

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO
E AUTOMAÇÃO INDUSTRIAL

UM GERENCIADOR DE PROJETOS DE ENGENHARIA DE SOFTWARE

JOSÉ EVANDRO MOTTA VARGAS

ORIENTAÇÃO:
PROF. DR. MARIO JINO

Dissertação apresentada à Faculdade de Engenharia Elétrica da Universidade Estadual de Campinas, como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica.

UNICAMP - Campinas
Outubro de 1990

Este exemplar corresponde à redação final da tese defendida por José Evandro Motta Vargas e aprovada pela Comissão Julgadora em 12 de outubro de 1990

Mário Jino

00/9/00-748

Agradecimentos .

Ao Prof.Dr.Mario Jino pela orientação eficiente e precisa , e pela atenção e paciência dispensadas para que este trabalho se tornasse possível .

Ao Eng. Fabio Nauras Akhras do Instituto de Automação do CTI , por nos ter possibilitado o uso do SIPS , e aos amigos Márcia Cristina Carvalho Costa e Wagner Cezarino também do IA-CTI , pelo auxílio no uso do SIPS e da linguagem C .

À todos os amigos que apesar de não serem nominalmente citados, sabem que grande parte deste trabalho não seria realizada sem o seu auxílio.

Finalmente , à Direção da Companhia Siderúrgica Nacional , à qual seremos eternos devedores, por nos possibilitar esta oportunidade de aperfeiçoamento profissional .

RESUMO

Neste trabalho são apresentados alguns modelos e ferramentas automatizadas para o gerenciamento de projetos ; seguindo o modelo clássico de gerenciamento de projetos, é proposto um modelo genérico de Gerenciamento de Projetos de Engenharia de Software criado através da união de dois conjuntos de informações .

O primeiro conjunto é formado por um sub-conjunto das informações existentes em um ambiente de engenharia de software , contendo os modelos de ciclos de vida , as metodologias e seus produtos associados , juntamente com as informações sobre os recursos materiais e humanos disponíveis para o desenvolvimento dos projetos de engenharia de software . Denominamos este primeiro conjunto de informações de "ambiente de gerenciamento" .

O segundo conjunto é constituído pelas informações pertinentes aos projetos de engenharia de software que serão gerenciados . Denominamos este segundo conjunto de "ambiente de projetos" .

Baseando-se nesse modelo proposto , é apresentado o protótipo de uma ferramenta que automatiza o gerenciamento dos projetos de engenharia de software .

ABSTRACT

A description of models of Project Management and of automated tools to support Project Management is presented . A software Engineering Project Management Model , based on the Classical Model of Project Management , is proposed . The model proposed is created through the union of two sets of information .

One set is that composed by information obtained from a typical Software Engineering Environment - life cycles , methodologies , products - plus information on resources available for development of Software Engineering Projects (human and material resources).

The other set is composed of information concerning the Software Engineering Projects themselves .

A prototype of an automated tool to support Software Engineering Project Management , according to the proposed model was implemented and is described .

ÍNDICE

1	- INTRODUÇÃO	1
2	- MODELOS DE GERENCIAMENTO DE PROJETOS DE ENG. DE SOFTWARE E FERRAMENTAS AUTOMATIZADAS EXISTENTES	5
3	- O MODELO CLÁSSICO DE GERENCIAMENTO DE PROJETOS	18
	3.1 - CONSIDERAÇÕES INICIAIS	18
	3.2 - ATIVIDADES PRINCIPAIS DE CADA FUNÇÃO	19
	3.3 - TAREFAS ASSOCIADAS COM CADA ATIVIDADE	22
	3.3.1 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DO PLANEJAMENTO	22
	3.3.2 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DA ORGANIZAÇÃO	30
	3.3.3 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DA ALOCAÇÃO DE PESSOAL	37
	3.3.4 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DO DIRECIONAMENTO	40
	3.3.5 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DO CONTROLE	42
4	- MODELO PROPOSTO PARA O GERENCIAMENTO DE PROJETOS DE ENGENHARIA DE SOFTWARE	47
	4.1 - INFORMAÇÕES PRINCIPAIS	47
	4.1.1 - O PROJETO	47
	4.1.2 - OS RECURSOS	49
	4.1.3 - OS PROCEDIMENTOS E METODOLOGIAS	50
	4.2 - O AMBIENTE DE GERENCIAMENTO	51
	4.3 - O DIAGRAMA DE ENTIDADES E RELACIONAMENTOS DO MODELO PROPOSTO	52

4.4	- O FUNCIONAMENTO DO MODELO PROPOSTO	53
4.4.1	- O PLANEJAMENTO DO PROJETO DE ENGENHARIA DE SOFTWARE SEGUNDO O MODELO PROPOSTO	55
4.4.1.1	- SUBDIVISÃO DO TRABALHO	55
4.4.1.2	- QUANTIFICAÇÃO DO TRABALHO	55
4.4.1.3	- SEQUENCIAMENTO DO TRABALHO	57
4.4.1.4	- ORÇAMENTOS	57
4.4.1.5	- CRONOGRAMAS	58
4.4.1.6	- MONTAGEM DA EQUIPE	58
4.4.2	- A EXECUÇÃO DO PROJETO DE ENGENHARIA DE SOFTWARE SEGUNDO O MODELO PROPOSTO	59
4.4.2.1	- APURAÇÃO DOS GASTOS REAIS	59
4.4.2.2	- MEDIÇÃO DO PROGRESSO	59
4.4.2.3	- AVALIAÇÃO DA PERFORMANCE	65
4.4.2.4	- MEDIÇÃO DA PRODUTIVIDADE	65
4.4.2.5	- CONTROLE DE MUDANÇAS	68
4.4.2.6	- ESTIMATIVAS	68
5	- IMPLEMENTAÇÃO DO PROTÓTIPO	70
5.1	- CONSIDERAÇÕES INICIAIS	70
5.2	- O SIPS	70
5.2.1	- CONSIDERAÇÕES INICIAIS	70
5.2.2	- ARQUITETURA DO SIPS	71
5.2.3	- CLASSES DO SIPS	74
5.2.4	- O MRO	77
5.2.5	- LINGUAGEM DE DESCRIÇÃO DA BASE META	79
5.2.6	- FORMULÁRIOS PARA ENTRADA DE DADOS	81
5.3	- CLASSES DO PROTÓTIPO	87
5.4	- MRO DO PROTÓTIPO	89
5.5	- DESCRIÇÃO DA BASE META DO PROTÓTIPO	90
5.6	- FORMULARIOS UTILIZADOS NO PROTÓTIPO	91
5.7	- ARQUITETURA DO PROTÓTIPO	94
5.8	- ESPECIFICAÇÃO DOS MÓDULOS	95
5.9	- ESPECIFICAÇÃO DAS ROTINAS DE USO GERAL	103

6	- UTILIZAÇÃO DO PROTÓTIPO	105
6.1	- CONFIGURAÇÃO DO AMBIENTE DE GERENCIAMENTO	106
6.2	- GERENCIAMENTO DE PROJETOS	109
6.2.1	- O PLANEJAMENTO DO PROJETO	111
6.2.2	- A EXECUÇÃO DO PROJETO	117
7	- CONCLUSÕES	123
8	- BIBLIOGRAFIA	126
ANEXO 1 - LINGUAGEM DE DESCRIÇÃO DO MODELO PROPOSTO		129
ANEXO 2 - LISTAGEM DOS FORMULARIOS		143

LISTA DE FIGURAS

FIGURA 1	- FUNÇÕES PRINCIPAIS DO GERENCIAMENTO DE PROJETOS ..	5
FIGURA 2	- ÍCONES DO MODELO DE LIU	11
FIGURA 3	- EXEMPLO DE UM DESIGNNET	11
FIGURA 4	- ORGANIZAÇÃO FUNCIONAL	31
FIGURA 5	- ORGANIZAÇÃO POR PROJETO	31
FIGURA 6	- ORGANIZAÇÃO MATRICIAL	32
FIGURA 7	- O PROJETO	48
FIGURA 8	- OS RECURSOS	49
FIGURA 9	- OS PROCEDIMENTOS E METODOLOGIAS	50
FIGURA 10	- AMBIENTE DE GERENCIAMENTO	51
FIGURA 11	- DIAGRAMA DE ENTIDADES E RELACIONAMENTOS DO MODELO PROPOSTO	52
FIGURA 12	- RELACIONAMENTO ENTRE AS PARTES PRINCIPAIS DO GEPES	53
FIGURA 13	- DISTRIBUIÇÃO DO ESFORÇO	61
FIGURA 14	- CURVA "PERCENTAGEM COMPLETADA X (RECURSOS*UNIDADES)	62
FIGURA 15	- ARQUITETURA DO SIPS	73
FIGURA 16	- EXEMPLO DE CLASSES DO SIPS	76
FIGURA 17	- EXEMPLO DE MRO	78
FIGURA 18	- MODELO DE UM FORMULÁRIO	83
FIGURA 19	- LISTAGEM DE DESCRIÇÃO DO FORMULÁRIO	84
FIGURA 20	- CLASSES DO PROTÓTIPO	88
FIGURA 21	- MRO DO PROTÓTIPO	89
FIGURA 22	- FORMULÁRIO AMBIENTE-DE-GERENCIAMENTO	92
FIGURA 23	- FORMULÁRIO CATEGORIAS-RECURSOS-HUMANOS	92
FIGURA 24	- FORMULÁRIO TIPO-DE-CICLO-DE-VIDA	93
FIGURA 25	- FORMULÁRIO FASES	93
FIGURA 26	- ARQUITETURA DO PROTÓTIPO	94
FIGURA 27	- CRIAÇÃO DE UM AMBIENTE DE GERENCIAMENTO	106

FIGURA 28 - FORNECIMENTO DE DADOS DO AMBIENTE DE GERENCIAMENTO	107
FIGURA 29 - CARACTERIZAÇÃO DE PARTE DO CICLO DE VIDA CLÁSSICO	109
FIGURA 30 - ATIVAR A FERRAMENTA PARA O GEPES	110
FIGURA 31 - INCLUSÃO DE UM NOVO PROJETO	111
FIGURA 32 - PLANEJAMENTO DO PROJETO	112
FIGURA 33 - A SUBDIVISÃO DO TRABALHO	112
FIGURA 34 - SELECIONAR UMA ATIVIDADE DO PROJETO	114
FIGURA 35 - QUANTIFICAR ATIVIDADES DO PROJETO	115
FIGURA 36 - SEQUENCIAMENTO DAS ATIVIDADES DO PROJETO	116
FIGURA 37 - A EXECUÇÃO DO PROJETO	117
FIGURA 38 - APURAR GASTOS REAIS	118
FIGURA 39 - FIM DA APURAÇÃO DE TODAS ATIVIDADES	119
FIGURA 40 - MEDIÇÃO DO PROGRESSO	120
FIGURA 41 - MEDIÇÃO DA PRODUTIVIDADE	121
FIGURA 42 - AVALIAÇÃO DA PERFORMANCE	122

LISTA DE TABELAS

TABELA 1	- OBJETOS UTILIZADOS NO BDPP	9
TABELA 2	- FERRAMENTAS AUTOMATIZADAS EXISTENTES	13
TABELA 3	- FUNÇÕES PRINCIPAIS DO MODELO CLÁSSICO DE GERENCIAMENTO DE PROJETOS	18
TABELA 4	- ATIVIDADES PRINCIPAIS DO PLANEJAMENTO DE PROJETOS DE ENGENHARIA DE SOFTWARE	19
TABELA 5	- ATIVIDADES PRINCIPAIS DA ORGANIZAÇÃO DE PROJETOS DE ENGENHARIA DE SOFTWARE	20
TABELA 6	- ATIVIDADES PRINCIPAIS DA ALOCAÇÃO DE PESSOAL EM PROJETOS DE ENGENHARIA DE SOFTWARE	20
TABELA 7	- ATIVIDADES PRINCIPAIS DO DIRECIONAMENTO DE PROJETOS DE ENGENHARIA DE SOFTWARE	21
TABELA 8	- ATIVIDADES PRINCIPAIS DO CONTROLE DE PROJETOS DE ENGENHARIA DE SOFTWARE	21
TABELA 9	- COMPARAÇÃO DAS FERRAMENTAS UTILIZADAS PARA O PLANEJAMENTO DO PROJETO DE ENGENHARIA DE SOFTWARE.	27
TABELA 10	- PONTOS FORTES E FRACOS DE CADA ESTRUTURA ORGANIZACIONAL	33
TABELA 11	- PONTOS FORTES E FRACOS DE CADA TIPO DE ORGANIZAÇÃO DE EQUIPE	34

CAPÍTULO 1 - INTRODUÇÃO

Com o surgimento da " Crise de Software " , citada em toda e qualquer literatura sobre engenharia de software (inclusive nesta),houve um esforço nos anos 70 , no sentido de desenvolverem-se novas técnicas (projeto top-down , modularidade, ocultamento de informação) , novos métodos técnicos e gerenciais (método de Jackson , metodologias de especificação de requisitos , projeto e análise estruturados , organização do processo de desenvolvimento do software em fases de um ciclo de vida , aplicação de técnicas de PERT/CPM) e ferramentas automatizadas (gerenciadores de configuração , analisadores de cobertura de testes , analisadores de fluxogramas , geradores de código , ambientes de desenvolvimento) que melhorassem sensivelmente a tecnologia de software disponível para utilização em todas as fases do ciclo de vida do projeto de software [PEN88] .

Entretanto, a maioria dos componentes deste ferramental é, até hoje, do tipo STAND-ALONE , automatizando tarefas específicas do desenvolvimento do software , sem uma integração efetiva entre os componentes .

Existem esforços no sentido de criarem-se ambientes de Engenharia de Software automatizados que , integrando todas estas ferramentas, tirem vantagem do sinergismo da combinação destas ferramentas . Alguns exemplos locais seriam o SIPS do CTI , o MOSAICO da PUC , o PC-CASE do IBPI dentre outros , que visam produzir software dentro de prazos e custos pré-estabelecidos e que atenda à especificação de requisitos do usuário .

