auxiliando assim nas tarefas rotineiras de orçamento e impressão de projetos, e também no controle de clientes, arquivos e serviços com o finalidade de torná-las mais simplificadas, ágeis e organizadas.

Com a utilização deste software por empresas de plotagem, espera-se que as mesmas obtenham um aumento significativo do desenvolvimento de suas tarefas, uma melhoria no gerenciamento de seus recursos financeiros e, conseqüentemente, uma fidelização na sua carteira de clientes.

Este trabalho é uma continuação do projeto de pesquisa que iniciou-se no segundo semestre de 2006, na Faculdade de Informática de Taquara.

## **1.3. Organização do Trabalho**

O conteúdo do trabalho está distribuído em cinco capítulos, como segue:

- a) o Capítulo 1 apresentou uma breve introdução, com motivos que induziram a realização deste trabalho e os seus objetivos;
- b) o Capítulo 2 expõe as definições e conceitos, abordando o que o sistema irá controlar dentro da empresa;
- c) a metodologia utilizada para a concepção do desenvolvimento deste trabalho é o objetivo do Capítulo 3;
- d) no Capítulo 4 serão apresentados os resultados, como a modelagem e o

desenvolvimento do software com suas telas, e as demais etapas da metodologia empregada;

e) finalizando, o Capítulo 5 abordará as conclusões deste trabalho.

## 2 REFERENCIAL TEÓRICO

### 2.1 Sistemas de Cad

*Computer Aided Design (CAD)*, ou desenho auxiliado por computador, é o nome genérico de sistemas computacionais (*software*) utilizados pela engenharia, geologia, arquitetura e design para facilitar o projeto e desenho técnicos. No caso do *design*, este pode estar ligado especificamente a todas as suas vertentes (produtos como vestuário, eletroeletrônicos, automobilísticos, etc.), de modo que os jargões de cada especialidade são incorporados na interface de cada programa (WIKIPEDIA, 2006, p.1).

Na Década de 50 diversas pesquisas industriais e acadêmicas em Desenho Auxiliado por Computador criaram ferramentas bidimensionais usadas para gerar caminhos de ferramentas em máquinas de CNC, ou seja, Controle Numérico Computadorizado.

Na Década de 60 desenvolveu-se a computação gráfica interativa, onde grandes companhias, como ITEK e General Motors, desenvolveram seus próprios *softwares* de *CAD* baseado em grandes *Mainframes* (computador central), empregados inicialmente na indústria automobilística, aeroespacial e em agências governamentais (DALTON STATE COLLEGE, 2000).

Em 1962, um sistema chamado *Sketchpad* foi desenvolvido no Lincoln Laboratory (MIT) por Ivan E. Sutherland, baseado em sua tese de doutorado intitulada “*A Machines Graphics Communications System*”. Sutherland propôs a idéia de utilizar um teclado e um lápis óptico para selecionar, situar e desenhar, juntamente com uma imagem representada na tela (DALTON STATE COLLEGE, 2000; BCOLLEGE OF

COMPUTING AT GEORGIA TECH, 2006).



**Figura 1: Ivan Sutherland operando o computador TX-2 com o Sketchpad**  
Fonte: College of Computing at Georgia Tech (1962).



**Figura 2: Detalhe da tela do sistema Sketchpad**  
Fonte: College of Computing at Georgia Tech (1962).

Em 1965, a empresa ITEK Control Data Corporation comercializa o primeiro *CAD* pelo preço de US\$ 500.000,00.

Na década de 70 a marinha americana inicia o desenvolvimento de programas de *CAD* em três dimensões (3D), baseado nas primitivas geométricas caixas, cones, cilindros, etc., e grandes companhias do setor automobilístico e aeroespacial como



Ford, Chrysler e Lookheed adotam o uso de sistemas CAD. No final dos anos 70 um sistema *CAD* custava US\$ 125.000,00.

Os softwares *CAD* costumam ser utilizados por um nicho pequeno de usuários, devido a sua intensa especialização e seu alto custo, mas foi no início dos anos 80, mais precisamente em 1982, que estes sistemas se popularizaram, com o lançamento do *AutoCAD* (USP, 2005; CARLSON, 2006).

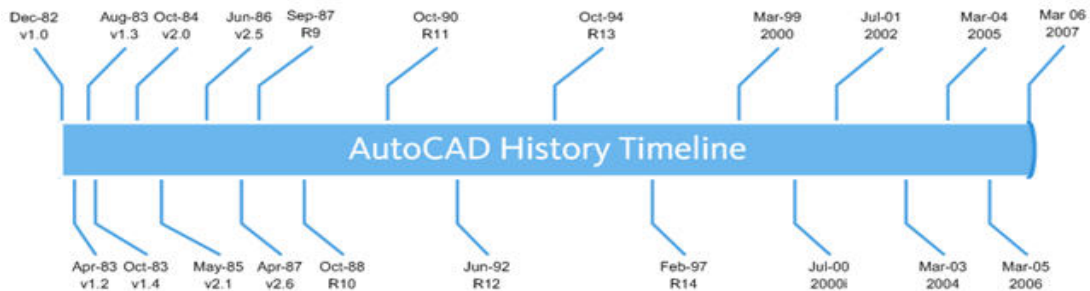
### 2.1.1 O Autocad

Em 1982, John Walker juntamente com mais 12 sócios fundam a AUTODESK com a idéia de produzir um programa *CAD* de baixo custo para PC. É criado então o *AutoCAD*. A primeira versão saiu com o nome *MicroCAD* e foi apresentada na COMDEX Trade Show de Las Vegas em novembro de 1982. Este primeiro *AutoCAD* foi executado sob o sistema operacional CP/M em computadores com processador Intel 8080 e custava aproxima-damente US\$ 10.000,00. Com a primeira versão do *AutoCAD* começou uma revolução em projetos e design, ele permitia muito mais que representar graficamente coordenadas e pontos, já oferecia *Layers*, texto e um menu de comandos, mas tudo muito limitado.

O *AutoCAD* se consolidou no mercado como uma alternativa viável levando-se em consideração os altos valores que eram praticados aquele momento, tanto em termos de *hardware* quanto de *software*, pois a partir de sua criação qualquer computador pessoal poderia executá-lo. Com a popularização do PC, o *AutoCAD* tornou-se um referencial para o ramo de *CAD* (AUTODESK, 2006).

Diversas foram as versões lançadas deste produto com o passar dos tempos, agregando novas funcionalidades e evoluindo juntamente com o *hardware* e o

*software* de cada época, como podemos observar na figura a seguir:



**Figura 3: Histórico linha do tempo do *AutoCAD***

Fonte: Unofficial AutoCAD History Pages (2007).

O *AutoCAD* é o padrão de fato em soluções genéricas de CAD, e os seus formatos de arquivos DXF e DWG são os mais comuns para o intercâmbio entre os sistemas CAD. Muito difundido nos dias de hoje, a maioria dos projetistas ou desenhistas de alguma forma utiliza os recursos ofertados por um sistema de CAD.

## 2.2 *Plotter*

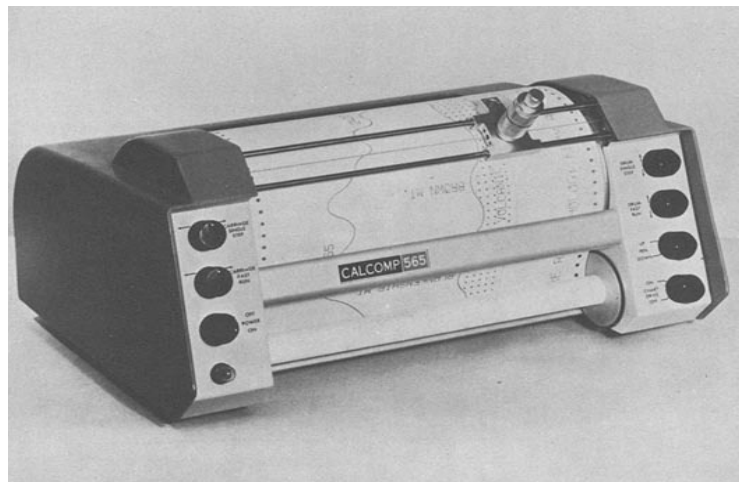
Para que seja possível imprimir um projeto ou desenho que possua tamanho maior que o convencional e padronizado pelas impressoras domésticas, que oferecem suporte ao formato A4 por exemplo, devemos possuir uma impressora que trabalhe com tamanhos de papéis maiores que este citado. Estas impressoras são chamadas de *plotters* de impressão ou impressoras de grandes formato.

A primeira *plotter* comercial foi desenvolvida aproximadamente em 1961 por uma empresa chamada Calcomp, que comercializou-a com o nome de Calcomp 565. Esta mesma *plotter* também foi vendida como IBM 1627, pela empresa IBM.

juntamente com o computador IBM 1620, sendo este então o primeiro periférico de outro fabricante vendido com seus computadores (WIKIPEDIA, 2006).

Este dispositivo imprimia gráficos vetoriais movendo uma caneta sobre a superfície do movimentador do papel, sendo restrita ao desenho de linhas apenas. Neste início de desenvolvimento, esta era uma das diferenças de *plotters* e impressoras, pois estas últimas só conseguiam imprimir caracteres. Outra diferença é que as *plotters* podiam imprimir textos, porém o faziam muito devagar devido a seu mecanismo de movimentação das canetas. Nesta época, a impressão de um texto acontecia de uma forma mais simples do que imprimir um projeto numa *plotter*, pois a impressora foi concebida para textos, onde o usuário simplesmente enviava o comando com o texto e ela imprimia. No caso da *plotter*, onde apenas linhas são impressas, teve-se a necessidade de serem criadas linguagens de controle para este dispositivos, informando onde determinada linha deveria começar e onde deveria terminar, culminando assim numa tarefa complexa para o hardware e software disponíveis na época (BREVARD USER'S GROUP, 2006).

A maioria dos *plotters* desenvolvidos nas décadas de 60 e 70 funcionavam com canetas, podendo ser esferográficas ou do tipo “caneta de nanquim”, que se deslocavam através de um eixo no sentido horizontal, chamado de eixo Y. O papel por sua vez se deslocava no sentido vertical, chamado de eixo X. Melhoramentos foram feitos com o passar dos anos, como o suporte ao papel em rolo e a múltiplas canetas, podendo assim serem plotados desenhos multicoloridos (WIKIPEDIA, 2006).



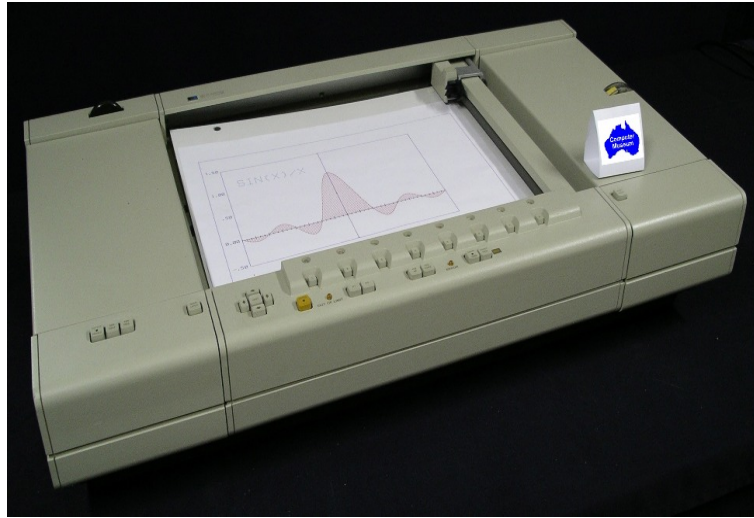
**Figura 4: Plotter Calcomp 565 de 1961**

Fonte: Unofficial AutoCAD History Pages

Em 1977 é lançada no mercado a *plotter* HP modelo 9872A, que foi a primeira *plotter flatbed*<sup>2</sup> multi-penas, isto é, operava com até quatro canetas ao mesmo tempo. Foi nesta época também que a HP implementou o suporte a linguagem HP-GL (Hewlett Packard Graphics Language). Diversas outras melhorias ocorreram nos anos seguintes nesta *plotter*, gerando assim outros modelos, como a 9872S que possuía um sistema de alimentação de papel automático, a 9872C que possuía suporte para mais quatro canetas, totalizando assim oito canetas e a 9872T que juntava todos os benefícios dos modelos 9872S e 9872C, com suporte para um rolo de papel com 30 centímetro de largura e 60 metros de comprimento.