Apesar de todo o ferramental disponível , os projetos de software continuam sendo entregues muito além da data prevista , com um custo muito maior do que o estimado e ,na maioria das vezes, não atendendo os requisitos do usuário.

Por que ? A resposta não é trivial . Existem incontáveis fatores que levam ao insucesso de um projeto . Um destes fatores , que talvez se inclua dentre os mais importantes é o *GERENCIAMENTO DE PROJETOS* .

Ainda que existam projetos realizados com sucesso , sem aplicação das técnicas de gerenciamento de projetos , acreditamos que , para projetos de médio a grande porte , o bom gerenciamento de projetos é um dos fatores primordiais do sucesso destes projetos.

Para projetos pequenos , o gerenciamento de projetos pode não ser imprescindível ; entretanto, com sua aplicação estaremos eliminando uma das possíveis causas de fracasso , o que por si só justifica sua aplicação em todos os tipos de projetos .

Apesar da importancia do Gerenciamento de Projetos de Engenharia de Software , que referenciaremos como GEPES , não existe ainda nenhum modelo que capture todas as informações envolvidas no GEPES. Existe apenas o consenso de que todas as informações pertinentes ao GEPES devem estar integradas em um único banco de dados .

Além de não existir um modelo específico para o GEPES , as ferramentas automatizadas existentes para uso comercial , restringem-se praticamente ao cálculo de redes de precedência (PERT , CPM) , que não deixam de ser muito úteis na fase de planejamento do projeto, quando se está fazendo a subdivisão do projeto em atividades , e quando se estão alocando recursos materiais e humanos a estas atividades . Termina aqui, porém, a utilidade destas ferramentas , pois as mesmas não levam em consideração informações imprescindíveis para o GEPES , como por exemplo o tipo de ciclo de vida adotado para o projeto , as metodologias utilizadas , os produtos gerados , etc. As ferramentas automatizadas são ainda muito falhas no que concerne à execução do projeto , isto é , o acompanhamento do projeto , não informando ao gerente do projeto , de agora em diante denominado GP , o desempenho nem a produtividade do pessoal ou dos recursos alocados ao projeto .

Com a medição do progresso , que consiste em determinar a percentagem completada de cada atividade ou fase do projeto de engenharia de software , podemos avaliar se o projeto está progredindo ou não conforme o planejado . Algumas ferramentas existentes não calculam a percentagem completada , e as que o fazem baseiam-se no número de unidades produzidas em cada atividade , utilizando a fórmula :

$$\text{percentagem completada} = 100 \times \frac{\text{número unidades produzidas}}{\text{número unidades previstas}}$$

No capítulo 4 mostraremos que existem formas mais corretas para calcular o progresso do que esta utilizada pelas ferramentas existentes .

A proposta deste trabalho é apresentar um modelo genérico e o protótipo de uma ferramenta automatizada para o GEPES , que leve em conta as particularidades da Engenharia de Software .

Para levarmos em conta as particularidades da Engenharia de Software , o modelo possui as informações dos tipos de ciclos de vida adotados , metodologias utilizadas e produtos gerados por cada metodologia .

Outro tipo de informações necessárias para gerenciarmos projetos são os recursos humanos e materiais disponíveis para a execução dos projetos de engenharia de software .

Estas informações sobre Engenharia de Software e recursos disponíveis foram agrupadas em um conjunto que denominamos "ambiente de gerenciamento", cuja existência é uma das características principais do modelo proposto.

Além do ambiente de gerenciamento, necessitamos das informações sobre os projetos de engenharia de software que serão executados. As informações sobre os projetos estão reunidas no que denominamos "ambiente de projetos".

Portanto, o modelo genérico proposto está subdividido em duas partes principais, que foram devidamente integradas entre si, que são o ambiente de gerenciamento e o ambiente de projetos de engenharia de software.

O modelo é genérico porque praticamente qualquer tipo de ciclo de vida pode ser decomposto em fases, subfases e tipos de atividades de desenvolvimento de software com suas metodologias e produtos específicos. Esta decomposição do modelo de desenvolvimento é diretamente mapeada no ambiente de gerenciamento, permitindo representar qualquer modelo de desenvolvimento para o software. Aliando-se a isto o fato de que o modelo clássico de gerenciamento de projetos adotado neste trabalho também é de uso genérico, podemos afirmar que tanto o modelo quanto a ferramenta implementada para o GEPES são genéricos, servindo para qualquer tipo de projeto de engenharia de software.

O protótipo foi implementado no SIPS (Sistema Integrado para Produção de Software), em desenvolvimento no IA-CTI (Instituto de Automação - Centro Tecnológico para Informática) de Campinas SP. A opção de implementação no SIPS deveu-se ao fato de o mesmo ser um gerenciador de bancos de dados de uso geral, baseado no MRO - Modelo de Representação de Objetos [TRAB6] [TRAB8], e que oferece recursos poderosos para este tipo de trabalho, como ficará claro na descrição do mesmo, apresentada no capítulo 5.

Um modelo para o GEPES, proposto por LIU [LIU89], o qual detalharemos mais à frente neste trabalho, parece ser inovador e possuir a maioria das informações necessárias para o GEPES. Possui, entretanto, a falha clássica de não estar associado com os procedimentos de engenharia de software, servindo para o gerenciamento de projetos de engenharia de software, mas não sendo específico para este fim. Nossa ferramenta também pode ser considerada de uso geral, porém com a possibilidade de configurarmos o ambiente de gerenciamento segundo os procedimentos da engenharia de software (ou quaisquer outros tipos de procedimentos) particularizando-a para este uso específico.

A seguir apresentamos a organização deste trabalho .

No Capítulo 2 apresentamos os modelos de gerenciamento de projetos existentes e sua aplicabilidade aos projetos de engenharia de software.

No Capítulo 2 temos também uma descrição das ferramentas automatizadas existentes e uma pequena análise de cada ferramenta. Estas ferramentas são de uso genérico para o gerenciamento de projetos, já que não encontramos nenhuma específica para o GEPES .

No Capítulo 3 detalhamos o modelo clássico de gerenciamento de projetos , modelo adotado em nossa ferramenta .

No Capítulo 4 apresentamos uma descrição do modelo proposto, baseado no modelo clássico de gerenciamento de projetos, porém considerando as particularidades da engenharia de software .

No Capítulo 5 descrevemos a implementação do protótipo da ferramenta automatizada .

Finalmente, no Capítulo 6 , através de um exemplo de utilização da ferramenta, descrevemos algumas situações com as quais defronta-se o GP de Engenharia de Software, e como o mesmo poderá utilizar a ferramenta para auxiliá-lo nestas situações.

CAPÍTULO 2 - MODELOS DE GERENCIAMENTO DE PROJETOS DE ENGENHARIA DE SOFTWARE & FERRAMENTAS AUTOMATIZADAS EXISTENTES

2.1 - MODELOS DE GERENCIAMENTO DE PROJETOS DE ENGENHARIA DE SOFTWARE

O gerenciamento de projetos é definido como um conjunto de procedimentos, tecnologias e "know-how", que apóiam o planejamento, organização, alocação de pessoal, direcionamento e controle necessários para gerenciar com sucesso um projeto de engenharia. Know-how neste caso significa a habilidade, suporte e discernimento necessários para aplicar efetivamente o conhecimento. Se o produto do projeto for Software, o gerenciamento é chamado Gerenciamento do projeto de Desenvolvimento de Software ou mais recentemente GEPES.

O GEPES não é diferente do gerenciamento de qualquer outro projeto. As funções são as mesmas do modelo clássico de gerenciamento [K0084]: planejamento, organização, alocação de pessoal, direcionamento e controle (Fig. 1), apenas as atividades para implementar estas funções são diferentes.

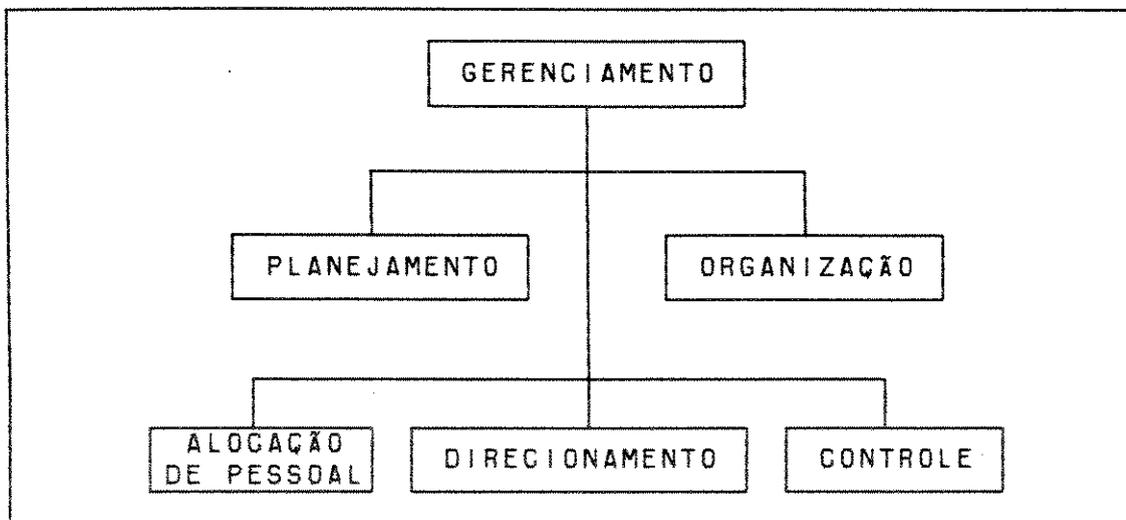


Fig. 1 - Funções principais do gerenciamento de projetos

Neste modelo, o gerenciamento é dividido em 5 funções ou componentes: *planejamento*, *organização*, *alocação de pessoal*, *direcionamento* e *controle*. Todas as atividades de gerenciamento tais como confecção de orçamentos, cronogramas, determinação de autoridade, treinamento, comunicação, acompanhamento e assim por diante, encaixam-se em uma destas 5 funções.

Isto não significa que as 5 funções são totalmente independentes. Existem interrelacionamentos entre elas, como: a) planejar para organizar uma atividade; b) controlar a função de planejamento etc. Entretanto, apesar destes interrelacionamentos, cada atividade de um gerente pode ser mapeada diretamente em uma e apenas uma das 5 funções de gerenciamento. Por exemplo, a) acima é uma função de planejamento, b) é uma função de controle.

A descrição do GEPES segundo o modelo clássico será feita no capítulo 3 por meio de uma abordagem TOP-DOWN, conforme mostrada em [THAB8], sendo grande parte das atividades também extraídas deste artigo.

Cada função do modelo clássico de gerenciamento será apresentada, sendo em seguida detalhadas suas atividades, e para cada atividade, suas tarefas relacionadas.

Além de descrevermos o modelo clássico do GEPES, no capítulo 3 informamos também as atividades do modelo clássico: que foram implementadas no modelo proposto; as que não foram implementadas embora existam informações suficientes no modelo proposto para implementá-las na ferramenta automatizada; e as atividades do modelo clássico que não podem ser executadas no modelo proposto.

As funções de Planejamento e Controle são as principais do GEPES, pois conforme [K0076]:

"O processo de gerenciamento é normalmente considerado como uma combinação de procedimentos diferentes, tais como estabelecer objetivos, planejar, organizar, integrar, coordenar, alocar pessoal, direcionar, supervisionar, motivar, avaliar desempenhos, controlar, etc. Do ponto de vista do sistema de software, planejamento e controle são as funções gerenciais básicas. O planejamento é a seleção dentre alternativas, de caminhos futuros de ação, e o controle é a medição e correção de desempenho de atividades, para garantir que os objetivos e planos gerenciais estão sendo atendidos."

Além de serem as mais importantes, as funções de Planejamento e Controle são as que mais prestam-se a implementação numa ferramenta automatizada, tendo a maioria de suas atividades sido implementadas no modelo proposto. As atividades do Planejamento, Controle, Organização, Alocação de Pessoal e Direcionamento que não foram implementadas no modelo proposto, serão grandemente auxiliadas pela informações contidas no ambiente de gerenciamento.

É importante ressaltar novamente, que o GEPES não é diferente do gerenciamento de qualquer outro projeto, e que todos os tipos de projetos são gerenciados segundo o modelo clássico de gerenciamento, mudando-se apenas os tipos de atividades necessárias para implementar o modelo. Este foi portanto o modelo adotado em nossa ferramenta, o qual está detalhado no Capítulo 4.

O modelo proposto , bem como algumas propostas existentes na literatura , e que apresentaremos a seguir , não são bem "modelos", mas sim diferentes formas de representar o modelo clássico de gerenciamento aplicado à Engenharia de Software.

A forma mais utilizada atualmente para representar o modelo clássico de gerenciamento aplicado à Engenharia de Software, consiste em associar as atividades do GEPES às atividades do desenvolvimento de Projetos de Engenharia de Software segundo o tradicional modelo em cascata de desenvolvimento de software.

Existem várias alternativas para o modelo de desenvolvimento em cascata , como por exemplo os tres modelos apresentados por W.W.Agresti [AGR86] em 1986, denominados : prototipagem , especificações operacionais e implementação transformacional . Mas como o modelo de gerenciamento proposto no capítulo 4 deste trabalho é genérico , o mesmo será utilizado qualquer que seja o modelo de desenvolvimento escolhido .

Encontramos na literatura mais recente , apenas uma proposta inovadora em termos de modelagem do processo de desenvolvimento e que engloba o gerenciamento de projetos de Engenharia de Software, apresentada por Lung-Chun Liu e Ellis Horowitz [LIU89] .

Ainda que inovadora na modelagem , a proposta de Liu obtém as mesmas informações do modelo clássico de gerenciamento . Antes porém de apresentarmos esta modelagem , comentaremos o artigo de Maria H.Penedo [PEN85] por concordarmos plenamente com a necessidade de um banco de dados integrado , que contenha todas as informações do projeto .

Este artigo preconiza a utilização de um Banco de Dados de Projetos Integrado , chamado BDPP - Banco de Dados Principal do Projeto (PMDB - Project Master Database) que armazene todos os dados pertinentes a um projeto : recursos , produtos , informações gerenciais , etc. , como uma forma de aumentar a produtividade do processo de desenvolvimento de software .

O BDPP é descrito como o núcleo de qualquer ambiente integrado de engenharia de software . Integrando-se as informações em um banco de dados , eliminam-se os problemas de duplicação de dados , inconsistência de dados e complexidade de acesso às informações , causados pela pequena conectividade dos dados , quando estes estão em sistemas diferentes .

O modelo do BDPP, que não é apresentado no artigo, leva em conta dois grupos principais de informações , que são os procedimentos e métodos associados com o BDPP , e as diferentes visões que os usuários devem ter do BDPP.

Os procedimentos e métodos associados com o BDPP incluem por exemplo : quando os componentes são criados , revisados e congelados ; quem pode criar , modificar , ler , executar e apagar componentes ; quais os mecanismos usados para controle de versões ; quais componentes são afetados por mudanças em outros componentes ; quem deve ser notificado sobre mudanças no banco de dados ; quais são os tipos de relatórios e formulários usados nas diferentes fases do ciclo de vida do projeto , etc.

As diferentes visões do usuário levam em conta que além do GP , também programadores , engenheiros de software , secretárias , etc terão acesso ao BDPP , e que cada indivíduo , de acordo com seu papel no projeto , manuseia um subconjunto das informações do BDPP. Além disso, a estrutura do BDPP deve suportar diferentes níveis de abstração , objetos compostos e granularidades , conforme o papel das pessoas no projeto .

Para descrever o BDPP , foi utilizado o modelo de entidades-relacionamentos [CHE76] . O modelo consiste de 31 objetos (nosso protótipo possui apenas 10) , aproximadamente 220 atributos e aproximadamente 170 relacionamentos . A Tabela 1 apresenta os 31 objetos utilizados no modelo , a fim de dar uma idéia das informações contidas no BDPP .

O protótipo da ferramenta automatizada para o GEPES foi implementado no SIPS (Sistema Integrado para Produção de Software) que também pode ser considerado como um Banco de Dados Principal de Projeto , formado por duas bases de dados , que são a Base Meta e a Base Sistema, cuja descrição é apresentada na seção 5.2 - O Sips .

TABELA 1 - OBJETOS UTILIZADOS NO BDPP

NOME	DESCRIÇÃO
TAREFA CONTÁBIL	As tarefas ou jobs a serem executados pelo projeto
ITEM DE MUDANÇA	Uma mudança que ocorre em um projeto, causada por solicitação do cliente, problemas ou outras razões
CONSUMÍVEL ADQUIRIDO	A compra de consumíveis utilizados pelo projeto
CONTRATO	Contrato ou subcontratos que são parte do projeto
COMPONENTE DE DADOS	Os itens de dados que são parte do desenvolvimento de software
DICIONÁRIO	O dicionário do projeto ,ie , termos e acrônimos
DOCUMENTO	Documentos do projeto ,ie , planos, especificações, manuais, análises de custos , layouts de hard, etc.
EQUIPAMENTO ADQUIRIDO	Compra de equipamentos (computadores, componentes , peças etc.)
COMPONENTE EXTERNO	Sistemas externos ao sistema sendo construído mas que interfacelam com este
ARQUITETURA DE HARDWARE	Configuração de hardware
COMPONENTE DE HARDWARE	Componentes de hardware
DESCRIÇÃO DE COMPONENTE DE HARDWARE	Características dos diferentes tipos de hardware , produzidos ou adquiridos
INTERFACE	Interfaces entre hard/software ou soft/software
MARCO	Cada um dos marcos principais do cronograma do projeto
CENÁRIO OPERACIONAL	Cenários que comprovam que o sistema funciona sob certos critérios de performance .
PESSOA	Pessoal do projeto
RELATÓRIO DE PROBLEMAS	Problemas que tenham sido relatados
PRODUTO	Versões de produtos entregues
DESCRIÇÃO DE PRODUTO	Características dos diferentes produtos
REQUISITO	Requisitos do projeto
RECURSO	Recursos do projeto (viagens, copias, etc)
RISCO	Elementos identificados como de risco para o projeto
SIMULAÇÃO	Rodadas de simulação das arquiteturas de hard/software
COMPONENTE DE SOFTWARE	Cada componente do software
CONFIGURAÇÃO DE SOFTWARE	Configurações de software
TAREFA EXECUTÁVEL DE SOFTWARE	O empacotamento de componentes de software
SOFTWARE ADQUIRIDO	Compra de componentes de software
CASO DE TESTE	Casos de testes
PROCEDIMENTO DE TESTE	Procedimentos que usam um ou mais casos de testes
FERRAMENTA	Ferramentas ou programas utilizados pelo projeto em seu ciclo de vida
ELEMENTO DE EST	Elementos da EST (WBS)

O modelo proposto por LIU chama-se DesignNet (manteremos o nome no original) . A motivação para criação do modelo , segundo os autores , foi a de que ainda não existe um modelo com suficiente poder de expressão para representação e monitoração dos componentes , relacionamentos , mecanismos e diferentes visões do processo de desenvolvimento e gerenciamento do software .

O próprio modelo em cascata , mais utilizado atualmente , apresenta setas apontando para frente , mas também para trás , reconhecendo que o modelo não é preciso , e que pode haver retorno para fases prévias . Existem várias outras críticas ao modelo em cascata e que aparecem em [COO83] e [CUR87] e que não citaremos aqui . Ainda dentro da motivação , Liu reconhece que não deverá existir um único modelo que descreva o processo de desenvolvimento de software sob todos os ângulos . A seguir ele lista 6 fatores que considera essenciais em qualquer modelo :

- 1 - O modelo deve descrever adequadamente o fato de que o desenvolvimento de software é um processo de criação . Criação é um processo inerentemente evolucionário , onde passos já executados são continuamente revisados e até mesmo abandonados , e onde novos passos podem ser inseridos .
- 2 - O modelo deve ser capaz de incluir o fato de que um projeto de software de grande escala é um processo paralelo , com muitas pessoas executando tarefas simultaneamente .
- 3 - O modelo deve ser capaz de indicar que um diferente conjunto de condições deve existir antes que uma atividade possa ser executada .
- 4 - O modelo deve indicar todos os artefatos que são produzidos nos vários pontos do processo .
- 5 - Se uma atividade falhar , o modelo deve ser capaz de indicar as atividades e recursos afetados . Atividades afetadas podem ter de ser reexecutadas .
- 6 - O modelo deve ser capaz de indicar a extensão e natureza dos recursos envolvidos numa tarefa , incluindo pessoas , recursos consumíveis e não consumíveis .

O DesignNet pretende satisfazer todos estes 6 critérios . É um modelo híbrido , que utiliza operadores AND/OR para descrever a EST - Estrutura de Subdivisão do Trabalho (WBS - Work Breakdown Structure) , e rede de Petri para representar as dependências entre atividades , recursos e produtos .

O DesignNet utiliza 5 ícones para representar os objetos do projeto (Fig.2) .

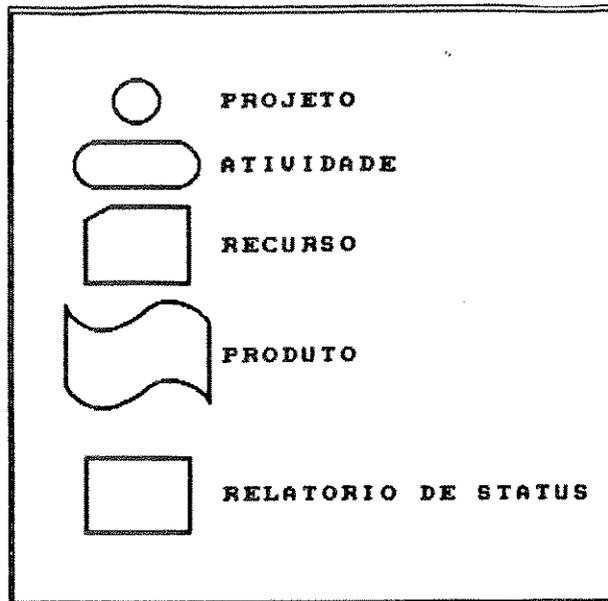


Fig. 2 - Ícones do modelo de Liu

O diagrama da Fig.3 informa-nos que a atividade "PROJETO" só pode ser iniciada (disparada) quando houver um CONTRATO, um USUÁRIO disponível para acompanhar o projeto e a ANÁLISE DO SISTEMA já esteja pronta. Informa-nos também, por exemplo, que a atividade INTERFACE é composta das duas atividades ENTRADA E SAIDA, e que a atividade ENTRADA é composta de uma das atividades COM MOUSE OU SEM MOUSE.

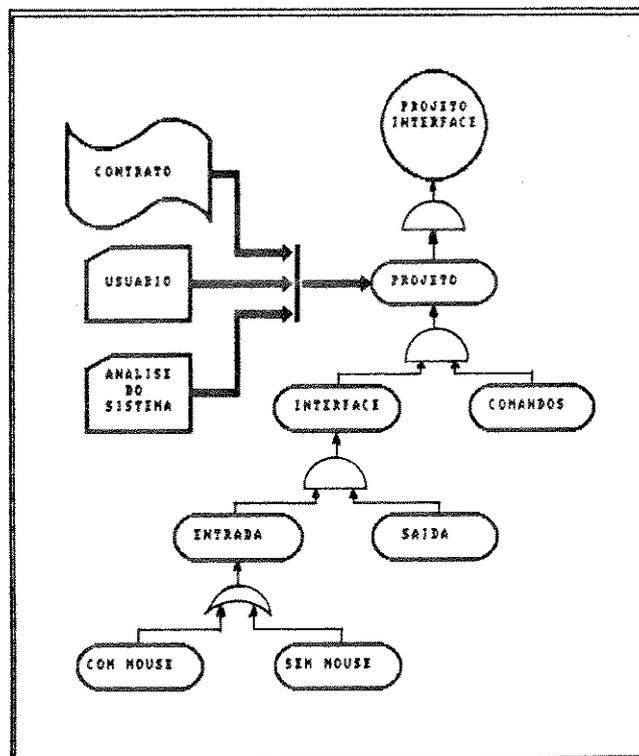


Fig. 3 - Exemplo de um DesignNet

Uma modificação interessante na rede de Petri , adotada no DesignNet é a de que o disparo de transições é um processo não volátil , ou seja , a cada disparo de uma transição o token não é removido e sim criada uma nova instancia do token com informações dependentes do tempo , o que possibilita saber-se por exemplo , quantas vezes uma atividade foi executada , ou fazer o controle de versões dos produtos .

Outra característica interessante da formalização do modelo do DesignNet , é a possibilidade de acessarmos propriedades importantes de um projeto bem gerenciado , que seriam a boa-execução , a conectividade , a completeza e a consistencia do projeto .

Um projeto "bem executado" é o que possui em qualquer instante, no máximo um token em estado ativo em qualquer lugar . Isto significa que as atividades não estão sendo reexecutadas , ou o que é pior, que estão sendo disparadas por duas ou mais transições diferentes, criando versões paralelas de um mesmo produto .

Um projeto é "conectado" se para qualquer lugar p , existe um caminho de p até a raiz do gráfico do projeto . Isto significa que todas as atividades visam um objetivo comum , e que nenhuma atividade ou produto irrelevante é criado .

Um projeto é "completo" se a sua DesignNet for conectada e se qualquer produto intermediário possuir um caminho que o leve a um produto final .

Um projeto é "consistente" se ele for completo e se o nível de cada atividade (distância até a raiz) for o mesmo de suas atividades predecessoras e sucessoras .

Ainda que bastante interessante , este modelo também não captura as informações específicas de engenharia de software.

O modelo de Liu não foi implementado porque o modelo clássico tem atualmente uma melhor aceitação pelos GP's. O modelo de Liu quando apresentado graficamente para projetos de médio a grande porte é muito difícil de ser manuseado. O modelo de gerenciamento proposto neste trabalho, além de capturar as informações específicas de engenharia de software possui , com exceção das informações sobre controle de versões (gerenciamento de configuração) , todas as informações do modelo de Liu , apenas sob uma forma diferente .

A seção a seguir apresenta uma análise das ferramentas automatizadas existentes a nível comercial para o gerenciamento de projetos .

2.2 - FERRAMENTAS AUTOMATIZADAS EXISTENTES

Em termos de automatização do gerenciamento, não conhecemos nenhuma ferramenta *específica* para o GEPES. [FAR89] apresenta um quadro comparativo de diversas ferramentas existentes no mercado, o qual reproduzimos parcialmente na Tabela 2.