---

<sup>2</sup> *plotter flatbed*: tipo de *plotter* plana, onde o papel permanece estático e as canetas se movem na sua superfície.



**Figura 5: Plotter HP 9872T de 1981**

Fonte: HP Computer Museum

Em 1981, a HP inova e lança o modelo 7580B que marca sua entrada no mercado de *plotters* com capacidade para grandes formatos. Neste modelo o papel é tracionado para dentro da *plotter*, como nas impressoras que conhecemos nos dias de hoje, mas o restante do mecanismo funciona de maneira similar, permanecendo o traçado dos gráficos através de canetas (HP COMPUTER MUSEUM, 2006).



**Figura 6: Plotter HP 7580b de 1982**

Fonte: HP Computer Museum

Nesta época outras tecnologias de impressão foram desenvolvidas e utilizadas, como a *plotter* eletrostática que faz plotagens utilizando um processo de sensibilização eletrostática em um papel especial que é exposto ao toner<sup>3</sup> e que adere aos pontos. As *plotters* eletrostáticas eram mais rápidas que as plotadoras de penas, mas seu custo também era maior e exigiam mais processamento do computador a que ela estava conectada. Também foram desenvolvidos equipamentos que utilizavam tecnologias similares a dos aparelhos de fax, onde a impressão ocorre através de transferência térmica para o papel, porém os custos eram altos e a impressão sumia do papel com o passar do tempo (GOVERT, 2006).

Somente em 1991 que a HP e outros fabricantes, como Encad, Océ e Calcomp, iniciam a comercialização de *plotters* de grandes formatos com tecnologia jato de tinta (*inkjet*). A HP entra neste mercado através da *plotter* monocromática Designjet 600 que contava uma tecnologia bastante avançada para sua época, pois possuía alimentador de papel que suportava um rolo de 90 centímetros de largura por até 50 metros de comprimento, cortador automático do papel, fila de impressão dos trabalhos já executados, diversos modos de impressão podendo alcançar até 600 dpi. Outro fator muito importante para o sucesso deste produto foi o fato de que ele interpretava a nova versão de linguagem de controle de dispositivos da HP chamada de HP-GL/2, uma evolução da antiga HP-GL que era utilizadas nas primeiras *plotters*.

Como já havia acontecido com os modelos de tecnologia de canetas, as *plotters* baseadas em jato de tinta também evoluíram e diversos outros modelos da HP surgiram, e em todos eles estava presente a linguagem HP-GL/2. Esta linguagem tornou-se um padrão para aplicações de gráficos vetoriais e até outros fabricantes que

---

<sup>3</sup> *Toner*: pigmento em pó muito fino obtido através da junção de carbono e polímeros

concorriam nesta categoria de produto licenciaram e passaram a usá-la em seus equipamentos (HP COMPUTER MUSEUM, 2006).



**Figura 7: Plotter HP Designjet 430 de 1991**

Fonte: HP Computer Museum

## **2.3 Linguagem de controle**

Para o funcionamento de um dispositivo de impressão, todo o fabricante deve desenvolver ou licenciar uma linguagem que controle o dispositivo. Diversas linguagens de diversos fabricantes e desenvolvedores foram criadas ao longo dos anos, mas a que mais se popularizou foi a linguagem HP-GL e posteriormente HP-GL/2, da empresa HP. (KAY & LEVINE, 1995).

### 2.3.1 A linguagem HP-GL

A linguagem de programação HP-GL foi criada no início de 1970 por uma equipe de desenvolvimento chamada Divisão San Diego, na HP. Mas foi somente em 1977 que Norm Johnson e Dale Shapper a adaptaram para as primeiras *plotters* da HP série 9872A, do tipo X/Y que possuíam quatro canetas e formato A3 e tornou-se assim a primeira linguagem de alto nível disponível para estes dispositivos do mercado, definindo um novo padrão para os demais fabricantes. Este conjunto de comandos e instruções HP-GL é implementado em uma ROM que é acoplada a placa controladora principal da *plotter* e tem a finalidade de reduzir o trabalho requerido por programadores de software que desenvolvem aplicações e *drivers*<sup>4</sup> para *plotters* (BANGERT, 1996).

A linguagem HP-GL usa 2 caracteres mnemônicos<sup>5</sup> como instrução para efetuar os desenhos de linhas, arcos, círculos e textos. Este caracteres são números inteiros (dígitos de 0 à 9, positivos ou negativos) para determinar os parâmetros de onde os dados serão desenhados e caracteres alfabéticos para escolher os comandos que efetuarão os desenhos. São usados ainda sinais de pontuação comum, como ponto, vírgula, ponto-vírgula e espaço. Foram escolhidos um par de caracteres para os comandos porque apenas um caractere não permitia comandos suficientes e três caracteres geravam muitas combinações, podendo assim confundir os programadores e desenvolvedores (GOVERT, 2006).

Também na sua definição inicial para *plotters*, a linguagem foi adaptada para

---

<sup>4</sup> drivers: programas que possibilitam a comunicação entre o sistema operacional e dispositivos periféricos ligados a um computador

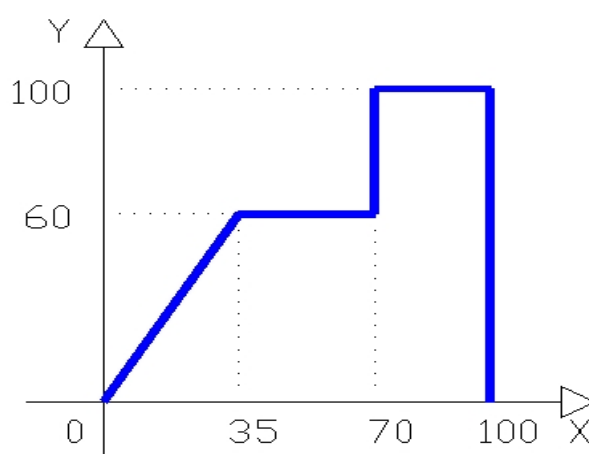
<sup>5</sup> mnemônicos: símbolos ou abreviaturas usados para auxiliar a memorização



*firmware*<sup>6</sup> com endereçamento de 16 bits<sup>7</sup> e resolução gráfica de 40 un./mm. (unidades por milímetros). Isto significa que o limite físico do tamanho da impressão é limitado em aproximadamente um metro quadrado, o que para a época em que a tecnologia foi desenvolvida supria amplamente as necessidades comerciais (HP COMPUTER MUSEUM, 2006).

### 2.3.1.1 Exemplo de instruções em HP-GL

Este exemplo, conforme a Figura 8, demonstra como seria utilizada a linguagem HP-GL para efetuar um simples desenho no papel, através de uma *plotter* devidamente configurada. Após o envio do arquivo para o dispositivo de impressão, a *plotter* internamente, através de seu *firmware*, executa uma análise nos dados de entrada e os traduz para comandos HP-GL, que movimentam o motor da *plotter* nas direções X e Y, enquanto o carro de impressão com as canetas move-se para cima e para baixo, traçando assim o desenho, que foi traduzido em comandos, para o papel.



**Figura 8: exemplo de traçado numa plotter em HP-GL**

Fonte: HP Computer Museum

<sup>6</sup> *Firmware*: programa que reside no dispositivo e é responsável pelo seu funcionamento

<sup>7</sup> *Bits*: menor unidade de medida digital, que agrupados em 7 ou 8 formam bytes

Instrução	Significado
IN;	Inicializa a <i>plotter</i> , zerando todos parâmetros HP-GL para o padrão
SC0,100,0,100;	Define a escala para os valores de X ( $X_{min}=0$ and $X_{max}=100$ ). Os mesmos valores serão usados para o Y
SP1;	Seleciona a caneta n° 1 (usado nas <i>plotters</i> antigas para escolher a primeira caneta do carro de impressão)
PU;	Levanta a caneta, nenhuma linha é desenhada no papel
PA0,0;	Move a caneta para posição absoluta 0,0 (x,y)
PD;	Baixa a caneta no papel para desenhar
PA35,60,70,60, 70,100,100,100, 100,0;	Move a caneta através das diferentes coordenadas x,y (35,60), (70,60), (70,100), (100,100), (100,0)
PU;	Levanta a caneta novamente
SP0;	Seleciona a caneta n° 0 (usado nas <i>plotters</i> antigas para não selecionar nenhuma caneta e mover o carro para sua posição original)
NR; ou PG;	Termina a seqüência de comandos, informando que a impressão terminou e que o papel já pode ser removido

**Quadro 1: exemplo de codificação em linguagem HP-GL**

Fonte: HP Computer Museum

### 2.3.2 A linguagem HP-GL/2

Com o passar do anos e a evolução tecnológica crescente, a linguagem HP-GL começou a dar os primeiros sinais de sua limitação, pois os arquivos gerados para impressão eram cada vez maiores. Nas linguagens de controle baseadas em *raster* o dispositivo começa a impressão assim que recebe os primeiros comandos, pois o computador opera no modo escravo, enviando os dados continuamente para o dispositivo até ser finalizada a impressão. Se ao longo da impressão a comunicação lógica for interrompida, o dispositivo irá parar e o trabalho terá que ser enviado

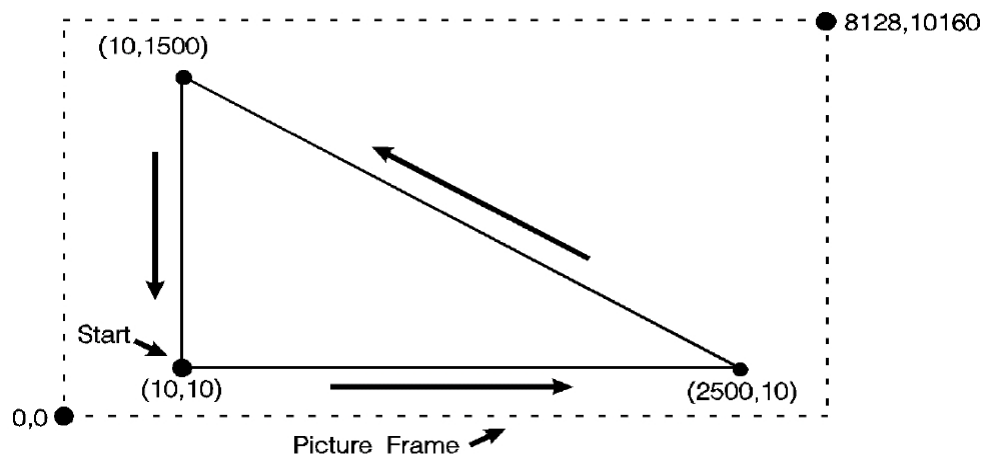
novamente. Numa linguagem baseada em vetores, o arquivo é enviado primeiramente em sua totalidade para o dispositivo de impressão e fica residente na sua memória, para em seguida começar o trabalho de impressão (WIKIPEDIA, 2006).

Com a demanda de arquivos de desenho cada vez mais detalhados e conseqüentemente maiores tanto em dimensões físicas como em seu tamanho lógico, esta era uma desvantagem perceptível da linguagem HP-GL, pois o tempo de recepção dos arquivos se tornavam cada vez maiores. Para superar esta limitação, em 1988 a HP introduziu uma nova versão de sua linguagem de controle, que foi chamada de HP-GL/2 (GOVERT, 2006).

Esta nova versão foi completamente re-escrita usando agora endereçamento de 32 *bits* para permitir impressões mais longas que as anteriores e uma nova tecnologia de compressão de dados chamada de *polyline encoding* foi adicionada na linguagem com a finalidade de reduzir o tamanho dos arquivos e agilizar a sua transmissão e impressão. Foi implementado também suporte a arquivos do tipo *raster* juntamente com arquivos vetoriais e as instruções básicas foram todas herdadas da linguagem HP-GL, sendo criado apenas comandos especiais para cada série, evitando assim que cada modelo possuísse uma linguagem específica (HEWLETT PACKARD, 1996).

#### 2.3.2.1 Exemplo de instruções em HP-GL/2

Neste outro exemplo, de acordo com a Figura 9, agora em HP-GL/2, nota-se que os comando básicos são idênticos ao da versão anterior da linguagem, apenas foram adicionados alguns controles que foram implementados, como a entrada e saída do modo vetorial (HP-GL/2) para o modo padrão do dispositivo.



**Figura 9: exemplo de traçado numa *plotter* em HP-GL/2**

Fonte: The HP-GL/2 and HP-RTL Reference Guide

Instrução	Significado
033E	Envia o código para re-inicializar o dispositivo
033%0B	Envia o código para entrar no modo HP-GL/2
IN;	Inicializa a <i>plotter</i> , zerando todos parâmetros HP-GL/2 para o padrão
SP1PA10,10	Seleciona a caneta n° 1 e move a caneta para posição absoluta 10,10 (x,y)
PD2500,10,10,1500,10,10;	Baixa a caneta no papel e desenha o triângulo
033%0A	Finaliza o modo HP-GL/2
033E	Envia o código para re-inicializar o dispositivo

**Quadro 2: exemplo de codificação em linguagem HP-GL/2**

Fonte: The HP-GL/2 and HP-RTL Reference Guide

### 2.3.2.2 Exemplo do conteúdo de um arquivo em HP-GL/2

Através do exemplo a seguir, visualiza-se o parâmetros e instruções iniciais necessárias para uma *plotter* compatível com a linguagem HP-GL/2 imprimí-lo em

uma mídia suportada pelo equipamento, como um rolo de papel.

```
%-12345X@PJL JOB NAME="Drawing.dwg"
```

```
@PJL SET RENDERMODE=COLOR
```

```
@PJL SET RESOLUTION=300
```

```
@PJL ENTER LANGUAGE=HPGL2
```

```
E%0B;BP5,1,1,"Drawing.dwg"INPS11879,8399RO270IPMC0TR0QL6
```

```
6NP256CR0...
```

Onde:

Instrução	Significado
%-12345X@PJL JOB NAME="Drawing.dwg"	Envia o nome do trabalho para o dispositivo
@PJL SET RENDERMODE=COLOR	Envia o tipo de impressão, neste caso colorida
@PJL SET RESOLUTION=300	Envia a resolução, neste caso 300 dpi
@PJL ENTER LANGUAGE=HPGL2	Envia o código para entrar no modo HP-GL/2
E%0B;BP5,1,1,"Drawing.dwg"INPS11879,8399RO270IPMC0TR0QL66NP256CR0...	Informa o tamanho do papel em unidades de impressão (11879 e 8399) e o restante dados para impressão do arquivo

**Quadro 3: exemplo do conteúdo de um arquivo em linguagem HP-GL/2**

Fonte: The HP-GL/2 and HP-RTL Reference Guide

### 3 METODOLOGIA DA ANÁLISE

Com a intenção de proporcionar uma melhor compreensão do trabalho, este capítulo apresenta inicialmente como uma empresa que presta serviços de plotagem desenvolve suas atividades, e também uma visão geral de como os recursos disponíveis de *hardware e software* são utilizados, para depois demonstrar a análise do sistema através do método de Análise Estruturada.

Foi escolhida esta metodologia por adequar-se com o modelo de desenvolvimento em cascata adotado aqui, pois ele possui fases bem definidas e como o ambiente de análise é bastante conhecido, cada etapa pode ser feita com consistência e sem a possibilidade de propagação de falhas ou riscos para as demais etapas.

Ainda destaca-se outras vantagens como:

- a) uma melhor produtividade em linhas de codificação por dia;
- b) uso mais apropriado do tempo de teste;
- c) os usuários obtêm uma idéia mais clara do sistema proposto pelo diagrama de fluxo de dados, do que a obtida através da narrativa e fluxograma de sistemas físicos;
- d) as interfaces do novo sistema são mostradas de um modo de fácil entendimento.

#### 3.1 O contexto atual

Tomando como exemplo uma empresa especializada em prestação de serviços na área de impressão, tendo como base para o seu trabalho a cópia de arquivos de

sistemas de *CAD*, que é um *software*<sup>8</sup> normalmente utilizado para desenhos técnico em duas e três dimensões, explicaremos ao longo deste capítulo como a empresa desenvolve suas atividades.

A empresa oferece o serviço de impressão em formatos superiores aos normalmente encontrados nos dispositivos de impressão padrão de mercado, ou seja, impressoras de pequeno porte no formato A4. A empresa, tem um ciclo de trabalho bastante rotineiro, recebendo os arquivos dos clientes in loco ou via *e-mail* usando a infra-estrutura da Internet, e que geralmente são projetos dos mais variados ramos de atividades como engenharia civil, elétrica, mecânica, arquitetura ou qualquer outro tipo de projeto.

A empresa presta seus serviços transformando estes arquivos de um meio digital, ou seja, o arquivo de desenho que esta no computador dentro do sistema de *CAD*, para o papel e prosseguindo neste fluxo de andamento do trabalho. Podem ocorrer duas situações que devem ser consideradas quanto a metodologia do processo para o serviço a ser prestado:

- a) a primeira situação acontece quando o cliente entrega à empresa um arquivo contendo um ou mais modelos para ser feita a reprodução em papel. Geralmente este arquivo está em um formato padrão utilizado pela grande maioria dos usuários de sistemas *CAD*, ou seja, em formato *DWG*. Para uma correta graficação deste arquivo, a empresa deverá possuir este *software*, e ainda assim muitas vezes podem ocorrer diferenças entre versões do mesmo produto, resultando em discrepâncias no resultado esperado;
- b) a segunda situação ocorre quando o cliente imprime digitalmente o arquivo, isto é, ele faz todo procedimento de impressão como se tivesse o equipamento

---

<sup>8</sup> *Software*: programa de computador

necessário para tal tarefa, chamado de *plotter*<sup>9</sup>, porém ele não tem, apenas simula que o possui e manda imprimir para um arquivo com a configuração pré-estabelecida do seu sistema, para somente depois enviar para a empresa que irá imprimir finalmente este arquivo em papel.

Na primeira situação, a empresa deve possuir o *software*, como já foi mencionado, porém como este programa custa US\$ 5 mil (em setembro de 2006 segundo a empresa Grapho<sup>10</sup>), o negócio pode tornar-se inviável, pois o *software* será sub-utilizado e o produto final não tem valor agregado, por se tratar de papel. Para esta situação existem programas como *Voloview*<sup>11</sup>, que só imprimem arquivos do tipo *DWG*, por um custo mais acessível, mas ainda assim não garantindo a integridade do desenho, podendo o resultado ser diferente do esperado no momento de entrega do material finalizado ao cliente e, tratando-se de projetos, a fidelidade na transposição do meio digital para o papel torna-se imprescindível neste momento.

Analisando a segunda situação, nota-se que o mercado de *plotter* consolidou um formato de arquivo de impressão vetorial chamado *HP-GL/2*, de propriedade da empresa Hewlett Packard (HP), que tornou-se um dos mais usados, inclusive por outros fabricantes de *plotters*, como podemos notar em especificações técnicas de fabricantes como *Encad* e *CalComp*. Assim, este formato é compatível e está presente em quase todas as *plotters* de impressão vendidas atualmente. Este tipo de equipamento tem características especiais, podendo trabalhar com folhas unitárias de diversas gramaturas e tamanhos, ou rolos de papéis nos formatos de 60, 90 ou 106 centímetros de largura por 50 metros de comprimento, sendo estes últimos mais usados pela sua praticidade.

---

<sup>9</sup> *Plotter*: impressora de grande formato

<sup>10</sup> Grapho: empresa que representa a Autodesk no sul do Brasil

<sup>11</sup> Voloview: programa da Autodesk que possui somente módulo de impressão



No início dos anos 90 uma plotter era definida como uma impressora específica para desenhos vetoriais<sup>12</sup>. Atualmente, uma plotter pode tanto trabalhar no modo vetorial como no modo raster<sup>13</sup>.

### 3.1.1 As vantagens da utilização de arquivos HP-GL/2

O uso de arquivos no formato HP-GL/2 tornou-se bastante popular, por estes possuírem uma série de vantagens para o desenhista ou projetista, que podemos relacionar abaixo:

- a) proteção contra cópia indevida do seu desenho, visto que ninguém pode copiar ou alterar, apenas imprimir;
- b) fidelidade do seu projeto, pois o arquivo de impressão será gerar uma cópia idêntica, com todos os componentes como fontes, padrões de preenchimento (hachuras), etc., que ele possui instalado no seu sistema;
- c) não haverá problemas de incompatibilidade, pois a linguagem HP-GL/2 trabalha apenas com as primitivas de desenho já definidas, ou seja, linhas, círculos, arcos, etc.;
- d) tempo de transferência e tamanho do arquivo reduzidos, seja ele por meio eletrônico (*e-mail*, FTP, etc..) ou através de disquete ou outra mídia física, respectivamente; os arquivos de impressão geralmente são 70% menores que os arquivos do projeto em no seu formato nativo.

A estas facilidades do ponto de vista do desenhista somam-se outras, considerando a posição da empresa prestadora do serviço de impressão, como por

---

<sup>12</sup> *desenhos vetoriais: desenhos digitais definidos por equações matemáticas*

<sup>13</sup> *raster*: imagens que contém a descrição de cada pixel, chamadas também de mapa de bits ou bitmaps

exemplo:

- a) a isenção quanto a qualquer erro detectado no arquivo pelo seu autor;
- b) a facilidade de impressão, pois não precisará “diagramar” o projeto no papel, bastando apenas o comando de impressão para efetua-la;
- c) a redução de custos, pois não necessitará dos sistemas de *CAD* (em suas mais variadas versões) que geraram os arquivos.

Outra particularidade importante que deve ser notada é que os arquivos neste formato possuem sua estrutura de dados no formato de texto plano, sendo possível visualizar o conteúdo das instruções com um editor de texto comum, como o *Notepad*<sup>14</sup>. Pelo fato de existir esta característica, pode-se extrair informações relevantes destes arquivos, como o nome do trabalho, o tipo e a resolução da impressão, o tamanho do papel a ser utilizado, etc.

### 3.1.2 Os problemas da utilização de arquivos HP-GL/2

Porém, nem tudo são vantagens, pois para cada problema resolvido em uma etapa, pode surgir outros mais ao longo da operação. Como vimos na primeira situação mencionada no item 2.1, não possuímos muitas alternativas, pois já existe uma solução definida por grandes fabricante de *software* para esta situação, porém ela é totalmente proprietária e fechada, não permitindo nenhuma integração com outro tipo de programa. Entretanto na segunda situação, encontramos alguns problemas críticos para o desenvolvimento do negócio, como podemos observar abaixo:

- a) este tipo de trabalho de impressão é calculado por metro (quadrado ou linear) ou por formatos que já são um padrão no ramo de projetos (formato A0,

---

<sup>14</sup> *Notepad*: programa simples para edição de textos, também chamado de Bloco de Notas, que acompanha o sistema operacional *Windows*.

A1, etc.). Se o cliente não documentou ou não sabe o tamanho de cada arquivo que foi gerado para impressão, não existe uma maneira prática e rápida do operador da *plotter* saber o que será impresso, somente se ele imprimir os arquivos, sem ter condições de prever o tamanho de papel mais adequado a ser utilizado;

b) se o cliente desejar apenas um orçamento, também não existe uma maneira de atendê-lo sem fazer toda impressão;

c) do mesmo modo que se o cliente executou uma falha na operação de impressão, como um tamanho de papel muito grande ou muito pequeno para o seu projeto, o operador não tem como visualizar este arquivo, e como nos outros casos já citados, terá que imprimir para descobrir qual será o resultado.