Tabela 2 - Ferramentas Automatizadas Existentes

NOME\CHARACTER.	CLASSIFICACAO	DISCO RIGIDO/ ESPACO	TAREFAS NA REDE	NUMERO RECURSOS NO PROJETO	NUMERO RECURSOS POR TAREFA	INTER- DEPEN- DENCIAS /TAREFA	ACOMPAN- HAMENTO PROGRESSO	CONTROLE DE CUSTOS
ADVANCED PROJECT WORKBENCH	MEDIO	OP	500	200	1	12	SIM	SIM
FINEST	AVANÇADO	OB 3 Mb	10000	96	6	130	SIM	SIM
HORNET	AVANÇADO	OB 10 Mb	255	128	1	-	SIM	NAO
HIPM II 2.0	MEDIO	OP	S/LIMIT	S/LINITE	1	S/LIMIT	SIM	SIM
INSTAPLAN 1.03 B	BASICO	OP 1.2 Mb	1400	S/LINITE	1	S/LIMIT	OPCIONAL	OPCIONAL
MS-PROJECT 4.0	BASICO	OP	1000	255	-	16	SIM	SIM
MICROTRACK 1.6	MEDIO	OP 400 Kb	5000	S/LINITE	1	100	SIM	SIM
PERTMASTER ADVANCED 2.1	AVANÇADO	OP 2.1 Mb	1800	S/LINITE	S/LINITE	1800	SIM	SIM
PLANTRAC	AVANÇADO	OB 4.3 Mb	2500	200	-	-	SIM	OPCIONAL
* 8.1	AVANÇADO	OB 1.0 Mb	3000	192	1	1500	SIM	OPCIONAL
PRIMAVERA PROJECT PLANNER	AVANÇADO	OB 3 Mb	10000	S/LINITE	1	130	SIM	SIM
PROJECT SCHEDULER NETWORK	MEDIO	OP	2000	S/LINITE	-	-	SIM	SIM
PROJECT : VISION	BASICO	OP	500	100	1	12	SIM	SIM
PRO-PATH PLUS	BASICO	OP	500	60	1	10	SIM	SIM
QUICKNET PROFESSIONAL	AVANÇADO	OB	S/LIMIT	200	-	-	SIM	SIM
SSP'S PROMIS	AVANÇADO	OP 2.5 Mb	S/LIMIT	S/LINITE	-	-	SIM	SIM
SUPERPROJECT EXPERT 1.0	AVANÇADO	OP 1.4 Mb	1560	1500	1	S/LIMIT	SIM	SIM
TIMELINE 3.0	MEDIO	OB 1.9 Mb	1000	300	1	S/LIMIT	SIM	SIM
TIMEPIECE 1.3	BASICO	OB 0.7 Mb	500	S/LINITE	2	S/LIMIT	SIM	SIM
TOPDOWN PROJECT PLANNER 1.0	BASICO	OP 0.6 Mb	1500	3000	1	S/LIMIT	NAO	NAO
VIEWPOINT 3.0	AVANÇADO	OB 0.8 Mb	S/LIMIT	S/LINITE	10	-	SIM	SIM

OP OPCIONAL
OB OBRIGATORIO

A classificação (Basico, Medio e Avançado) apresentada na Tab 2, levou em conta vários fatores ponderados pelo autor, tais como: facilidade de utilização, rapidez, recursos disponíveis, etc.

A seguir apresentamos uma análise sucinta de cada uma das ferramentas. Estas análises foram extraídas dos artigos [FER87] e [W0088], pois só tivemos oportunidade de avaliar duas destas ferramentas, a Microsoft Project 4.0 e a Superproject Expert 1.0.

A **ADVANCED PROJECT WORKBENCH** foi projetada para empreendimentos que utilizam um recurso único . O seu gráfico de Gantt alinha-se entre os mais elegantes e sua interface não é muito lenta. O nivelamento dos recursos tem alguns defeitos, como não manipular as atividades que estão no caminho crítico e não resolver sobredemandas de recursos. No entanto oferece a função SHIFT que permite ao usuário mover a atividade em tela, enquanto observa a totalização da demanda de recursos, permitindo o nivelamento manual .

A **HARVARD TOTAL PROJECT MANAGER** faz uso dos recursos gráficos e das cores de modo soberbo, com excelentes resultados em tela. O gráfico de Gantt é um dos mais legíveis. Seu principal defeito é obrigar o usuário a criar um milestone sempre que duas atividades precedem ou sucedem uma outra. Embora a interface com o usuário seja razoavelmente simples , o programa apresenta vez por outra alguma dificuldade para reorganizar a rede, de vez que ele escolhe o traçado automaticamente e não dá ao usuário a chance de fazer alterações. Quando o usuário está planejando a tela, como é natural em quem usa um PC, o programa pode escolher alguns traçados bastante estranhos .

A **INSTAPLAN** possui interface com o usuário que obedece ao padrão Lotus 1-2-3 . A ferramenta oferece 3 telas básicas: o Gantt , o PERT e os histogramas de demanda dos recursos. O PERT segue a notação PDM (Precedence Diagram Method) e é desenhado na vertical. Apenas as antecessoras e sucessoras da tarefa assinalada são mostradas na tela de cada vez. Após terem sido declaradas as tarefas e os recursos , a ferramenta monta uma planilha cruzando estes dados. Na interseção de cada recurso (coluna) com cada tarefa (linha) aparece a demanda do recurso por aquela tarefa. A função de nivelamento automático de recursos só tem a opção de atrasar as atividades para não exceder os recursos disponíveis. A ferramenta permite alocação parcial dos recursos .

A **MS-PROJECT** é uma ferramenta boa, direta, sem muita elegancia mas fácil de usar e útil para projetos de pequeno e médio portes . A tela básica para entrada de dados é o gráfico de Gantt. Quando a função EDIT é selecionada, um campo se abre embaixo da tela para entrada dos dados em qualquer sequencia. Um conjunto de relatórios pode ser formatado pelo usuário com alguma liberdade. A falta de nivelamento automático de recursos limita a sua qualidade.

A **MICROTRAK** possui interface que parece ter sido desenvolvida para main-frames. O usuário dá entrada nos dados através do preenchimento de lacunas em um registro em tela. Ao final de cada tela pode optar por editar alguma das lacunas ou passar para a próxima tela. A interface com o usuário não é gráfica. Os cronogramas e a rede de PERT são editados através de símbolos e letras. A impressão gráfica só é possível através de um módulo de extensão. Outros pontos fracos são a ausência de nivelamento dos recursos e a limitação de apenas um calendário por projeto.

A **PERTMASTER ADVANCED** é uma ferramenta avançada, feita para quem pretende usar tudo que a tecnologia oferece. A interface tipo Lotus 1-2-3 é extremamente simples de aprender e usar. Os comandos e o help são descomplicados e sensíveis ao contexto. O excelente tutorial que acompanha o pacote leva o usuário a percorrer todas as funções em menos de duas horas. A importação de dados é das mais perfeitas do mercado, parecendo que a ferramenta foi feita para isso. O manual do usuário traz o formato detalhado de cada arquivo, de modo a facilitar a importação/exportação de/para outros sistemas. O programa administra o compartilhamento de recursos entre diversos projetos. O tratamento hierárquico dado aos recursos também é único: cada recurso pode ser composto por sub-recursos, que podem por sua vez serem compostos também por outros sub-recursos e assim sucessivamente. As telas são montadas pelo usuário com todos os graus de liberdade possíveis. O acompanhamento é bastante sofisticado, permitindo que se armazenem tantas versões anteriores do plano ou linha base do projeto (orçamentos e cronogramas) quantas se queiram para posterior comparação entre versões. O programa a cada atualização numera automaticamente a nova versão.

A **PLANTRAC** não pode ser considerada uma ferramenta típica de gerenciamento. Oferece quase tudo que se pode esperar em termos de relatórios e gráficos, mas seu layout de tela, principalmente dos gráficos não é dos melhores. Apenas 5 recursos podem ser alocados por tarefa.

A **PMS** ainda não oferece a função de nivelamento automático de recursos. Seu ponto forte é o gerador de relatórios em linguagem de alto nível, altamente sofisticada e bastante útil para o usuário exigente.

A **PRIMAVERA PROJECT PLANNER** é a mais poderosa ferramenta disponível no mercado, em termos de capacidade e velocidade. É também uma das mais difíceis de se aprender a usar em todo seu potencial. Para quem pretende utilizá-la intensivamente, no entanto, suas facilidades de geração de relatórios são das melhores do mercado. A ferramenta trabalha com dicionários para códigos de atividades, recursos e custos. Para utilizar um recurso em uma atividade por exemplo, ele deve ser cadastrado no dicionário de recursos, o que obriga o usuário a planejar tudo antes de sentar em frente ao micro.

A **PROJECT:VISION** disputa a faixa mais baixa do mercado, tendo como fator diferencial a abordagem de planilha de dados que utiliza. As atividades são colocadas em células de uma matriz e interligadas por linhas que trafegam entre as diversas células. O efeito visual é de um diagrama de blocos ou um organograma. Para aumentar a semelhança com uma planilha eletrônica, a tela é numerada no alto e tem letras nas laterais.

A **QUICKNET PROFESSIONAL** foi desenvolvida para ser usada por especialistas. Repleta de funções com alta capacidade, esta ferramenta não ganha prêmio de interface amigável. Sua ênfase está colocada em relatórios e tabelas em vez de gráficos. A entrada de dados requer um passeio através das mais diversas máscaras de tela. Permite que se programem diversos projetos em paralelo, utilizando uma árvore de prioridades para arbitrar a alocação dos recursos comuns.

A **SSP'S PROMIS** é uma ferramenta de destaque para montar e acompanhar um cronograma de projeto. Fácil de usar, embora se tenha de pular de tela em tela para conseguir qualquer coisa, seu ponto fraco é a baixa qualidade dos gráficos.

A **SUPERPROJECT EXPERT** é a mais conhecida ferramenta de gerenciamento e também uma das mais completas. Uma das poucas a oferecer o método PERT original de determinação probabilística da duração das atividades e talvez a única a permitir mostrar estas informações no gráfico de Gantt. A ferramenta permite a adoção de um calendário para cada recurso, em separado do calendário do projeto. Sua interface é das mais poderosas do mercado, ficando os pontos fracos por conta da lentidão de cálculo e da capacidade relativamente pequena.

A **TIMELINE** é uma líder na sua categoria, com excelente performance, flexibilidade e simplicidade de uso. A versão 3.0 acrescenta à anterior um gráfico de rede bem desenhado, o que era antes talvez seu único ponto fraco.

A **TIMEPIECE** se coloca na faixa baixa de preços e oferece funções não muito sofisticadas, compatíveis com seu baixo preço. A interface com o usuário obedece a menus, que variam em função do contexto. Valores default aparecem sempre que o usuário tem que fazer uma declaração.

A **TOPDOWN PROJECT PLANNER** requer um monitor colorido para operar. O nome não poderia ser mais sugestivo, já que ela é estruturada em torno de uma árvore hierárquica de redes PERT. O usuário define uma rede de nível 1 criando e conectando blocos. Isso feito, cada bloco pode ser definido como uma rede em si, como subprojetos, e assim indefinidamente. Ao observar a rede pode-se optar por um modo totalmente gráfico, onde, em vez de texto nos blocos, vê-se o desenho das redes que eles contêm. O terceiro nível neste modo já é indistinguível. A ferramenta dispõe de nivelamento automático de recursos, mas o próprio fabricante adverte que deve ser usado com cautela, pois pode espalhar as atividades fora do caminho crítico em alguns casos. Pode-se atribuir um calendário a cada recurso. O ponto fraco da ferramenta é limitar cada PERT a apenas uma tela.

A **VIEWPOINT** possui interface tipo Lotus 1-2-3 , o que faz dela uma ferramenta fácil de operar, com um cursor que se move na tela sob comando de um dispositivo mouse e que serve para atribuir datas e relações de precedência. A tela de fundo para entrada de dados é o PERT . A cada definição que deve ser feita, a ferramenta oferece um registro pré-formatado com lacunas para preencher. A medida que os dados vão sendo declarados, a ferramenta recalcula a rede e eventualmente altera a posição dos blocos para torná-la mais agradável. Como as demais ferramentas, permite que se armazene o plano original para análise de variação no futuro.

Para um gerente que não dispõe de nenhum recurso automatizado , qualquer uma destas ferramentas já será um grande avanço . Entretanto o ponto forte de todas estas ferramentas está na fase de **PLANEJAMENTO** do projeto , contemplando muito pouco do acompanhamento da **EXECUÇÃO** do projeto .

O acompanhamento só é feito em termos do progresso , sendo a percentagem completada , ou o número de unidades produzidas , diretamente informada pelo usuário da ferramenta o que em termos de engenharia de software é uma medida extremamente grosseira , pois para uma previsão de 1000 linhas de código , 500 linhas já produzidas não significam necessariamente 50% de progresso . Além disso , não existe forma de se obter a produtividade e a performance dos recursos envolvidos no projeto . As ferramentas permitem apenas que sejam guardadas versões da linha base do projeto , para posteriores comparações entre diferentes versões .

Em contra partida, nossa ferramenta permitirá as avaliações automáticas do progresso, produtividade e performance, além de ser uma ferramenta que pode ser configurada através de seu ambiente de gerenciamento, para uso específico no GEPES.

Tivemos a oportunidade de utilizar duas das ferramentas apresentadas no quadro anterior : MS-Project 4.0 e SuperProject Expert 1.0 . Ambas são equivalentes , já que o que uma oferece a mais de recursos (Expert) é compensada pela facilidade do interface homem/máquina da outra (MS Project) . Ambas as ferramentas , assim como a maioria das outras do quadro , não procedem à alocação automática dos recursos . Porém avisam ao gerente quais os recursos utilizados além de sua capacidade , auxiliando bastante a etapa de nivelamento dos recursos durante a fase de planejamento do projeto . Ambas permitem ainda que projetos sejam divididos em subprojetos , ou vice-versa , ou seja, que projetos sejam agregados em um super-projeto , o que em alguns casos também é interessante para o GP , quando por exemplo os recursos disponíveis têm que ser alocados a vários projetos .

O próximo capítulo apresenta de forma bastante detalhada o modelo clássico do gerenciamento de projetos .

CAPÍTULO 3 - O MODELO CLÁSSICO DE GERENCIAMENTO

3.1 - Considerações iniciais

O modelo clássico de gerenciamento de projetos é composto por 5 funções principais, conforme apresentado na tabela 3.

Tabela 3 - Funções Principais do Modelo Clássico de Gerenciamento

Planejamento	- Predeterminar a sequencia de ações para atingir os objetivos organizacionais .
Organização	- Ordenar o trabalho necessário para atingir os objetivos e delegar autoridade e responsabilidades para obter estes objetivos.
Alocação de Pessoal	- Selecionar e treinar pessoas para as posições da organização .
Direcionamento	- Criar uma atmosfera que ajude e motive as pessoas a alcançar os resultados finais.
Controle	- Medir e corrigir o desempenho das atividades , visando objetivos de acordo com os planos .