Todos os casos acima podem gerar um certo descontentamento, tanto da parte do cliente, como da parte operacional da empresa, pois o resultado não será o esperado por ambos. O cliente não ficará satisfeito por que sua expectativa era que o resultado final fosse o esperado. A empresa também não ficará satisfeita, pois não poderá cobrar do cliente por um serviço que para ele foi insuficiente, mesmo que ela o tenha prestado mais corretamente possível, e em muitos casos, a empresa vai ter que arcar com o prejuízo do material e do tempo que foi desperdiçado.

Observa-se que também existem programas que imprimem arquivos diretamente do formato HP-GL/2, como por exemplo o *PlotVision*<sup>15</sup> e o *Unplot*<sup>16</sup>, porém somente são vendidos no exterior, dificultando assim sua compra, o seu licenciamento e sua operação, por ambos serem feitos em outra língua diferente da portuguesa utilizada no Brasil.

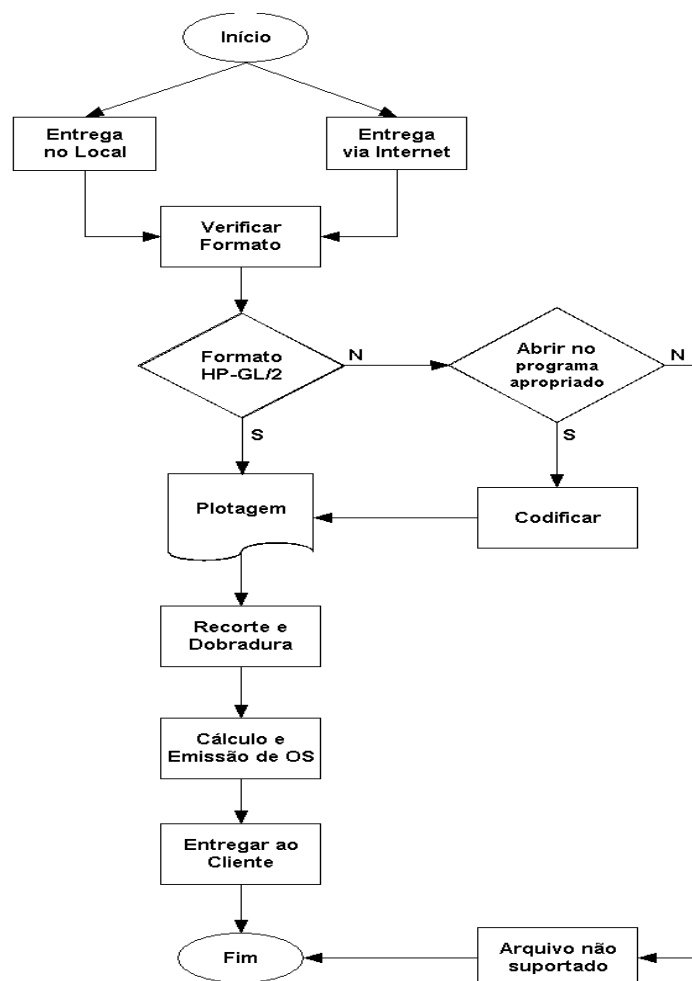
---

<sup>15</sup> *PlotVision*: desenvolvido pela *Torro Softworks* ([www.torro-softworks.com](http://www.torro-softworks.com)), acessado em 29/ago./2006

<sup>16</sup> *Unplot*: desenvolvido pela *Mountain Software*

### 3.2 Análise do processo de trabalho

Realizou-se um estudo detalhado do fluxo de trabalho de uma empresa pequeno porte chamada Drast Informática Ltda, da cidade de Taquara, no estado do Rio Grande do Sul. Esta empresa atua no mercado de impressões de grandes formatos há nove anos e é pioneira no uso desta tecnologia na região do Vale do Paranhana, possuindo grande experiência na área. Todo o fluxo de trabalho foi monitorado na empresa através do método de observação direta, desde o início do processo da entrada de serviços, passando pelo seu andamento e até a sua finalização, culminado na entrega ao cliente, como é mostrado na figura do diagrama a seguir:



**Figura 10: diagrama do fluxo de trabalho**

### 3.3 Análise dos sistemas de informação utilizados

Para a coleta de mais dados relevantes ao desenvolvimento do sistema, foi também realizado uma entrevista com o gerente da empresa e seus colaboradores, a fim de obter um resultado mais adequado com as suas necessidades e fazer um levantamento dos recursos de *hardware* e *software* disponíveis. Desta entrevista, que encontra-se no anexo 1 do trabalho, extraímos as seguintes informações:

Recursos de <i>hardware</i>	Descrição
1 computador Intel Pentium 4 3.0 Ghz, 1 GB de memória RAM, 80 GB de disco rígido, monitor de 17"	Usado como servidor de arquivos e internet
1 computador Intel Pentium 4 Dual Core 3.0 Ghz, 1 GB de memória RAM, 80 GB de disco rígido, monitor de 17"	Usado para gerenciar a Plotter e como estação de trabalho
1 computador Intel Pentium Celeron 600 MHz, 128 MB de memória RAM, 4 GB de disco rígido, monitor de 15"	Usado como estação de trabalho
1 Plotter HP DesignJet 800 42"	Usada para impressões em grandes formatos

**Quadro 4: recursos disponíveis de *hardware***

Recursos de <i>software</i>	Descrição
Linux Fedora Core 4	Usado como sistema operacional do servidor de arquivos e internet
Microsoft Windows XP Professional	Usado como sistema operacional do computador que gerencia a Plotter

Recursos de <i>software</i>	Descrição
Microsoft Windows 98	Usado como sistema operacional da estação de trabalho
Autocad LT 98	Usado para abrir arquivos DWG compatíveis com AutoCAD R14, fabricado pela Autodesk
VoloView 6.0	Usado para abrir arquivos DWG compatíveis com AutoCAD 2000 e posteriores, fabricado pela Autodesk
Hábil Gerenciador Financeiro	Usado para o controle das finanças, fabricado pela Koinonia Software e de uso gratuito
OpenOffice.org 2.0	Usado para planilhas e documentos de texto, mantido pela comunidade de Software Livre e de uso gratuito

**Quadro 5: recursos disponíveis de *software***

### 3.4 Análise do sistema a ser desenvolvido

Com base nas informações coletadas com os procedimentos descritos nos itens 3.1 e 3.2, iniciam-se as seguintes fases de análise para o desenvolvimento do sistema.

#### 3.4.1 Definição dos requisitos

O Sistema de Pré-Visualização e Controle de Arquivos HP-GL/2, chamado a partir deste momento de *PlotControl*, tem o objetivo de tornar mais dinâmicas e automatizadas as tarefas, que até este momento eram feitas e calculadas de uma

forma manual e posteriormente lançados numa planilha eletrônica. Esta é uma mudança que transforma radicalmente sistema de trabalho e de fundamental importância, tornando-o mais evoluído do ponto de vista tecnológico e satisfazendo assim completamente a necessidade atual da empresa. O *PlotControl* irá adaptar-se perfeitamente neste ambiente empresarial, sem alterar a maneira de como as pessoas interagem com o restante do processo físico de trabalho.

O sistema *PlotControl* irá atuar somente nas seguintes operações:

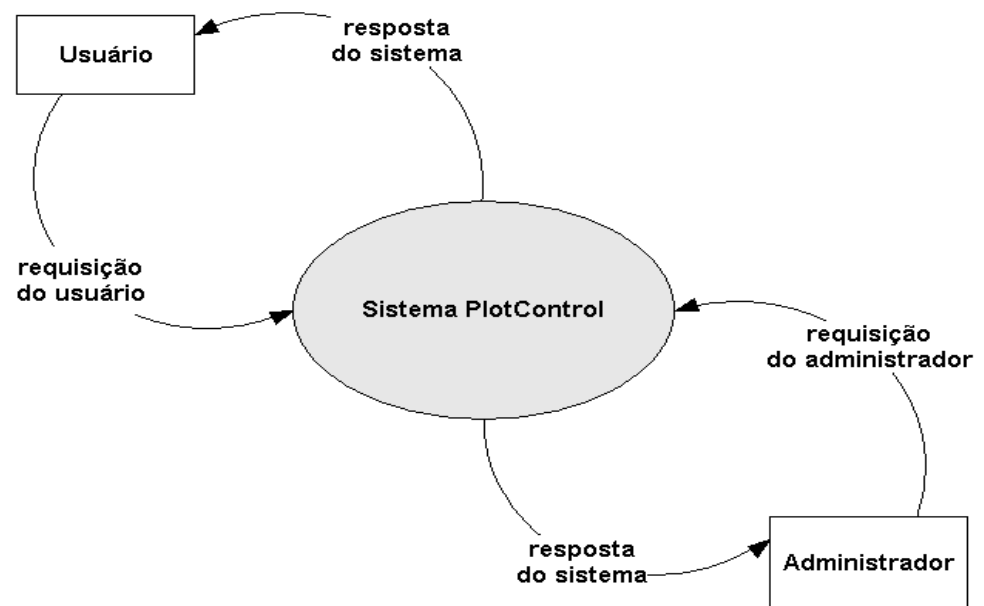
- a) controle dos usuários que podem operar o sistema;
- b) visualização de arquivos codificados com a linguagem HP-GL/2;
- c) cadastro de clientes, tipos de serviços e ordens de serviços;
- d) envio do arquivo para *Plotter* ;
- e) cálculo e emissão das ordens de serviços.

Após a fase de levantamento de requisitos inicia-se a fase de documentação do projeto, fase para qual foi adotada a metodologia de Análise Estruturada, constituída dos seguintes componentes: Diagrama de Fluxo de Dados (DFD), Dicionário de Dados, Especificação de Processos e Diagramas entidade-relacionamentos

### 3.4.2 Diagrama de fluxo de dados (DFD)

O DFD de mais alto nível, que representa todo o sistema como um único processo é conhecido como diagrama de contexto e é composto por fluxos de dados que mostram as interfaces entre o sistema e as entidades externas. O diagrama é uma forma de representar o objeto do estudo, o

projeto, e sua relação ao ambiente. Um diagrama de contexto permite identificar os limites dos processos, as áreas envolvidas com o processo e os relacionamentos com outros processos e elementos externos à empresa (ex.: clientes, fornecedores) (WIKIPEDIA, 2006, p. 1).