Estas funções são as mesmas para qualquer tipo de projeto . Apenas as atividades para implementá-las é que diferenciam um tipo de gerenciamento de projetos de outros. Assim sendo, em cada uma destas funções principais, consideramos as atividades inerentes ao gerenciamento de projetos de engenharia de software, apresentando portanto neste capítulo , o Modelo Clássico do Gerenciamento de Projetos de Engenharia de Software, que foi o modelo implementado no protótipo.

Este capítulo está subdividido em duas seções principais : a seção 3.2 - Atividades Principais de cada função do GEPES, onde apresentamos em forma de tabelas as atividades principais de cada uma das 5 funções principais do modelo clássico do GEPES ; a seção 3.3 - Tarefas associadas com cada atividade , lista as tarefas relacionadas com cada uma das atividades principais de cada função. Além de listarmos as tarefas, incluímos um resumo do estado da arte de cada tarefa; por exemplo, na tarefa de "organização da equipe", citamos os tipos de organização existentes e os seus pontos fortes e fracos . Nesta seção informamos também para cada atividade : se a mesma foi contemplada no modelo proposto ; se é possível implementá-la usando o modelo proposto ; e se a mesma não é exequível usando o modelo proposto.

O fato de uma atividade não ser exequível utilizando o modelo proposto significa , apenas , que ela não foi e nem temos dados suficientes para implementá-la com o modelo proposto . Porém *TODAS* atividades , implementadas ou não , devem ser executadas pelo GP.

3.2 - Atividades principais de cada função do GEPES

Tabela 4 - Atividades Principais do Planejamento	
ATIVIDADE	DEFINIÇÃO
Determinar objetivos ou metas.	Determinar objetivos , atividades , produtos ou resultados.
Desenvolver estratégias.	Decidir as metas principais da organização e o programa geral de ações para atingir estas metas.
Desenvolver políticas .	Tomar decisões firmes em assuntos importantes e repetitivos , para criar um guia para tomada de decisões.
Determinar cursos de ações.	Procurar e examinar diferentes cursos de ação.
Tomar decisões .	Avaliar e selecionar um rumo de ações dentre as alternativas disponíveis.
Determinar procedimentos e regras	Estabelecer métodos , diretrizes e limites para realizar uma atividade.
Desenvolver programas	Estabelecer políticas , procedimentos , regras , tarefas , cronogramas e recursos necessários para alcançar uma meta ou cumprir um curso de ação .
Prever situações futuras	Antecipar acontecimentos futuros ou tecer hipóteses sobre o futuro ; predizer resultados futuros ou expectativas dos cursos de ação.
Preparar orçamentos	Alocar custos e/ou numeros aos planos/programas.
Documentar os planos do projeto	Registrar decisões , cursos de ação , programas/planos, situações futuras prováveis , orçamentos , etc .

Tabela 5 - Atividades Principais da Organização

ATIVIDADE	DEFINIÇÃO
Identificar e agrupar as tarefas necessárias	Definir , estimar tamanho e agrupar as tarefas a serem executadas.
Selecionar e estabelecer a estrutura organizacional	Definir ou selecionar as estruturas necessárias para atender as tarefas e obrigações , e para controlar , coordenar e comunicar entre estas tarefas ou obrigações.
Criar posições na organização	Estabelecer títulos , obrigações , objetivos , escopo e relacionamento para cada posição.
Definir responsabilidades e autoridade	Definir as responsabilidades para cada posição , e a autoridade permitida para cumprir com as responsabilidades.
Estabelecer a qualificação das posições	Definir as qualificações das pessoas que preencherão cada posição.
Documentar a estrutura organizacional	Registrar a estrutura organizacional , definições de autoridade e responsabilidades , descrição das posições , etc .

Tabela 6 - Atividades Principais da Alocação de Pessoal

ATIVIDADE	DEFINIÇÃO
Preencher as posições organizacionais	Selecionar , recrutar ou promover pessoal qualificado para cada posição.
Assimilar o pessoal novo	Orientar e familiarizar os novos empregados com a organização , recursos e tarefas a serem executadas.
Educar ou treinar	Eliminar deficiências na qualificação através de treinamento e educação.
Fornecer condições para o desenvolvimento	Melhorar o conhecimento , atitudes e habilidades.
Avaliar o pessoal	Avalliar e registrar a qualidade e a quantidade de trabalho realizado por cada empregado.
Compensar	Dar bonus , benefícios , licenças ou remuneração financeira.
Terminar a alocação	Transferir ou separar pessoal redundante.
Documentar as decisões de pessoal	Registrar as decisões de alocação pessoal requisitos e planos de treinamento , etc.

Tabela 7 - Atividades Principais do Direcionamento

ATIVIDADES	DEFINIÇÃO
Suprir liderança	Induzir os subordinados a cumprirem suas tarefas com entusiasmo e confiança.
Supervisionar o pessoal	Dar ao pessoal instruções diárias, orientação e disciplina para cumprir suas obrigações.
Delegar autoridade	Dar autoridade e discernimento aos subordinados para agirem e utilizarem recursos disponíveis.
Motivar o pessoal	Persuadir e induzir o pessoal a tomar as ações desejadas.
Coordenar atividades	Integrar as atividades nas formas e combinações mais efetivas e eficientes.
Facilitar comunicações	Garantir fluxo livre de informações corretas.
Resolver conflitos	Encorajar diferenças de opiniões e resolver os conflitos resultantes.
Gerenciar mudanças	Estimular a criatividade e inovações visando atingir os objetivos.
Documentar decisões de direcionamento	Documentar decisões envolvendo delegação de autoridade, políticas e procedimentos de comunicação e coordenação etc.

Tabela 8 - Atividades Principais do Controle

ATIVIDADE	DEFINIÇÃO
Desenvolver padrões de performance	Determinar condições ou medidas que existirão quando as tarefas forem executadas corretamente.
Estabelecer sistemas de acompanhamento e relat.	Determinar os dados necessários, quem vai recebê-los e quando.
Medir Resultados	Determinar quanto foi realizado ou os desvios das metas ou padrões.
Iniciar ações corretivas	Reforçar os padrões, ajustar as metas ou replanejar.
Recompensar/disciplinar	Elogiar, remunerar e disciplinar o pessoal devido.
Documentar os métodos de controle	Documentar os padrões, métodos de relatórios e controle, pontos de decisão, etc.

3.3 - TAREFAS ASSOCIADAS COM CADA ATIVIDADE

3.3.1 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DO PLANEJAMENTO DE PROJETOS DE ENGENHARIA DE SOFTWARE

O planejamento de um projeto de ES consiste de todas as atividades de gerenciamento que levam à seleção, dentre alternativas, dos cursos de ação futuros para o projeto e o programa para executar estas ações.

O planejamento envolve a seleção dos objetivos e metas do projeto e das estratégias, políticas, programas e procedimentos para atingi-las. Planejamento é decidir antecipadamente o que fazer, como fazer e por quem.

Podemos citar alguns problemas que o Gerente de Projeto (GP) encontra na fase de planejamento:

- a) A especificação de requisitos do software é frequentemente incompleta, ambígua, inconsistente ou incomensurável, resultando em estimativas irrealistas de prazos e custos
- b) A estimativa de recursos e prazos é difícil de ser feita, pois apesar de existirem numerosos modelos de custos e cronogramas no mercado, todos exigem que o GP estime a quantidade de linhas de código ou alguma outra medida do sistema que AINDA será entregue.
- c) Não existem critérios disponíveis para selecionar a melhor ou mais apropriada metodologia para análise, projeto, teste e gerenciamento do projeto.

A seguir apresentamos as atividades do planejamento, com suas tarefas associadas.

Determinar objetivos ou metas.

Esta atividade não foi diretamente contemplada no modelo proposto. Porém, como as demais atividades que também não foram diretamente contempladas no modelo proposto, elas serão em grande parte auxiliadas pelas informações existentes no ambiente de gerenciamento.

Tarefas:

- a) - Analisar e documentar os requisitos técnicos do projeto: funções, performance, interfaces externas, restrições ao projeto e atributos de qualidade.
- b) - Analisar e documentar os requisitos e restrições gerenciais: cronograma e orçamento disponíveis, metodologias requeridas, padrões, etc.
- c) - Estabelecer metas do projeto, objetivos técnicos, e critérios de sucesso para o projeto.

Desenvolver estratégias.

Estratégias são as metas de longo prazo e os métodos para atingir estas metas.

Esta atividade não foi diretamente contemplada no modelo proposto.

Tarefas :

- a) - Adotar as estratégias da companhia ou do nível hierárquico superior para atingir as metas ou objetivos do projeto
- b) - Desenvolver ou complementar as estratégias da companhia para atingir as metas ou objetivos do projeto .

Desenvolver políticas.

Políticas dizem respeito às decisões pré-determinadas do gerenciamento . O GP estabelecerá políticas dentro do seu projeto para auxiliar e guiar outros GPs, supervisores e membros da equipe na tomada de decisões rotineiras. Estas políticas economizam muito tempo . Elas reduzem a necessidade do GP interagir com todas decisões . Elas também dão um senso de segurança aos membros da equipe.

Esta atividade não foi diretamente contemplada no modelo proposto.

Tarefas:

- a) - Desenvolver ou adotar políticas da companhia que se apliquem ao desenvolvimento de software , i.e. fases e documentos do ciclo de vida , padrões e metodologias de desenvolvimento, abordagens de controle de qualidade e gerenciamento de configuração.
- b) - Desenvolver ou adotar políticas da companhia que se apliquem ao gerenciamento das atividades não técnicas do projeto , i.e., políticas de pessoal (avaliações , férias , ...), organização , recursos , vagas no estacionamento etc.

Determinar cursos de ação.

Na maioria dos projetos , existe mais de uma forma de terminar com sucesso o projeto - mas não com os mesmos custos , cronogramas e riscos . É responsabilidade do GP determinar um número razoável de abordagens ou programas para implementar os requisitos do software .

Esta atividade pode ser realizada no modelo proposto , através da subdivisão em diferentes formas do trabalho a ser realizado, permitindo analisar vários cursos de ação .

Tarefas :

- a) - Analisar e desenvolver para cada objetivo , possíveis cursos de ação .
- b) - Analisar e desenvolver um programa para cada possível curso de ação.
- c) - Estabelecer vantagens , desvantagens , riscos e benefícios para cada curso de ação ou programa .

Tomar decisões.

O GP toma a maior parte das decisões do projeto . Ele em conjunto com seu cliente determina o mais apropriado rumo futuro do projeto . Ele determina qual dos vários cursos de ação é mais apropriado para atender às metas e objetivos do projeto .

O GP também é responsável pela seleção dos métodos pelo qual o projeto será gerenciado e desenvolvido.

Selecionar os métodos e ferramentas para o desenvolvimento do software mais apropriadas e custo-efetivas é uma tarefa muito difícil .

Ferramentas automatizadas não garantem o sucesso de um projeto. Porém a seleção e instalação de um conjunto de ferramentas automatizadas é visto como necessária para o desenvolvimento efetivo de um projeto . O gerente de projeto deve reduzir a complexidade do projeto , impondo uma estruturação no processo de desenvolvimento . As ferramentas automatizadas ajudam a manusear a complexidade restante .

A NASA no seu laboratório de ES e o Rome Air Development Center (RADC) fizeram pesquisas e desenvolvimentos para determinar métodos que o GP pode usar para selecionar o conjunto apropriado de ferramentas de desenvolvimento , métodos ou metodologias .

O RADC desenvolveu um manual para selecionar a metodologia de desenvolvimento de software para um projeto .Este manual é para gerentes técnicos e permite a seleção de metodologias de especificação de requisitos e de projetos , apropriadas para vários ambientes de desenvolvimento e tipos de software .O manual cobre a análise de requisitos , projeto de arquitetura e as fases de projeto detalhado do ciclo de vida do software [RAD85] .

Esta atividade não foi diretamente contemplada no modelo proposto.

Tarefas :

- a) - Selecionar cursos de ação apropriados , baseados nos objetivos e estratégias do projeto . i.e. menor risco , menor custo , menor cronograma , maiores benefícios ...
- b) - Selecionar métodos e ferramentas apropriadas de gerenciamento para atender aos requisitos e objetivos do projeto , i.e., que tipos de relações de autoridade, revisões , padrões de produtos , ferramentas (PERT , CPM, WBS , ..) serão necessários .

Existe um artigo de Kent A. Cori [COR85] , onde é feita uma análise comparativa das 5 ferramentas mais usadas para o planejamento do projeto de ES . Esta comparação visa apenas dar uma idéia ao GP dos objetivos , vantagens e desvantagens de cada técnica . Uma apresentação detalhada da mecânica de cada técnica foge ao escopo deste trabalho e pode ser facilmente encontrada em qualquer literatura sobre o assunto .

As 5 técnicas mais usadas atualmente são : MARCOS , GANTT, PAREDE , CPM e PERT .

Para a utilização de qualquer uma das técnicas existem 7 passos básicos que devem ser seguidos :

- 1 - Definir os objetivos do projeto .
- 2 - Subdividir o trabalho a ser realizado .
Neste ponto usa-se a técnica da EST , Estrutura de Subdivisão do trabalho (Work Breakdown Structure - WBS) , sobre a qual comentaremos algo logo a seguir
- 3 - Sequenciar as atividades do projeto .
- 4 - Estimar a duração e custo de cada atividade .
- 5 - Reconciliar o cronograma com as restrições temporais do projeto .
- 6 - Reconciliar o cronograma com as restrições de recursos do projeto .
- 7 - Revisar o cronograma .

O gráfico de **MARCOS** (MILESTONES) é o método mais simples. É muito útil para projetos pequenos , executados por apenas algumas pessoas , pois não exibe o interrelacionamento das atividades . Ele pode também ser usado para resumir cronogramas complexos , contendo várias tarefas .