**Figura 11: diagrama de contexto ou de nível 0**

Define-se aqui também os Diagramas de Fluxo de Dados de diversos níveis, que representam uma visão estruturada das principais funções, demonstrado a seguir:



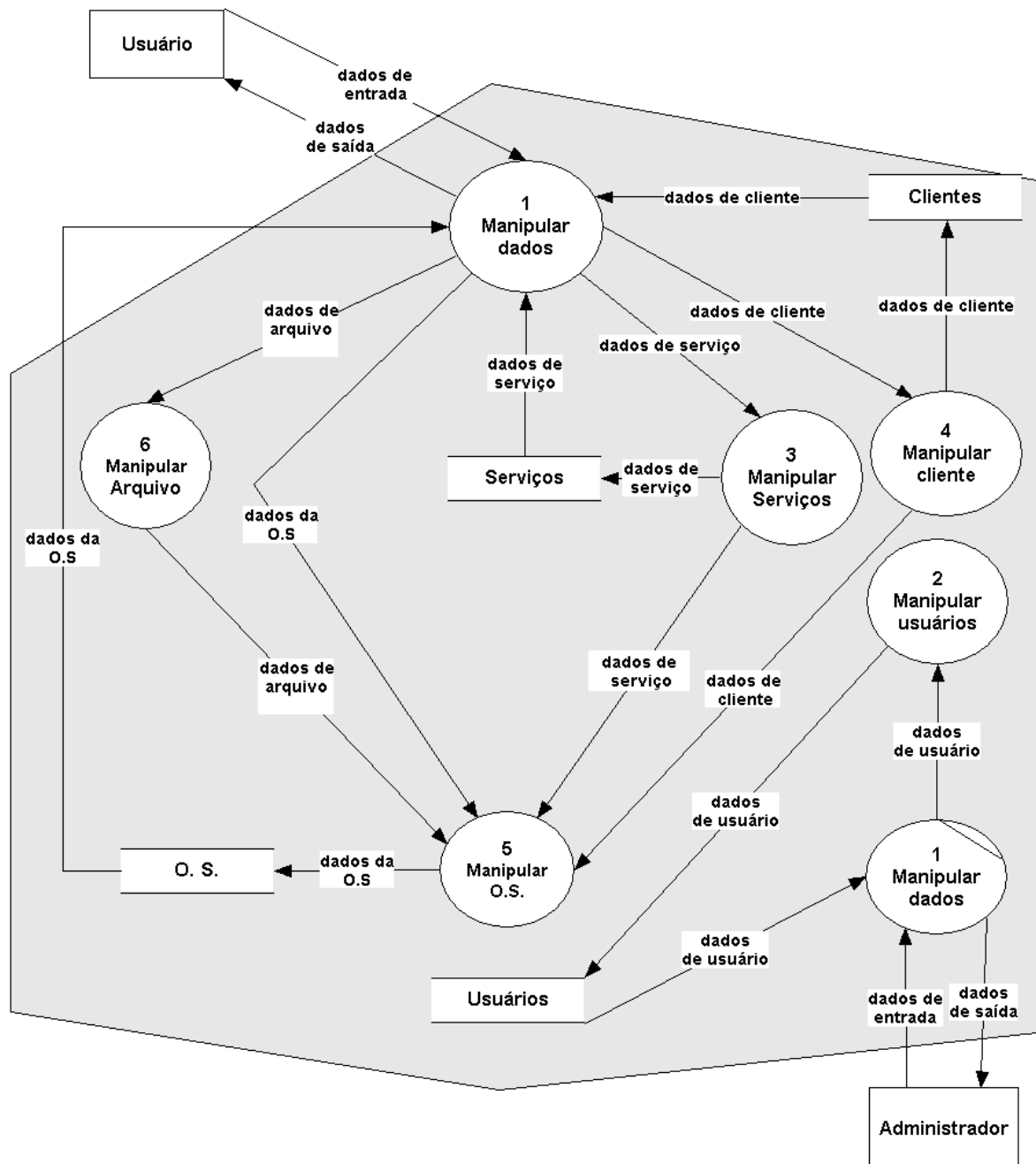


Figura 12: diagrama de fluxo de dados de nível 1

E os demais níveis como segue:

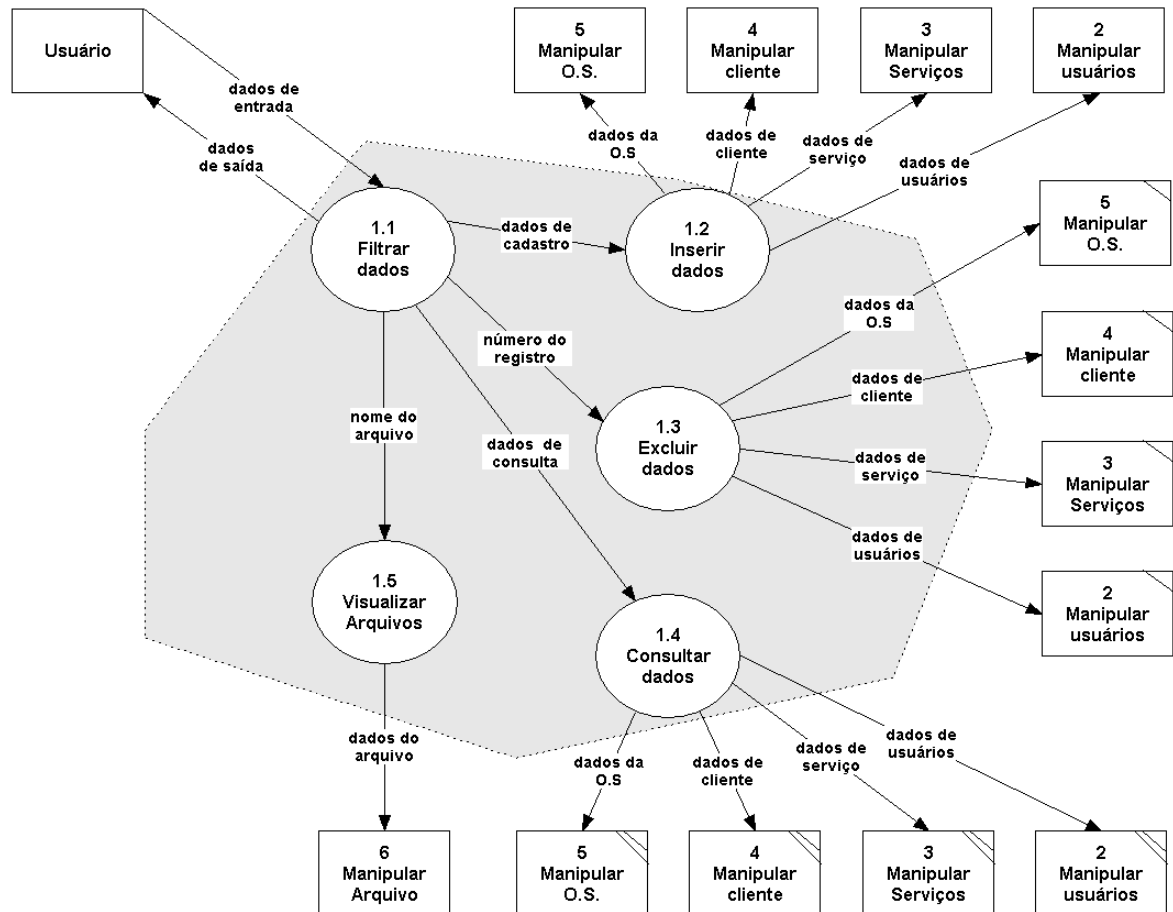
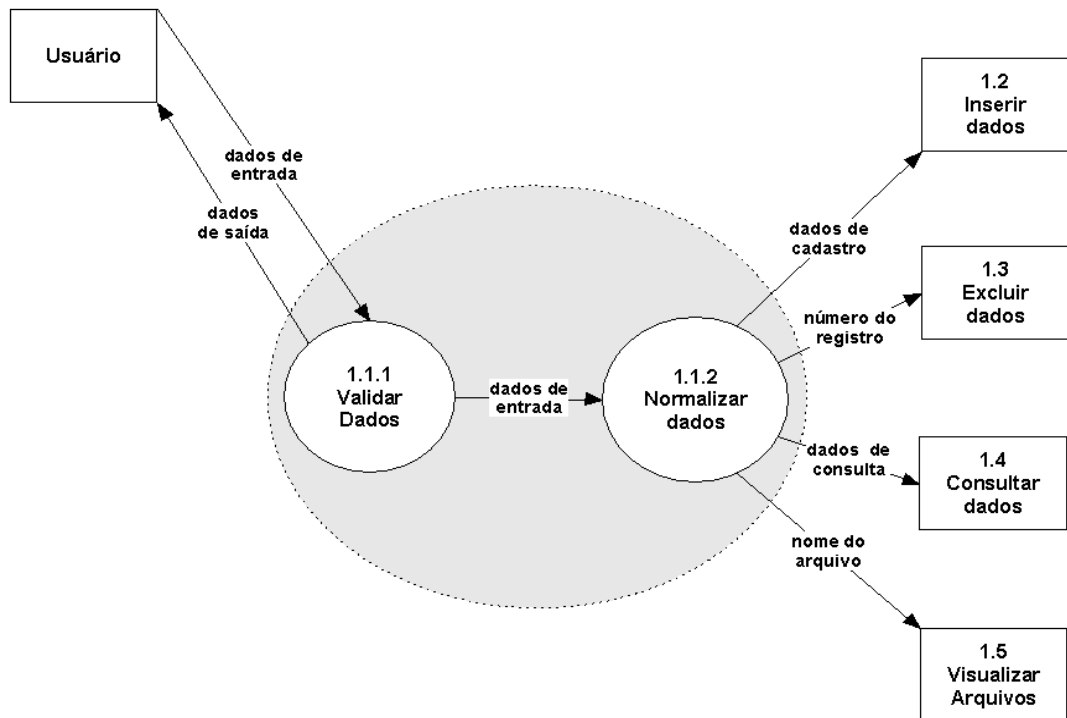


Figura 13: diagrama de fluxo de dados de nível 2 do processo 1



**Figura 14: diagrama de fluxo de dados de nível 3 do processo 1.1**

Após estes 3 níveis, obtêm-se os processos 1.1.1 (Validar Dados) e 1.1.2 (Normalizar dados), que neste caso são indivisíveis e por esta razão chamam-se Unidades Funcionais, e que mais adiante serão descritas na Especificação de Processos, juntamente com o restante dos processos do nível 2, que também são Unidades Funcionais.

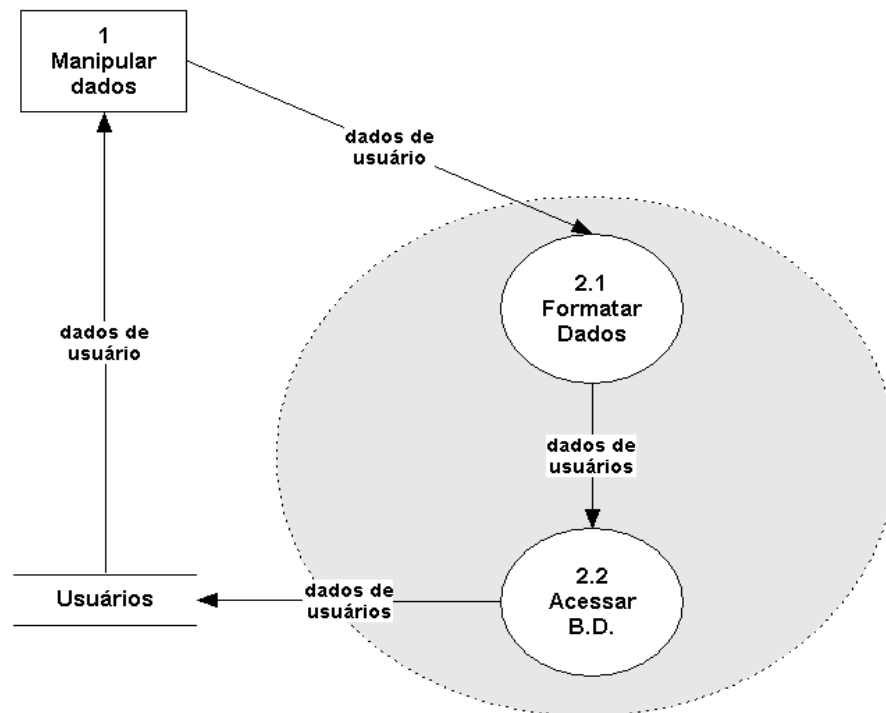


Figura 15: diagrama de fluxo de dados de nível 2 do processo 2

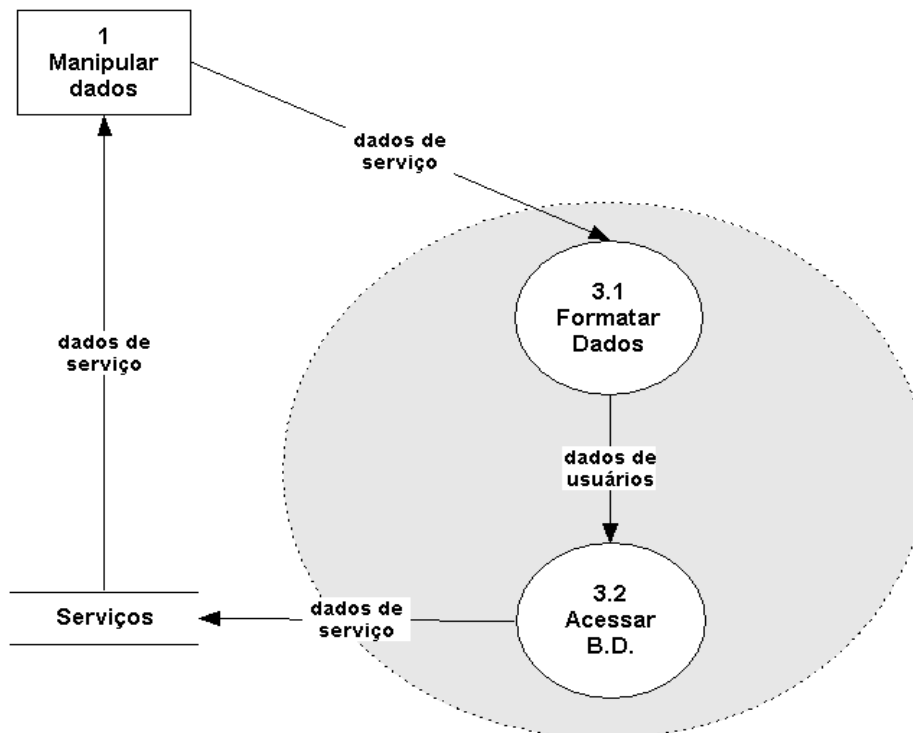


Figura 16: diagrama de fluxo de dados de nível 2 do processo 3

### 3.4.3 Dicionário de dados

“Listagem organizada de todos os elementos de dados pertinentes ao sistema, com definições precisas e rigorosas para que o usuário e o analista de sistemas possam conhecer todas as entradas, saídas, componentes de depósitos e cálculos intermediários” (YOURDON, 1990, pg. 236).