As vantagens dos gráficos de **MARCOS** são o baixo custo e a facilidade de preparação .

O gráfico de **GANTT** ou de **BARRAS** é frequentemente usado para projetos pequenos (menos de 25 atividades) e elimina algumas desvantagens do gráfico de **MARCOS** . Este tipo de gráfico é o mais usado atualmente , e as pessoas acham mais fácil entender um gráfico de **GANTT** do que uma rede **PERT** ou **CPM** . Em muitos casos as redes **PERT** e **CPM** são mapeadas diretamente num gráfico de **GANTT** , sendo este então usado para estimar recursos e orçamentos no tempo .

A técnica da **PAREDE** tem sido usada com sucesso recentemente . Escolhe-se uma grande parede e traçam-se linhas verticais separadas de 15 cm e linhas horizontais separadas de 10 cm . O espaço entre cada par de linhas verticais representa uma semana , e as linhas horizontais servem para separar membros da equipe de projeto . Este método funciona bem para uma equipe de 3 a 10 pessoas e mais de 25 e menos de 100 tarefas .

O GP faz um gráfico preliminar de **MARCOS** e uma lista das tarefas , identificando quem é responsável por cada tarefa . Cada tarefa é escrita em dois cartões . O primeiro cartão é rotulado **INICIO** e o segundo **FIM** . Cada membro da equipe recebe seus cartões . A equipe é toda reunida na sala da **PAREDE** do cronograma e cada pessoa fixa seus cartões na parede , numa semana que ela escolha. Conforme o cronograma evolui , as tarefas podem ser subdivididas em subtarefas ou melhor definidas se necessário . As tarefas adicionais não previstas pelo GP são também adicionadas neste instante .

A distribuição dos cartões é continuamente modificada , até que todos os participantes estejam convictos de que as restrições e interrelacionamentos de atividades foram considerados , e de que o projeto é exequível . O GP faz então uma cópia do cronograma completo e distribui para os membros do projeto .

A principal vantagem da técnica da PAREDE é o alto grau de interação que ocorre na reunião . Os conflitos são identificados cedo , discutidos e resolvidos . O método também força os membros da equipe a se comprometerem bem cedo com o GP , com relação às atividades que lhes forem alocadas .

Os métodos PERT (Program Evaluation and Review Technique) e CPM (Critical Path Method) baseiam-se em redes de precedências . São processos dinâmicos , nos quais subdivisões , modificações , adições ou retiradas de atividades podem ser feitas em qualquer ponto do tempo . O desenvolvimento inicial da rede obriga que o projeto esteja completamente definido e estudado . O diagrama de rede exhibe clara e precisamente o plano de ação para a equipe de projeto e os clientes .

Um projeto se adapta bem às técnicas de redes de precedências se :

- possuir atividades bem definidas
- as atividades possam ser iniciadas , terminadas e conduzidas separadamente dentro de uma dada sequencia .
- as atividades se interrelacionem com outras atividades
- as atividades são ordenadas de forma que uma siga a outra numa dada sequencia
- uma atividade uma vez iniciada deve continuar sem interrupção até o fim .

As diferenças entre PERT e CPM não são fundamentais mas apenas de pontos de vista . CPM enfatiza as atividades enquanto PERT é orientado por eventos . PERT leva uma pequena vantagem no sentido de que os eventos podem ser considerados MARCOS . Esses MARCOS podem ser descritos nos contratos ou relatórios para facilitar o controle gerencial . Além disso PERT permite um tratamento probabilístico para suas estimativas de tempos , o que é útil em projetos de ES que geralmente possuem um alto grau de incerteza em suas estimativas .

Variações recentemente desenvolvidas de PERT e CPM permitem que estas técnicas sejam usadas para estimativas, monitoração e controle dos recursos e custos dos projetos . Isto é feito plotando a rede numa escala de tempos no eixo horizontal e fornecendo estimativas do custo e recursos de cada atividade . Isto permite então a projeção de custos e recursos necessários para um período de tempo específico , sem o passo intermediário de traduzir a rede num gráfico de Barras .

A Tabela 9 resume o estudo de Ken A. Cori .

Tabela 9 - Comparação das Ferramentas Utilizadas para o Planejamento do Projeto de Engenharia de Software

	MARCOS	GANTT	PAREDE	CPM	PERT
Orientado por Evento ou Atividade	Evento	Atividade	Evento	Atividade	Evento
Adequado para grandes projetos	Fraco	Fraco	Razoável	Excelente	Excelente
Adequado para pequenos projetos	Bom	Bom	Fraco	Fraco	Fraco
Grau de Controle	Muito Baixo	Baixo	Moderado	Alto	Mais Alto
Receptividade de usuários	Melhor	Excelente	Bom	Razoável	Fraco
Montagem	Mais fácil	Fácil	Mais difícil	Difícil	Bem difícil
Grau de Flexibilidade	Mais baixa	Baixa	Moderada	Alta	Mais Alta
Calculo Manual	Mais fácil	Fácil	Moderada	Difícil	Mais Difícil
Precisão Projeções	Razoável	Razoável	Alta	Bem alta	Mais alta
Custo Prepar. Manut.	Mais baixo	Baixo	Mais alto	Alto	Bem alto
Escopo projeto indef.	Pior	Fraco	Razoável	Bom	Excelente
Logica projeto complexa	Mais fraco	Fraco	Bom	Excelente	Excelente
Data crítica término	Razoável	Razoável	Bom	Bom	Excelente
Checagem frequente	Bom	Bom	Bom	Razoável	Difícil
Atualiz. frequente	Mais fácil	Fácil	Mais difícil	Difícil	Bem difícil
Mudanças lógica frequente	Fraco	Fraco	Fraco	Razoável	Razoável
Apresent. cliente	Bom	Bom	Excelente	Excelente	Excelente

A EST-Estrutura de Subdivisão do Trabalho (WBS) , citada na página 25 , é um procedimento para subdividir um projeto de ES em subprojetos , tarefas , subtarefas , pacotes , etc . Mais formalmente , a EST consiste em enumerar todas as atividades do trabalho em refinamentos hierárquicos de detalhes para organizar o trabalho a ser feito em pequenas tarefas gerenciáveis , com entradas e saídas quantificadas , cronogramas e alocação de responsabilidades . A EST pode ser usada para orçar tempos e recursos ao nível mais baixo das tarefas , e mais tarde como uma base para relatórios de progresso relativos a MARGOS significativos do gerenciamento . Um plano de GEPES baseado na EST contém todas as informações necessárias para estimar custos e cronogramas acuradamente e para fornecer visibilidade e controle durante a execução do projeto. Como regra prática , recomenda-se a quebra do projeto em subtarefas de 1 a 2 semanas de duração .

Determinar procedimentos e regras.

Contrariamente às políticas , procedimentos estabelecem métodos correntes de manusear atividades futuras , e fornece guias para ação ao invés de tomada de decisão. Procedimentos detalham a maneira exata para completar uma atividade e permitem muito pouco desvio . Uma Regra , apesar de similar estabelece ações definidas e específicas para serem tomadas ou não numa situação particular e não permitem nenhum desvio. Esta atividade não foi diretamente contemplada no modelo proposto.

Tarefas :

- a) - Determinar procedimentos e estabelecer métodos para o planejamento e controle do projeto = sistema de gerenciamento do projeto .
- b) - Determinar procedimentos e estabelecer métodos para o desenvolvimento do sistema = metodologia padrão de desenvolvimento de software .
- c) - Estabelecer ou adotar procedimentos para a organização e alocação de pessoal no projeto .
- d) - Estabelecer planos e procedimentos para garantia de qualidade do software .
- e) - Estabelecer métodos para preparação e publicação de documentos .

Desenvolver programas de ação

Um programa de ação é o plano do projeto.

Esta atividade está integralmente contemplada no modelo proposto.

Tarefas :

- a) - Particionar os requisitos técnicos e gerenciais em tarefas mensuráveis.
- b) - Determinar milestones , prioridades e cronogramas necessários para atender os objetivos do projeto .
- c) - Estimar HH necessários e outros recursos para atender os objetivos do projeto.
- d) - Preparar planos e procedimentos que evitem riscos.

Prever situações futuras

O GP tem a responsabilidade de fazer predições sobre o futuro e como estas afetarão o projeto .

As estimativas sobre o futuro podem ser feitas de duas formas. A primeira é a estimativa de eventos futuros como disponibilidade de pessoal , taxas de inflação , disponibilidade de novos hardwares, e o impacto que estes eventos terão sobre o projeto de ES.

A segunda é estimar como o projeto de ES reagirá com relação a estas expectativas futuras .

Ou seja , uma estimativa prevê o futuro e a outra estima como o projeto reagirá a este futuro previsto .

Esta atividade pode ser facilmente implementada utilizando-se as informações disponíveis , para simulações com redes de PERT.

Tarefas :

- a) - Estimar o ambiente futuro do projeto .i.e. determinar a disponibilidade dos recursos necessários.
- b) - Estimar o uso dos recursos e fundos orçados.
- c) - Determinar possíveis riscos técnicos e gerenciais.

Preparar orçamentos

Esta atividade está contemplada no modelo proposto.

Tarefas :

- a) - Determinar os custos e os requisitos do projeto, estabelecer políticas , procedimentos e programas e estimar pessoal e recursos necessários.
- b) - Alocar recursos orçados para as tarefas ou centros de custos.

Documentar os planos do projeto.

O GP tem a responsabilidade de documentar os planos do projeto, bem como preparar outros documentos como planos de controle de qualidade , planos de gerenciamento de configuração , planos de alocação de pessoal , e outros planos , como testes. O plano de projeto é o meio principal do GP comunicar-se com os agentes externos que se interfaceiam com o projeto , e de informá-los o que o gerente pretende fazer e o que o gerente espera deles.

A maior parte desta documentação pode ser obtida do modelo proposto.

Tarefas :

- a) - Documentar o plano de gerenciamento do projeto.
- b) - Preparar outros documentos de planejamento como testes , qualidade de software , gerenciamento de configuração , alocação de pessoal e recursos.

3.3.2 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DA ORGANIZAÇÃO DO PROJETO DE ES

Organizar um projeto de ES compreende todas as atividades gerenciais que resultam em uma estrutura formal de tarefas de ES e relações de autoridade entre estas tarefas. A dificuldade desta atividade é de que não existem regras de decisão para selecionar a estrutura organizacional mais apropriada (projeto, matriz, funcional), assim como as responsabilidades pelas tarefas e atividades estão muitas vezes indefinidas ou obscuras.

A seguir apresentamos as atividades da organização do projeto, com suas tarefas associadas.

Identificar e agrupar as tarefas necessárias .

Esta atividade está implementada no modelo proposto.

Tarefas :

- a) - Definir e particionar os requisitos de software em atividades e tarefas.
- b) - Medir e agrupar as tarefas de software nas entidades organizacionais.

Selecionar e estabelecer a estrutura organizacional.

Muitos GP não podem ter o luxo de selecionar o melhor tipo de estrutura organizacional para o projeto, já que muitas vezes isto é determinado pelas políticas da companhia a nível corporativo. Entretanto, independentemente de quem o faz, o tipo e a estrutura organizacional que melhor atendam às necessidades e metas do projeto e do ambiente do projeto devem ser identificados.

Existem 3 tipos clássicos de estruturas organizacionais que são a *FUNCIONAL*, *PROJETO* e *MATRIZ*. A estrutura organizacional do projeto independe do gerenciamento do projeto, que é executado em qualquer dos 3 tipos de estruturas organizacionais de projeto. É claro que algumas estruturas favorecem uns aspectos do gerenciamento, e outras favorecem outros aspectos.

Robert Youker [YOU77] num artigo não muito recente, mas ainda bastante atual, apresenta uma descrição simples destas estruturas organizacionais para projetos e que transcrevemos abaixo.

A organização *FUNCIONAL* (Fig.4) é a dominante atualmente. É o modelo clássico em que temos a pirâmide hierárquica, com a chefia no topo e os subordinados abaixo. A organização é subdividida em diferentes unidades funcionais ou departamentos.

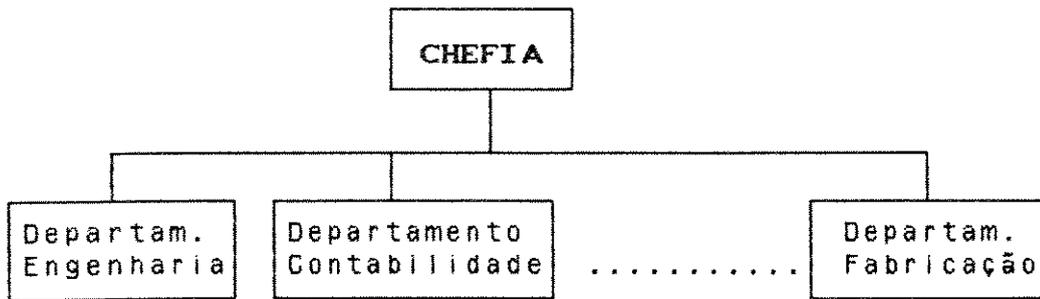


Fig. 4 - Organização Funcional

O ponto forte desta organização é a centralização de recursos similares, e a definição clara de planos de carreira. Como fraquezas, normalmente aparecem conflitos sobre a prioridade relativa de cada projeto em múltiplos projetos dentro do departamento, e os departamentos técnicos tendem a dar mais ênfase às suas especialidades técnicas do que às metas do projeto. Falta de motivação e inércia são também outros problemas.

Na organização por PROJETO (Fig.5), todos os recursos necessários para atingir um objetivo específico são retirados da estrutura funcional regular da empresa e alocados numa estrutura independente e autônoma chefiada pelo GP.