Neste item é apresentado o Dicionário de Dados, utilizando como base os DFD's demonstrados anteriormente e os respectivos fluxos de dados que foram citados.

DADOS DE ENTRADA = [ DADOS DE CADASTRO   NÚMERO DO REGISTRO   DADOS DE CONSULTA   NOME DO ARQUIVO ]
DADOS DE CADASTRO = [ DADOS DE USUÁRIOS   DADOS DE SERVIÇO   DADOS DE CLIENTE   DADOS DA O.S. ]
NÚMERO DO REGISTRO = NÚMERO ATRIBUÍDO PELO SISTEMA
DADOS DE CONSULTA = DADOS DE CADASTRO
NOME DO ARQUIVO = 1{CARACTER-VÁLIDO}255
CARACTER-VÁLIDO = [A-Z a-z 0-9 '  ]

**Quadro 6: dicionário de dados dos processos iniciais**

Para uma melhor compreensão, seguem os fluxos de dados separados por depósitos de dados.

DADOS DE USUÁRIOS = ID+USERNAME+NOME+SENHA+SETOR+ADMIN
ID = NÚMERO ATRIBUÍDO PELO SISTEMA
USERNAME = 1{CARACTER-VÁLIDO}15
NOME = 1{CARACTER-VÁLIDO}45
SENHA = 1{CARACTER-VÁLIDO}10
SETOR = 0{CARACTER-VÁLIDO}20
ADMIN = [S   N]
NÚMERO-VÁLIDO = [0-9]
CARACTER-VÁLIDO = [A-Z a-z 0-9 ' ' ]

**Quadro 7: dicionário de dados do fluxo Usuário**

DADOS DE CLIENTE = {ID+NOME+ENDERECO+CEP+ID_CIDADE+FONE_COML+FONE_CEL+CNPJ_CPF+INSC_ESTADUAL+CONTATO+E-MAIL}
ID = ATRIBUÍDO PELO SISTEMA
NOME = 1{CARACTER-VÁLIDO}45
ENDERECO = 0{CARACTER-VÁLIDO}60
CEP = 0{NÚMERO-VÁLIDO}8
ID_CIDADE = 0{CARACTER-VÁLIDO}45
FONE_COML = 0{NÚMERO-VÁLIDO}20
FONE_CEL = 0{NÚMERO-VÁLIDO}20
CNPJ_CPF = 0{NÚMERO-VÁLIDO}14
INSC_ESTADUAL = 0{NÚMERO-VÁLIDO}20
CONTATO = 0{CARACTER-VÁLIDO}30
E-MAIL = 0{CARACTER-VÁLIDO}30
NÚMERO-VÁLIDO = [0-9]
CARACTER-VÁLIDO = [A-Z a-z 0-9 ' ' ]

**Quadro 8: Dicionário de dados do fluxo Cliente**

DADOS DE SERVIÇO = {ID+DESCRICAO+UNIDADE+PRECO}
ID = ATRIBUÍDO PELO SISTEMA
DESCRICAO = 1{CARACTER-VÁLIDO}50
UNIDADE = 1{CARACTER-VÁLIDO}3
PRECO = {NÚMERO-VÁLIDO}
NÚMERO-VÁLIDO = [0-9]
CARACTER-VÁLIDO = [A-Z a-z 0-9 ' ' ]

**Quadro 9: dicionário de dados do fluxo Serviço**

#### 3.4.4 Especificação de processos

“Define o que deve ser feito para transformar entradas em saídas. Ela é uma detalhada descrição da orientação empresarial do usuário executada pelas bolhas”(YOURDON, 1990, pg. 253).

Neste item é apresentado as Especificações de Processos que demonstram as Unidades Funcionais descritas no final do item 3.3.2. Pode-se utilizar os métodos de Português Estruturado, Árvore de Decisão ou Tabela de Decisão. Para os exemplos apresentados a seguir, será utilizado o método de Português Estruturado com fluxo de dados Dados de Usuário.

O processo 1.1 (Filtrar Dados) foi dividido nas Unidades Funcionais 1.1.1 (Validar Dados) e 1.1.2 (Normalizar Dados), como segue:

```

1.1.1. VALIDAR DADOS (dados de usuário) {
    SE ( username.tamanho < 1 ) OU (username.tamanho > 15 )
        RETONAR ("tamanho de campo incompatível")
    SE ( nome.tamanho < 1 ) OU (username.tamanho > 45 )
        RETONAR ("tamanho de campo incompatível")
  
```

```

        SE ( senha.tamanho < 1 ) OU (username.tamanho > 15 )
            RETONAR ("tamanho de campo incompatível")
    }

```

```

1.1.2. NORMALIZAR DADOS (dados de usuário) {
    TRANSFORMAR EM MAIÚSCULAS (dados de usuário)
}

```

Os processos 1.2 (Inserir Dados), 1.3 (Excluir Dados) e 1.4 (Consultar Dados) não são possíveis de dividir, logo são Unidades Funcionais e seguem com suas respectivas especificações de processos.

```

1.2. INSERIR DADOS (dados de usuário) {
    ENVIAR (dados de usuário) PARA O DEPÓSITO usuários
    GRAVAR (dados de usuário) NO DEPÓSITO usuários
}

```

```

1.3 EXCLUIR DADOS (número do registro) {
    SELECIONAR (número do registro) NO DEPÓSITO usuários
    APAGAR (número do registro) NO DEPÓSITO usuários
    GRAVAR (dados de usuário) NO DEPÓSITO usuários
}

```

```

1.4 CONSULTAR DADOS (dados de consulta) {
    SELECIONAR (dados de consulta) NO DEPÓSITO usuários
    SE (dados de consulta) EXISTE
        EXIBIR (dados de consulta)
}

```

O processo 2 (Manipular Usuários) é composto dos processos 2.1 (Formatar



Dados) e 2.2 (Acessar B.D.) que são responsáveis pela interface de tela e acesso aos depósitos de dados. Segue o exemplo com o fluxo dados de usuário

```

2.1 FORMATAR DADOS (dados de usuário) {
    ORDENA (dados de usuário) PARA INTERFACE usuários
}

2.2 ACESSAR B.D. (dados de usuário) {
    CONECTA (dados de usuário) AO DEPÓSITO usuários
}

```

O processo 3 (Manipular Serviços) é composto da mesma estrutura do processo 2 (Manipular Usuários), porém com o seu fluxo de dados e depósito de dados correspondentes. Nos demais processos também são utilizadas estruturas similares.

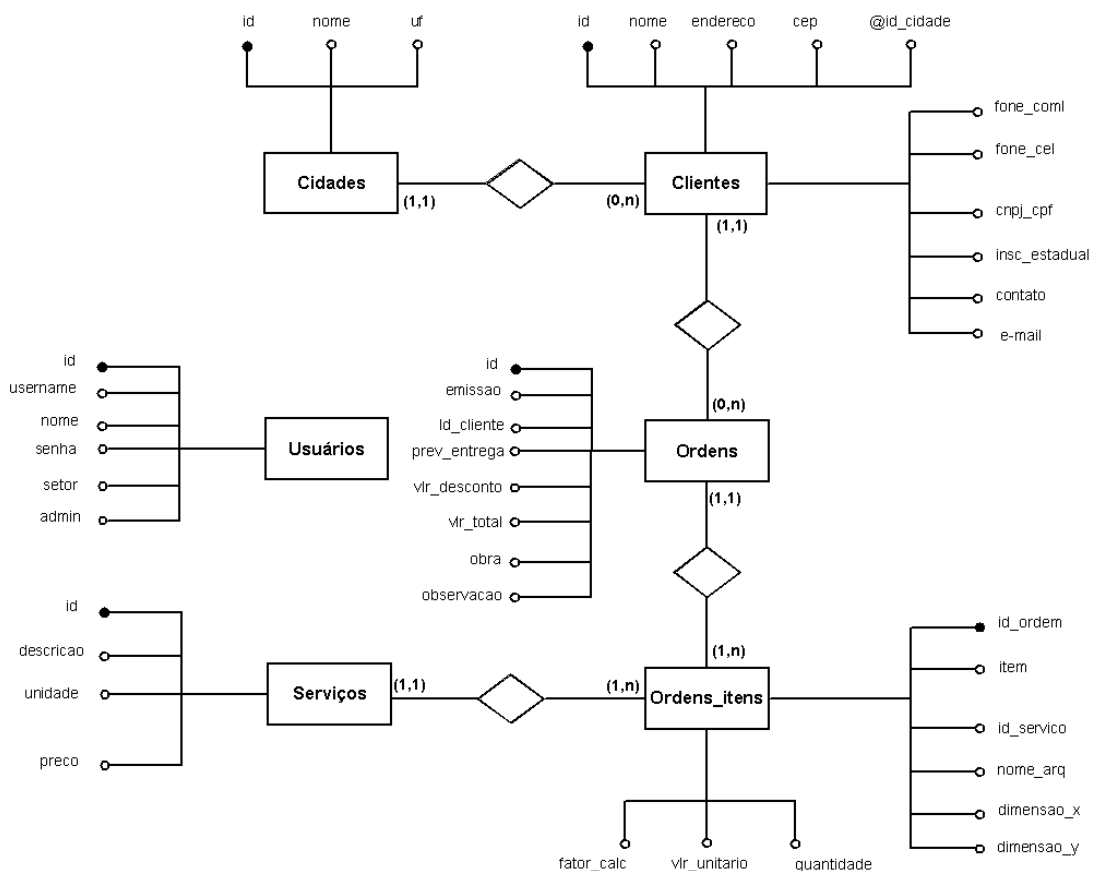
Os exemplos de Diagrama de Fluxo de Dados com seus diversos níveis, Dicionário de Dados e Especificação de Processos mostrados anteriormente foram citados de forma resumida, por esta razão foi incluída no texto apenas parte da documentação gerada durante processo de análise, como subsidio para validação da metodologia de Análise Estruturada num exemplo prático. O material apresentado neste texto, integra portanto a documentação do processo de análise de sistema, o qual foi integralmente desenvolvido para a posterior implementação do software proposto neste projeto. Foi tomada esta decisão para a leitura não tornar-se repetitiva, pois o sistema *PlotContol* foi concebido seguindo o padrão do sistema operacional *Windows* e com todas as telas e opções disponíveis muito similares, e assim sendo, ele utiliza basicamente os mesmos modelos de processos para cada opção disponível, variando apenas os depósitos e tipos de dados dos fluxos.

### 3.4.5 Diagrama de entidade relacionamento (DER)

De acordo com Yourdon (1990, p. 290): “O diagrama entidade relacionamento é um modelo em rede que descreve a diagramação dos dados armazenados de um sistema em alto nível de abstração”.

Seu propósito maior é visualizar o relacionamento entre tabelas de um banco de dados, no qual as relações são construídas através da associação de um ou mais atributos destas tabelas.(WIKIPEDIA, 2006).

A seguir é demonstrado o diagrama de entidade relacionamento:



**Figura 17: diagrama de entidade relacionamento**

## 4 MODELAGEM E DESENVOLVIMENTO DO SISTEMA

Neste capítulo defini-se como será a arquitetura interna e a interface externa do sistema *PlotControl*, que deve ser clara e intuitiva o suficiente para permitir rápido aprendizado, sendo muito importante para a compreensão do sistema como um todo.

### 4.1 Definição da tecnologia de desenvolvimento

O sistema *PlotControl* será desenvolvido em sua totalidade com a ferramenta de programação *Delphi versão 7*, que é uma *IDE*<sup>17</sup> composta por editor de texto, compilador/ligador e depurador fabricada pela *Borland*. Esta ferramenta foi escolhida por apresentar as seguintes vantagens:

- a) é bastante maduro e estável, tendo comprovado seu sucesso ao longo dos anos;
- b) o ambiente de desenvolvimento é totalmente visual, possuindo performance e versatilidade excelente;
- c) facilidade de uso, aprendizado e desenvolvimento;
- d) possui diversos componentes disponíveis;
- e) facilidade para desenvolver aplicações que acessam dados;
- f) vasta bibliografia a respeito da linguagem.

Também será utilizado um componente para visualização de arquivo HP-GL/2 desenvolvido por Andrey Radchenko, com licenciamento do tipo *freeware*<sup>18</sup> e com código fonte disponível, permitindo assim que se faça qualquer adaptação necessária

---

<sup>17</sup> *IDE*: *Integrated Development Environment* ou Ambiente de Desenvolvimento Integrado

<sup>18</sup> *freeware*: programa de computador gratuito

de acordo com a necessidade.

Para armazenamento dos dados, será utilizado o gerenciador de banco de dados Firebird versão 2.0, onde destaca-se as seguintes vantagens e características:

- a) instalação, configuração e administração simplificada;
- b) é rápido, multi-plataforma e com muitas funcionalidades que facilitam o desenvolvimento;
- c) simplificação nas atividades de cópia de segurança e restauração;
- d) código aberto e licença do tipo freeware;
- f) utiliza a linguagem SQL padrão;
- e) possuir uma excelente integração com a linguagem escolhida (*Delphi*).

O ambiente de execução do *software* desenvolvido será o sistema operacional *Microsoft Windows* na versão *XP Professional*, por possuir um ótima relação custo-benefício e um bom nível de estabilidade.

O desenvolvimento é composto das seguintes etapas, como é demonstrado detalhadamente na seqüência.

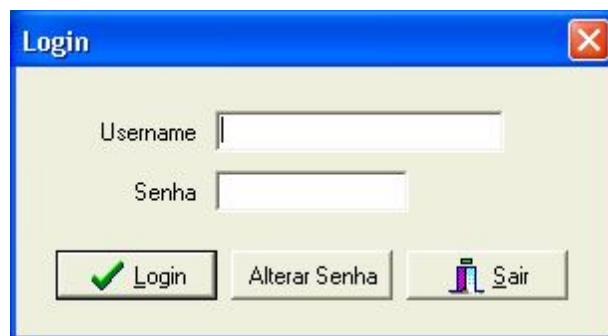
#### **4.2 Prototipação da interface de comunicação com o usuário**

O projeto de interfaces de *softwares* sempre envolve questões subjetivas associadas à critérios psico-sociais, tais como ergonomia, experiência dos usuários, requisitos não funcionais do *software*, etc. Para minimizar os riscos associados a esta parte do sistema, utiliza-se um processo que envolve prototipação (modelo evolucionário de processo de software). Desta forma, requisitos não bem definidos

podem ser validados pelo usuário final.

Demonstra-se aqui o modelo de todas as telas do sistema com base nos levantamentos realizados, priorizando a clareza e a fácil compreensão das opções dos menus e botões da janela, para causar o menor impacto possível no momento da implantação. Para a construção destas telas foi utilizado a própria a ferramenta de programação *Delphi*, como forma de antecipar uma visão real do sistema, pois todas as telas possuem botões com nomes apropriados e dicas do que o botão executa, seguindo os padrões do sistema operacional.

A Seguir, as figuras das telas com a descrição de sua funcionalidade para o sistema:



**Figura 18: tela de entrada no sistema**



Figura 19: tela de principal do sistema

The image shows a Windows-style application window titled "Cadastro de Usuários". The window has a blue title bar with standard minimize, maximize, and close buttons. Below the title bar is a toolbar with several icons for navigation and editing. To the right of the toolbar are two buttons: "Aplicar" (with a green checkmark icon) and "Desfazer" (with a red 'X' icon). The main area is a light beige color and contains a form with the following fields: "Código" (with a value of "1"), "Username" (with a value of "admin"), "Nome" (with a value of "ADMINISTRADOR"), "Senha" (with masked characters "xxxxxx"), and "Setor" (empty). Below these fields is a checkbox labeled "Administrador" which is checked. At the bottom of the window, there is a status bar that reads "Formulário".

Figura 20: tela de cadastro dos usuários do sistema

**Cadastro de Serviços**

Aplicar Desfazer

Código 1

Descrição METRO LINEAR PRETO (1000x1000)

Unidade M

Preço unit. 7.000

Lista Formulário

Figura 21: tela de cadastro de serviços do sistema

**Cadastro de Clientes**

Consultar Aplicar Desfazer

Código 0

Nome

Endereço

CEP

Cidade

Fone Coml.

Fone Cel.

CNPJ/CPF

Insc. Estadual

Contato

e-mail

Lista Formulário

Figura 22: tela de cadastro de clientes do sistema

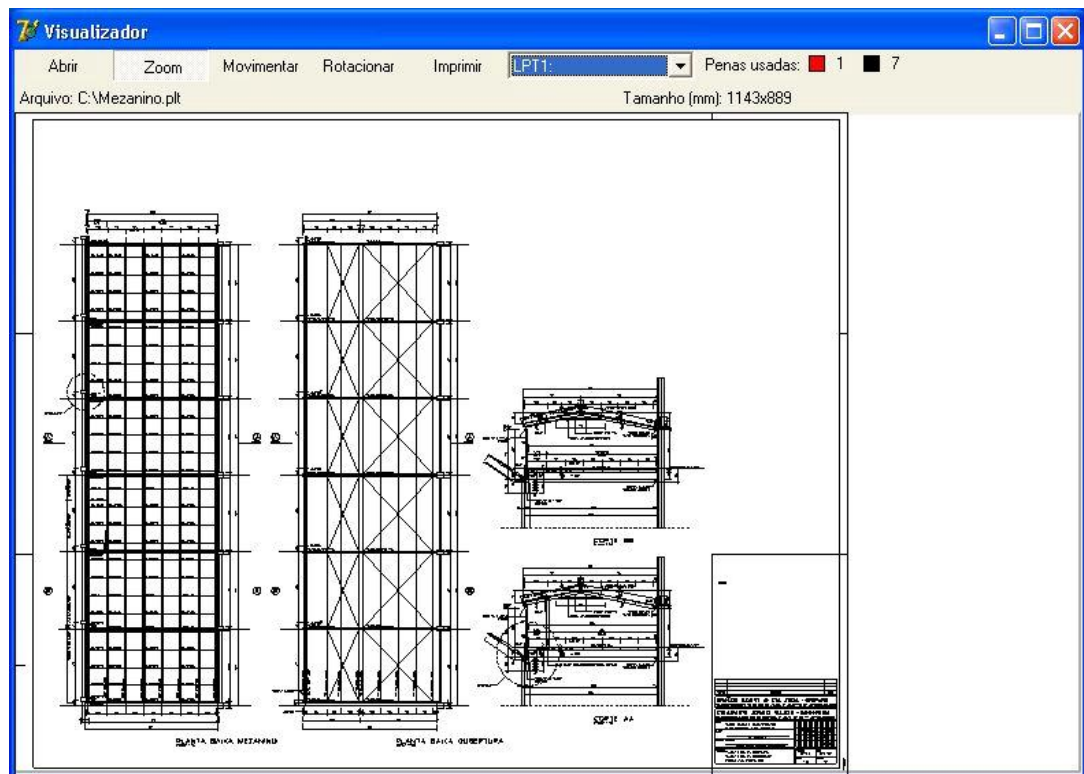


Figura 23: tela de visualização de arquivos do sistema

Item	Cód. Serviço	Serviço	dim X	dim Y	Quantidade	Valor Unitário	Valor Total
1							

Figura 24: tela de ordens de serviço do sistema



### 4.3 Modelagem do Banco de Dados

Atendendo a todos requisitos levantados e definidos anteriormente no item 3, modela-se as tabelas e os campos do banco de dados de uma forma coerente com a necessidade dos tipos de dados envolvidos, auxiliando assim no desempenho global do sistema. Para auxiliar na criação das tabelas foi utilizado o programa IBExpert Personal Edition, versão 2006 para Windows, que possui interface gráfica e com licença do tipo *freeware*, assim como o Firebird versão 2.0.

### 4.4 Codificação

Foi codificado num primeiro momento um protótipo para testes como foi definido na etapa de modelagem, porém sem funcionalidades. Após foi iniciada a inclusão das funcionalidades previstas na etapa de análise em cada função do sistema, com o desenvolvimento dos algoritmos necessários para funcionalidade do sistema.

Uma rotina foi criada para extrair informações dos arquivos desejados, que posteriormente são aproveitadas para o controle e cálculo dos serviços a serem executados, através da utilização de um componente para visualização de arquivo HP-GL/2 desenvolvido por Andrey Radchenko. Este componente já fornecia as dimensões do arquivo, porém no sistema de medidas inglês (polegadas). Então uma mudança se tornou necessária para transformar esta medida no sistema métrico e para extrair o nome do arquivo, juntamente com a tradução da interface de tela para português.

Quanto a estrutura de interface do programa, foi utilizado o modo SDI, tipo

padrão de aplicações do Delphi, que contém uma tela principal e todas as outras telas ficam dependentes desta, mas independentes entre si. Neste tipo de aplicação, pode-se estar com uma tela aberta e abrir uma segunda sem fechar a primeira, enquanto a segunda mantém-se ativa (GARCIA, 2005, p. 88).

Para o sistema *PlotControl* ficar mais flexível foi criado um arquivo de configuração, onde são armazenadas as informações que podem ser configuradas de acordo com cada empresa que for utilizá-lo. Este arquivo chama-se config.ini e deve estar localizado na mesma pasta que o sistema.

#### 4.5 Impressão de Ordem de Serviço

Foi também desenvolvido a opção de impressão da ordem de serviço e um relatório com o serviço executado para cada cliente, porém novas funções poderão ser adicionadas no futuro. Abaixo um exemplo de ordem de serviço:

<b>Drast Informática</b> Rua Pinheiro Machado, 1248 - Centro CEP 95600-000 Taquara / RS Fone (51) 3541 5140 / 3542 2435 E-mail: drast@tca.com.br	
Emissão: <b>30/10/2007</b> Prev. entrega: <b>30/10/2007</b> Ordem de Serviço	
Cliente: <b>CLIENTE</b> Obra: <b>TESTE</b> Observação:	<b>0000280</b>
Item: 1 Serviço: METRO LINEAR COLOR (1000X1000; Unit.:10,000 Qtde.: 2 Valor: 22,86 Arquivo: Mezanino.plt	
<b>Total:</b>	<b>22,86</b>
<b>Desconto:</b>	<b>0,00</b>
<b>Total da OS:</b>	<b>22,86</b>
Reconheço que pagarei a importância desta ordem de serviço.	

Figura 25: ordem de serviço impressa

## 4.6 Teste e Validação

Neste processo executa-se a validação do sistema quanto à sua consistência, precisão e contextualização de requisitos levantados no processo de identificação e de análise, envolvendo então uma revisão de todos os requisitos levantados.