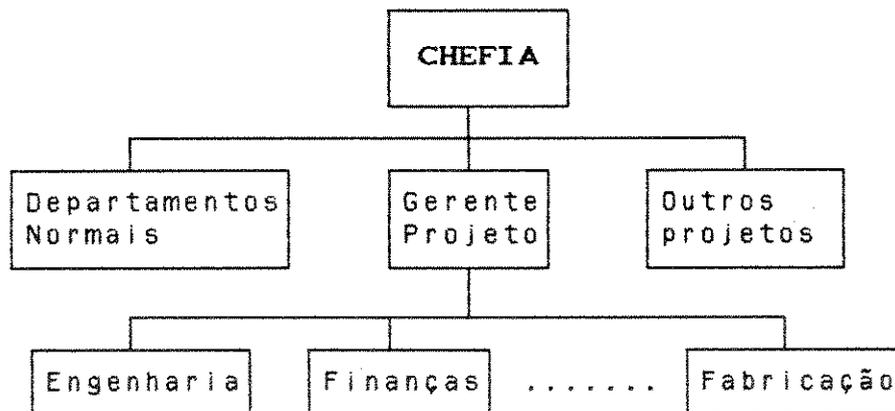


Fig. 5 - Organização por Projeto

É interessante notar que internamente a estrutura por PROJETO é FUNCIONAL. A vantagem da organização por PROJETO reside em ter objetivo e chefe únicos. O claro entendimento de um único objetivo desenvolve um espírito de equipe claro e positivo no time do projeto.

Entretanto a estrutura por PROJETO também tem suas fraquezas . Existe ciúme dos outros chefes de departamentos com relação à equipe de projeto . Existe também duplicação e uso ineficiente de recursos . Outro aspecto importante é a segurança no emprego após o término do projeto , pois o pessoal normalmente perde suas posições anteriores na estrutura funcional enquanto estavam trabalhando no projeto .

A estrutura MATRICIAL (Fig.6) é uma estrutura multidimensional que tenta maximizar os pontos positivos e minimizar os pontos negativos das estruturas FUNCIONAIS e por PROJETO . Ela combina a estrutura hierárquica vertical padrão com uma estrutura sobreposta , horizontal ou lateral de um coordenador de projeto .

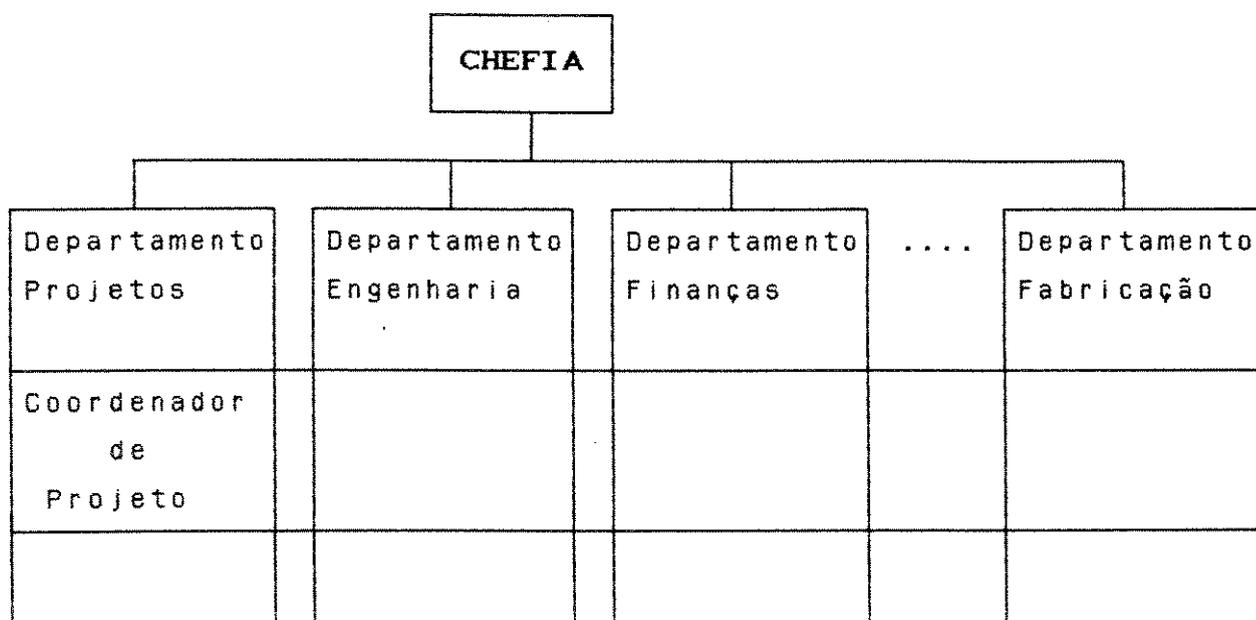


Fig. 6 - Organização Matricial

Os benefícios principais de uma estrutura MATRICIAL são o equilíbrio dos objetivos , a coordenação através das linhas de departamentos e a visibilidade dos objetivos do projeto como um todo através do Coordenador do Projeto .

A principal desvantagem é de que o funcionário no meio está trabalhando para dois chefes ; verticalmente para o chefe funcional e horizontalmente para o coordenador do projeto . O GP sente que tem pouca autoridade com relação aos departamentos funcionais e o chefe do departamento funcional sente que o gerente do projeto está interferindo no seu território . A solução para este problema é definir papéis , responsabilidades e autoridades de cada um claramente . O GP especifica o que deve ser feito e o chefe do departamento é responsável por como será feito .

Existem variantes da estrutura Matricial tais como estruturas matriciais FORTES ou FRACAS .

O GP deve verificar qual estrutura organizacional que melhor se adapta às características do projeto e da própria empresa .

A Tabela 10 apresenta um resumo dos pontos fortes e fracos de cada estrutura organizacional :

Tabela 10 - Pontos Fortes e Fracos de cada Estrutura Organizacional

<u>FORTES</u>	<u>FRACOS</u>
<u>FUNCIONAL</u>	
<ul style="list-style-type: none"> * Organização já existe * Recrutamento, treinamento e retenção do pessoal é mais fácil * Padrões, técnicas e métodos já estão estabelecidos 	<ul style="list-style-type: none"> * Nenhuma pessoa tem completa responsabilidade ou autoridade sobre o projeto * Problemas de interface são difíceis de resolver * É difícil monitorar e controlar o projeto
<u>PROJETO</u>	
<ul style="list-style-type: none"> * Existe uma posição central de autoridade e responsabilidade * Decisões podem ser tomadas rapidamente * Motivação é alta 	<ul style="list-style-type: none"> * Organização deve ser criada * Recrutamento, treinamento e retenção do pessoal é mais difícil * Não consegue economia de escala * Os projetos tendem a se perpetuar * Padrões, técnicas e procedimentos devem ser desenvolvidos.
<u>MATRICIAL</u>	
<ul style="list-style-type: none"> * Posição central de responsabilidade melhorada (com relação à funcional) * Interface entre funções controlada mais facilmente * Recrutamento, treinamento e retenção mais fácil que na PROJETO * Mais fácil iniciar e terminar do que no PROJETO * Padrões, técnicas e procedimentos já estabelecidos * Melhor e mais flexível uso das pessoas 	<ul style="list-style-type: none"> * Responsabilidade e autoridade é partilhada entre dois gerentes * Muito fácil mover pessoas de uma organização para outra * Maior competição por recursos do que na PROJETO

Além da estrutura organizacional , existe o aspecto da própria equipe do projeto . A equipe do projeto pode ser organizada conforme tres modelos básicos :

- a equipe DEMOCRÁTICA
- a equipe do CHEFE PROGRAMADOR
- a equipe HIERÁRQUICA

Existe um excelente artigo de Marilyn Matei [MAT81] , o qual recomendamos como leitura interessante para o GP . Transcreveremos abaixo o primeiro parágrafo de sua conclusão : "As características das tarefas fazem com que as tarefas interfiram umas com as outras . Um projeto grande é normalmente difícil . A estrutura de uma equipe que é efetiva para um aspecto de uma tarefa pode ser totalmente incorreta para outro aspecto . Na seleção da estrutura da equipe é importante usar um algoritmo de tomada de decisão para priorizar , dar pesos ou combinar as variáveis cruciais das tarefas " .

A Tabela 11 apresenta um resumo dos pontos fortes e fracos de cada organização de equipe de projeto . Como é feita a organização , deixamos para o leitor ler em várias referencias existentes [BAK72] [WEI71] [YOU76] .

Tabela 11 - Pontos Fortes e Fracos de cada tipo de Organização de Equipe

<u>FORTES</u>	<u>FRACOS</u>
<u>DEMOCRÁTICA</u>	
* Democracia funciona bem para problemas complexos e difíceis	* Produtividade baixa pois comunicação é complexa
* Melhora a produtividade e qualidade	* Funciona mal em projetos "apressados"
* Satisfação no trabalho é alta	
<u>CHEFE PROGRAMADOR</u>	
* Trabalho termina rápido	* Chefe Programador deve ter muita experiencia
* Funciona melhor com datas rígidas	* Abaixa a moral no projeto
	* Não é aplicável a problemas mal definidos
<u>HIERÁRQUICA</u>	
* Funciona para grandes projetos	* Funciona mal com projetos pequenos ou de pesquisas
* Uma pessoa é responsável pelo esforço da equipe	* Requer muito pessoal de supervisão

O modelo proposto não contempla a seleção do tipo de organização, mas permite a implantação de qualquer tipo de organização que tenha sido selecionada.

Tarefas :

- a) - Selecionar o tipo de organização :
 - Selecionar uma estrutura de linha ou de suporte (staff).
 - Selecionar uma estrutura por projeto , funcional ou matricial.
 - Selecionar os tipos de equipes.
- b) - Antecipar as mudanças organizacionais causadas pelas fases do ciclo de vida do software .
- c) - Fazer mudanças organizacionais quando necessário.
- d) - Identificar e estabelecer as relações entre o pessoal do projeto e os clientes/usuários.
- e) - Reconhecer e usar , quando existir , uma estrutura organizacional informal.

Criar posições na organização

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Criar títulos e descrições para cada posição organizacional criada.
- b) - Definir escopo , deveres e relacionamento de cada posição organizacional.

Definir responsabilidades e autoridade.

Autoridade pode ser definida como o grau de discreção, dando às pessoas ocupando as posições organizacionais o direito de usar seu julgamento na tomada de decisões .

Diz-se normalmente que a responsabilidade não pode ser delegada ou compartilhada. Koontz e O'Donnell[K0076] apóiam esta visão, definindo responsabilidade como a obrigação devida pelos subordinados aos seus superiores pelo exercício de autoridade a eles conferida de forma a alcançar os resultados esperados . Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Definir as responsabilidades de cada posição organizacional.
- b) - Definir a autoridade que deve ser exercida para cobrir as responsabilidades de cada posição .
- c) - Estabelecer autoridades e relacionamentos entre posições e tarefas.

Estabelecer a qualificação das posições

Esta atividade não está contemplada no modelo proposto.

Tarefas :

- a) - Estabelecer as qualificações de cada posição na estrutura do projeto.
- b) - Estabelecer necessidades de treinamento e educação para cada posição da estrutura do projeto.
- c) - Estabelecer requisitos de experiência para cada posição do projeto.

Documentar a estrutura organizacional

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Documentar os planos e hipóteses assumidas.
- b) - Documentar a estrutura organizacional , responsabilidades e autoridade das posições , autoridade entre tarefas a serem executadas , e os relacionamentos entre as posições organizacionais .

3.3.3 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DA ALOCAÇÃO DE PESSOAL DE UM PROJETO DE ES

O objetivo da alocação de pessoal é garantir que os papéis do projeto sejam executados por pessoas que possam e desejem ocupá-los.

Um dos problemas da alocação de pessoal é que não existem procedimentos e técnicas para a seleção dos gerentes de projeto. É prática comum elevar à posição de gerente os analistas ou programadores que demonstraram suas capacidades como analistas ou programadores, sem serem beneficiados com o treinamento de gerenciamento. Cabe lembrar que as pessoas são promovidas até seu nível de incompetência.

Isto significa que um excelente programador, poderá tornar-se um péssimo analista ao ser promovido para tal posição, caso não esteja devidamente qualificado para a mesma.

Atividades e tarefas da alocação de pessoal.

Preencher as posições organizacionais

Na alocação de pessoal de qualquer projeto, os fatores seguintes devem ser considerados no preenchimento das posições. Deficiências em qualquer destes fatores podem ser superadas por superioridade em outros fatores. Por exemplo, deficiências em educação podem ser superadas por uma maior experiência, um treinamento especializado ou entusiasmo pelo trabalho.

Deficiências sérias devem ser causa de rejeição.

Educação - O candidato tem a educação mínima necessária para o trabalho? O candidato tem a educação apropriada para seu futuro crescimento na empresa?

Experiência - O candidato tem um nível aceitável de experiência? É o tipo certo e variado de experiência?

Treinamento - O candidato está treinado na metodologia, linguagem, ou equipamento a ser usado, e na área de aplicação do sistema de software?

Motivação - O candidato está motivado para o emprego? Trabalhar para o projeto? Trabalhar para a companhia?

Compromisso - O candidato demonstrará lealdade para o projeto, a companhia e as decisões tomadas?

Auto-determinação - O candidato tem iniciativa própria, desejoso de levar o projeto até o término, sem direcionamento excessivo?

Afinidade com o grupo - O candidato se encaixa na estrutura atual? Existem alguns conflitos potenciais que necessitem ser solucionados?

Inteligência - O candidato tem a capacidade de aprender, de assumir tarefas difíceis e adaptar-se a ambientes variados?

Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Determinar os padrões de contratação para gerentes de projeto , engenheiros de software , analistas , programadores e pessoal de suporte capazes e produtivos .
- b) - Transferir pessoal qualificado para cada posição de dentro da própria companhia .
- c) - Recrutar pessoal qualificado para cada posição de outras firmas , instituições governamentais ou de turmas de graduação das universidades .