Foi utilizado o método de “caixa preta” para depuração dos erros, executado módulo por módulo, permitindo assim a correção de alguns erros na codificação do sistema e na definição das tabelas e campos do arquivo de banco de dados.

Neste tipo de teste, dados de entrada são fornecidos, o teste é executado e o resultado obtido é comparado a um resultado esperado previamente conhecido. Haverá sucesso no teste se o resultado obtido for igual ao resultado esperado.

A seguir, utiliza-se o módulo Validar Dados do processo Inserir Dados, para um teste de “caixa preta” como exemplo:

Entrada Digitada	Saída Esperada	Saída Real
Gustavo	correta	correta
(campo vazio)	erro	correta

**Quadro 10: exemplo de teste no módulo Validar Dados**

O restante dos problemas estão sendo resolvidos a medida em que as partes do sistema passam a ser colocadas em produção e os erros vão se manifestando.

## 4.7 Instalação

A instalação se dá pela implantação inicial do *software* no ambiente de produção, e suas duas sub-atividades são a configuração e a transferência. O procedimento utilizado foi mover o produto do ambiente de desenvolvimento para o

ambiente de produção, enquanto eram alteradas todas as configurações, tornando o sistema utilizável pelo usuário.

Foi também gerado de um pacote de instalação do sistema *PlotControl*, através dos programas Inno Setup e IS Tool, ambos de versão 5.2, que permitem agregar todos os arquivos necessários para a posterior instalação. O Inno Setup opera através de scripts, sendo esta uma maneira fácil e rápida de customizar-se uma implantação de software. Já o IS Tool é utilizado como uma interface gráfica para o Inno Setup.

Os componentes que fazem parte do arquivo de instalação são:

- a) programa sistema de Plotagem;
- b) gerenciador de banco de dados Firebird;
- c) arquivo de banco de dados padrão;
- d) arquivo de configuração do sistema;
- e) alguns arquivos de bibliotecas compartilhadas pelo sistema operacional.

#### **4.8 Manutenção**

O processo de manutenção de *software* pode ser dividido em duas partes: Manutenção Corretiva e Manutenção Preventiva.

A Manutenção Corretiva se dá em partes do sistema que não estejam funcionando de uma forma satisfatória, como haviam sido planejados. Esta será sempre executada quando se fizer necessária.

Já a Manutenção Preventiva deve ser executada periodicamente, através de uma cópia de segurança completa do sistema, afim de não ocorrer perda dos dados. Quanto a frequência da cópia, sugere-se que seja diária para este sistema, sempre utilizando mídias diferentes, como cd-r, *pendrive*, etc., ou até mesmo a Internet através de *e-mail* ou disco virtual.

## 5 CONCLUSÕES

Como foi exposto ao longo deste trabalho, para toda empresa é fundamental o uso da tecnologia para diferenciar-se no mercado, e ao mesmo tempo manter sob controle todas as tarefas desenvolvidas por ela.

Aqui demonstrou-se como o sistema *PlotControl* foi desenvolvido e utilizado para automatizar ainda mais o controle de ordens de serviço e oferecer uma opção alternativa de visualização de arquivos no formato HP-GL/2.

Com o uso deste sistema agiliza-se o processo de plotagem, porque foi analisado como a empresa trabalha, através de entrevistas e levantamentos, possibilitando assim a definição de um objetivo final, ou seja, desenvolver o sistema da melhor maneira possível e de uma forma intuitiva de operá-lo. Isto foi possível porque foram utilizados os conceitos que a empresa e seus colaboradores já haviam adotado.

Após a implantação, o desenvolvimento das atividades acontece de uma maneira totalmente computadorizada e eliminando re-trabalhos, isto porque o sistema já faz todos os cálculos necessários com maior precisão, minimizando assim o tempo de preenchimento das ordens, pois o operador necessita apenas atribuir o serviço ao cliente já cadastrado no sistema, que identifica, calcula e registra todas tarefas.

Este foi apenas o início de um sistema que ainda poderá ser continuamente desenvolvido, pois outros módulos poderão vir a ser agregados no futuro, como controle de insumos, contas a pagar e receber, integração bancária e com aplicativos para Internet, etc., reduzindo-se ainda mais o tempo despendido e automatizando processos.

## REFERÊNCIAS

AUTODESK. **Unofficial autocad history pages.** Disponível em: <<http://myfeedback.autodesk.com/history/area51.htm>> Acesso em: 07 Agot. 2006.

BANGERT, Lubert Dijkman. **Cstep.** Disponível em: <<http://cstep.lubert.com/hpgl.htm>> Acesso em: 15 Agot. 2006.

BREVARD USER'S GROUP. **A short history of the plotter.** Disponível em: <<http://bugclub.org/beginners/history/PlotterHistory.html>> Acesso em: 23 Set. 2006.

CARLSON, Wayne. **A critical history of computer graphics and animation: Section 3.** Disponível em: <<http://accad.osu.edu/~waynec/history/lesson3.html>> Acesso em: 12 Set. 2006.

COLLEGE OF COMPUTING AT GEORGIA TECH. **Biography of a luminary dr. Ivan E. Sutherland.** Disponível em: <[http://www.cc.gatech.edu/classes/cs6751\\_97\\_fall/projects/abowd\\_team/ivan/ivan.html](http://www.cc.gatech.edu/classes/cs6751_97_fall/projects/abowd_team/ivan/ivan.html)> Acesso em: 15 Out. 2006.

DALTON STATE COLLEGE. **History of cad.** Disponível em: <<http://cadlab.daltonstate.edu/historyofcad.pdf>> Acesso em: 10 Out. 2006.

USP-DEPARTAMENTO DE ARQUITETURA E URBANISMO. **The history of cad.** Disponível em: <[http://www.saplei.eesc.usp.br/sap645\\_apoio/pae\\_2005/historia\\_cad\\_pae\\_2005.pdf](http://www.saplei.eesc.usp.br/sap645_apoio/pae_2005/historia_cad_pae_2005.pdf)> Acesso em: 10 Out. 2006.

GOVERT. **Noliturbare.** Disponível em: <<http://www.noliturbare.com/index.html>> Acesso em: 04 Set. 2006.

HEWLETT PACKARD. **The hp-gl/2 and hp-rtl reference guide: A Handbook for Program Developers.** 2ª ed. Barcelona: 1996.

HP COMPUTER MUSEUM. **Peripheral products.** Disponível em: <<http://hpmuseum.net/exhibit.php?class=4&cat=24>> Acesso em: 12 Ago. 2006.

KAY, David C. , LEVINE, John. **Graphics files formats.** 2ª ed. Windcrest/McGraw-

Hill, 1995.

ROHLEDER, Edison. **História da modelagem sólida**. Disponível em: <<http://www.cce.ufsc.br/~rohleder/Model/Slides/historia/sld001.htm>> Acesso em: 14 Set. 2006.

WIKIPEDIA. **Plotter**. Disponível em: <<http://en.wikipedia.org/wiki/Plotter>> Acesso em: 27 Set. 2006.

WIKIPEDIA. **Calcomp plotter**. Disponível em: <[http://en.wikipedia.org/wiki/Calcomp\\_plotter](http://en.wikipedia.org/wiki/Calcomp_plotter)> Acesso em: 12 Set. 2006.

WIKIPEDIA. **Computervision**. Disponível em: <<http://en.wikipedia.org/wiki/Computervision>> Acesso em: 12 Set. 2006.

YOURDON, Edward. **Análise estruturada moderna**. Rio de Janeiro, Campus, 1990.

WIKIPEDIA. **Diagrama entidade relacionamento**. Disponível em: <[http://pt.wikipedia.org/wiki/Diagrama\\_entidade\\_relacionamento](http://pt.wikipedia.org/wiki/Diagrama_entidade_relacionamento)> Acesso em: 11 Jul. 2007.

BORRIE, Helen. **Dominando firebird**. 1ª ed. Ciência Moderna, 2006.

GARCIA, Carlos A. **Universidade delphi**. 1ª ed. Digerati Books, 2005.

## **ANEXO**



## ANEXO A - Análise do processo de trabalho

Empresa: Drast Informática Ltda

Entrevistados: Luis Gustavo Angeli e Vagner Petzhold

Cargos: Operadores de computador e plotter

Data: 02/02/2007

### 1) Como inicia-se o processo de trabalho?

O cliente nos fornece o arquivo que ele necessita imprimir ou plotar, seja por *e-mail* ou entregando ele aqui na própria empresa.

### 2) Quais os tipos de arquivo que o cliente fornece?

Em sua maioria (aproximadamente 70%) são do tipo PLT com codificação HP-GL/2. No formato DWG são aproximadamente 20% e os 10% restantes são em outros formatos que não são utilizado para projetos (ex: arquivos de planilha eletrônica).

### 3) Como desenvolve-se o processo inicial?

Após a entrega do arquivo, começamos a fazer a impressão de acordo com o número de cópias solicitadas. Cada arquivo pode conter um ou diversos desenhos, dependendo como foi gerado pelo cliente. Depois de cada arquivo impresso é feito uma anotação do tamanho do papel que foi consumido para aquela impressão para calcularmos o valor a ser cobrado.

### 4) Como é cobrado o serviço prestado?

Existe 2 métodos geralmente usado. O metro-linear e o metro-quadrado. Nós preferimos adotar o metro-linear, de acordo com as *plotters* padrão do mercado, que

são de 36" ou 90 centímetros. Então o tamanho do metro-linear é de um metro por noventa centímetros. Acima de noventa centímetros cobra-se a medida maior. Se for abaixo de noventa centímetros cobra-se a medida menor. De posse destas medidas, multiplicamos pelo valor do metro-linear, com segue o exemplo:

metro-linear monocromático: R\$ 7,00.

metro-linear colorido: R\$ 10,00.

Exemplo 1:

Medida da impressão: 0,70 metros x 0,5 metros

Se for em monocromático:  $0,5 \times \text{R\$ } 7,00 = \text{R\$ } 3,50$

Exemplo 2:

Medida da impressão: 1,10 metros x 0,5 metros

Se for em colorido:  $1,10 \times \text{R\$ } 10,00 = \text{R\$ } 11,00$

5) E após este cálculo, qual o procedimento seguinte?

Em 95% dos casos emitimos manualmente uma ordem de serviço (ou vale como chamamos) com os dados relativos àquele serviço e o cliente assina para uma posterior cobrança ao final de cada mês.

6) Como são contabilizados estas ordens de serviço?

Lançamos diariamente o fluxo de trabalho das ordens de serviço numa planilha eletrônica separada em colunas com o nome de cada cliente, afim de contabilizarmos mais facilmente o valor de cada um.

7) E no caso de orçamentos, como funciona o processo?

Se forem arquivos do tipo DWG, abrimos com o programa apropriado,

verificamos as medidas na tela e calculamos os valores seguindo a regra já citada no item 4. Se forem arquivos do tipo PLT, só conseguiremos saber os valores se imprimirmos as cópias.

Existem programas, como *PlotVision* ou *Unplot*, que nos permitem visualizar os arquivos neste formato, porém se o desenhista não colocou o tamanho do papel no selo do projeto, fica difícil de saber o tamanho real da impressão.

8) Quais programas de computador são utilizados atualmente pela sua empresa?

Utilizamos os seguintes programas todos em plataforma Microsoft Windows XP:

- Autocad LT 98 para arquivos DWG compatíveis com AutoCAD R14, fabricado pela Autodesk;
- VoloView para arquivos DWG compatíveis com AutoCAD 2000 em diante, fabricado pela Autodesk;
- Habil Gerenciador Financeiro 6.0 para controle das finanças, fabricado pela Koinonia Software e de uso gratuito;
- OpenOffice.org 2.0 para planilhas e documentos de texto, mantido pela comunidade de software livre e de uso gratuito.

9) O que você acha que poderia ser melhorado nos sistemas de informação atualmente utilizados na empresa?

Nossa necessidade é de um programa que integre estas rotinas todas já mencionadas, elimine o re-trabalho (como o de preencher uma ordem a mão e depois lançá-la em outra planilha), agilize a contabilização dos serviços e que forneça um histórico de impressões de cada cliente de uma forma simples e rápida.