Assimilar o pessoal novo

O GP é responsável não somente por contratar o pessoal , como também por familiarizar o pessoal com os procedimentos , recursos e planos do projeto necessários para garantir sua efetiva integração no projeto .
Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Orientar os membros novos do projeto com os objetivos , políticas e planos do projeto .
- b) - Familiarizar e orientar os membros novos do projeto com os recursos , procedimentos e políticas da companhia .

Educar ou treinar pessoal

Não é sempre possível recrutar ou transferir empregados com exatamente as habilidades necessárias para um projeto particular . Assim sendo , o GP é responsável pela educação e treinamento do pessoal alocado , para garantir que eles atendam os requisitos do projeto .

Educação é diferente de treinamento . Educação envolve ensinar as bases , teorias e conceitos de uma disciplina com vistas a um retorno a longo prazo . Treinamento significa ensinar uma habilidade , um conhecimento de como usar , operar ou fazer alguma coisa .Ele é tipicamente necessário no futuro imediato e tem um retorno a curto prazo .

Alguns dos principais métodos de treinamento utilizados atualmente são o on-the-job , cursos formais na própria empresa, cursos em universidades e escolas locais , auto-estudo , e palestras formais ou informais .

Cada indivíduo dentro de uma organização deve ter um plano de treinamento que mostre metas e os passos necessários para atingirem-se estas metas . A administração superior deve apoiar ativamente um programa de treinamento .

Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Educar e treinar gerentes em gerenciamento e administração.
- b) - Educar e treinar engenheiros de software e o pessoal técnico na área de aplicação do projeto , métodos e políticas de desenvolvimento de software e técnicas avançadas do estado da arte .
- c) - Educar e treinar o pessoal de suporte nos métodos , habilidades e equipamentos necessários .

Fornecer condições para o desenvolvimento geral

Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Desenvolver conhecimento profissional , atitudes e habilidades.
- b) - Aumentar a base de habilidades da companhia.

Avaliar o pessoal

A avaliação do pessoal deve ser concentrada no desempenho e não na personalidade (a não ser que interfira com o desempenho). Deve-se utilizar o gerenciamento por objetivos, avaliando pelas metas realizadas e não pela limpeza , pontualidade , número de gols no campeonato interno e assim por diante .

Esta atividade está contemplada no modelo proposto através das avaliações de desempenho e produtividade .

Tarefas :

- a) - Avaliar o pessoal.
- b) - Estabelecer os objetivos do empregado para o próximo período de avaliação.

Dar benefícios e incentivos financeiros

Esta atividade não está contemplada na modelo proposto.

Tarefas :

- a) - Determinar a escala de salários , política de promoções e benefícios.
- b) - Recompensar financeiramente o pessoal de projeto de acordo com seu treinamento , habilidades e desempenho.

Terminar a alocação

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Transferir , realocar , ou terminar a alocação do pessoal ao final do projeto ou quando o desempenho do pessoal for inaceitável.

Documentar as decisões de pessoal

Esta atividade não está diretamente contemplada no modelo proposto .

Tarefas :

- a) - Documentar os planos de alocação de pessoal , políticas de desenvolvimento e escala de salários .
- b) - Documentar os planos de treinamento individuais refletindo o curso do trabalho e o progresso feito .

3.3.4 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DO DIRECIONAMENTO DE UM PROJETO DE ES

Direcionar um projeto de ES consiste de todas as atividades de gerenciamento que tratam de aspectos interpessoais através dos quais o pessoal de projeto compreenda e contribua para alcançar as metas do projeto .

Atividades e tarefas :

Suprir liderança

O GP fornece liderança para sua equipe. Seu trabalho é interpretar os planos e requisitos para garantir que todo mundo na sua equipe de projeto está trabalhando na direção de uma meta comum .

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Interpretar planos e requisitos para o pessoal do projeto.
- b) - Alinhar as metas do pessoal do projeto com as metas do projeto.

Supervisionar o pessoal

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Dar ao pessoal alocado instruções diárias , orientação e disciplina necessárias para realizar suas obrigações.
- b) - Tomar decisões a curto prazo sobre ações afetando o projeto .

Delegar autoridade

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Definir a autoridade do pessoal do projeto compatível com suas responsabilidades .
- b) - Alocar tarefas a subgrupos , equipes e indivíduos.
- c) - Delegar autoridade até o nível mais baixo do projeto .

Motivar o pessoal

Liderança apenas não é suficiente . O GP é responsável por motivar e inspirar o pessoal a fazer seu melhor . Várias técnicas motivacionais das correntes principais do gerenciamento são aplicáveis ao gerenciamento do projeto de ES, tais como : gerencia por objetivos [DRU64], Hierarquia de necessidades de Maslow [MAS43] [MAS54], Fatores higiênicos de Herzberg [HER59], e algumas vezes apenas o carisma do gerente. Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Motivar e inspirar o pessoal para tomar as ações desejadas.
- b) - Utilizar técnicas de motivação .
- c) - Reconhecer atenção especial necessitada por engenheiros e pessoal técnico altamente qualificados .

Coordenar atividades

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Reconciliar diferenças de abordagens e usar estas diferenças para o bem do projeto.
- b) - Informar as partes interessadas de ações que afetaram ou os afetarão .
- c) - Obter consenso nas ações.

Facilitar comunicações

Nada destrói mais rapidamente a moral de uma organização do que rumores falsos . Um bom GP manterá seu pessoal bem informado , para que rumores sejam rapidamente dispersados.

Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Agilizar e acompanhar o progresso das informações , internas e externas ao projeto .
- b) - Disseminar os planos do projeto e outros documentos pertinentes.

Resolver conflitos

Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Resolver desavenças entre pessoal do projeto e entre o projeto e entidades externas ao projeto .
- b) - Reduzir a oportunidade de conflitos futuros , removendo fontes potenciais de desavenças sempre que possível .

Gerenciar mudanças

Esta atividade não é exequível no modelo proposto.

Tarefas :

- a) - Encorajar pensamentos independentes e inovativos para atingir as metas do projeto .
- b) - Controlar as mudanças mas não eliminá-las.

Documentar decisões de direcionamento

A maior parte desta documentação está contida no modelo proposto.

Tarefas :

- a) - Documentar alocação de tarefas e autoridade.
- b) - Documentar decisões sobre os canais de comunicação.
- c) - Documentar decisões envolvendo mudanças e resolução de conflitos .

3.3.5 - TAREFAS ASSOCIADAS COM AS ATIVIDADES PRINCIPAIS DO CONTROLE DE UM PROJETO DE ES

Controlar um projeto de ES são todas as atividades de gerenciamento que garantem que as operações estão sendo feitas de acordo com o planejado. O controle mede a performance contra as metas e planos, mostra quando existem desvios, e colocando em movimento ações para corrigir desvios, ajuda a garantir o cumprimento do planejado.

O processo de desenvolvimento de software é inerentemente sujeito a riscos (custos e prazos não cumpridos e requisitos não atendidos). As fontes de risco podem ser grupadas em tres categorias [R0086] :

- *perturbações* - mudanças nos requisitos , detecção de problemas, erros e falhas.
- *pessoal* - pessoas erradas , excesso ou falta de pessoal .
- *ambiente do projeto* - metodologia indefinida , qualidade desconhecida , erros tardiamente detectados , controle inadequado.

Se o projeto deve ter sucesso , os riscos potenciais devem ser eliminados ou controlados . O sistema de controle de um projeto é baseado no princípio usual de estabelecer loops de feedback para garantir que o sistema sendo controlado busca seus objetivos de produzir um produto correto , em tempo e dentro do orçamento .

Atividades e tarefas :

Desenvolver padrões de desempenho

Esta atividade não está diretamente contemplada no modelo proposto.

Tarefas :

- a) - Desenvolver e especificar a quantidade e a qualidade dos padrões de desempenho .
Um *PADRÃO* é um conjunto de critérios disponíveis , aprovados e documentados , usados para determinar a validade de uma ação ou objeto . Um *Padrão de ES* é um conjunto de : (1) procedimentos que definem os processos para obtenção de um *produto* e (2) descrições que definem a quantidade e qualidade de um *produto* de um projeto de ES .
- b) - Padronizar e documentar os procedimentos de engenharia, produtos , milestones , configurações básicas e documentos das fases do ciclo de vida do desenvolvimento de software .
- c) - Especificar os métodos para garantir a qualidade do software .
"Controle de qualidade de software" é um padrão de ações planejadas e sistemáticas necessárias para ter-se confiança de que os itens ou produtos estão conforme os requisitos técnicos estabelecidos .

O controle de qualidade do software inclui os métodos de desenvolvimento (para especificação de requisitos e projeto) , padrões , métodos de gerenciamento de configuração , procedimentos de revisões técnicas , padrões de documentação , verificação e validação , e especificações e procedimentos para teste . O controle de qualidade do software é uma das principais técnicas de controle disponíveis para o GP .

- d) - Desenvolver medidas para a produtividade e qualidade da engenharia de software .

Estabelecer sistemas de monitoração e relatórios .

O processo de monitoração envolve a medição da performance atual . O sistema de relatórios guarda , analisa e filtra informações sobre o progresso do projeto , fornecidas pelo sistema de monitoração . O GP revê o estado atual , progresso e problemas identificados , servindo como base para decisões .

Para a monitoração do processo de desenvolvimento podemos usar :

Gerenciamento de bases - É uma estratégia que é usada para controlar o ciclo de vida do desenvolvimento de software .

Revisões do orçamento - Uma comparação de estimativas com gastos reais , feitas pelo cliente , gerente , supervisores , para determinar se os últimos estão de acordo ou se desviam-se do planejado .

Auditoria independente - Fazer uma auditoria independente do projeto de ES com o objetivo de determinar se está de acordo com os planos , especificações e padrões . Uma auditoria independente é uma auditoria feita por uma organização externa , que não esteja associada com o projeto . Ocasionalmente é interessante trazer uma equipe independente com a finalidade de auditar o projeto . Um aspecto positivo é que a equipe de auditoria pode fornecer uma opinião totalmente imparcial , não favorecendo nenhum lado , no caso de existir alguma animosidade organizacional . Esta equipe externa pode também auxiliar o pessoal do projeto com conhecimentos especializados . O aspecto negativo da auditoria independente é de que os auditores têm de alcançar o estado atual do projeto , sobrepunhando a curva de aprendizagem . A equipe de auditoria necessita envolvimento total para ser efetiva .

Revisões formais - A análise dos conceitos e especificações pelos clientes , usuários e gerentes para medir o progresso sempre que um milestone for atingido .

Padrões do processo de desenvolvimento - Um padrão que define os procedimentos ou operações usadas para produzir ou alcançar um produto .

Controle de qualidade de software - Um padrão de ações sistemáticas e planejadas para dar confiança de que o processo está conforme os padrões estabelecidos .

Caderno de desenvolvimento de unidades - Uma forma específica de caderno de anotações de desenvolvimento, que fornece uma abordagem ordenada ao desenvolvimento de uma unidade de programa, e que fornece ao GP visibilidade e controle sobre o processo de desenvolvimento.

Para a monitoração do *produto*, podemos usar:

Gerenciamento de configuração - Um método de controlar e fazer relatórios sobre o status do software.

Padrões do produto - Um padrão que define no que se constitui a completeza de itens que são produzidos como resultado de um processo.

Controle de qualidade de software - Um conjunto de padrões de ações sistemáticas e planejadas que garantam que o produto está conforme as especificações de requisitos técnicos estabelecidas.

Teste (unidades e integração) - A execução controlada do código dos programas para expor erros.

O teste de unidade é o teste de uma unidade de código (geralmente um módulo) pelo programador que programou a unidade, com um conjunto de dados de testes preparados por outra pessoa. O teste de integração é o teste de cada unidade ou subelemento em combinação com o sistema sendo testado, para mostrar a existencia de erros entre as unidades ou subelementos e o sistema.

Verificação e validação - A garantia de que cada fase do ciclo de vida do software interpreta corretamente a especificação da fase anterior, e de que cada produto de software entregue funciona como prescrito nos requisitos.

Walkthroughs e inspeções - Uma crítica do produto de software, pelo pessoal da própria equipe de desenvolvimento com o único objetivo de encontrar erros.

O sistema de inspeção é outra forma de revisão pelo próprio pessoal de desenvolvimento, desenvolvida por Michael Fagan da IBM em 1976. O sistema de inspeção era originalmente muito mais formal do que o walkthrough. Mudanças recentes nos procedimentos de walkthroughs os tornaram mais formais. Ainda que o termo preferido para uma revisão pelos próprios companheiros ainda seja walkthrough, o termo inspeção está tornando-se bastante popular.

Para os relatórios, poderemos ter:

Relatórios de orçamento - Compara o orçado com os gastos e permite fazer novas estimativas de orçamentos.

Relatórios de cronograma - Fornece o estado do cronograma e das milestones atingidas

Relatórios de hh por atividades - Fornece informação sobre o número de hh de pessoal que foram gastas numa certa atividade

Relatórios de homens dias por tarefas - Fornece o número de homens dias alocados para uma tarefa. Uma tarefa pode ser ou não maior do que uma atividade.

- Relatórios de milestones* - Informa quais milestones foram alcançadas ou atrasadas e a razão para o atraso .
- Relatórios de progresso do projeto* - Uma narrativa indicando o estado do progresso ou uma lista das atividades terminadas.
- Relatórios de atividades* - Fornece , tipicamente em algum período de tempo , quais atividades foram executadas.
- Cartas de tendências* - Mostra as tendências em áreas como orçamento , número de erros encontrados , hh perdidos por doenças , etc . Os relatórios de tendências servem para prever o futuro .
- Relatórios de mudanças significativas* - Um relatório geral de mudanças , indicando exceções e mudanças significativas , boas ou ruins . Normalmente eles mostram uma perda de progresso.

Uma parte importante do processo de controle diz respeito à quantificação e medição de termos abstratos como "produtividade" , "desempenho" e "progresso" . Uma vez que se utilize a EST , o progresso é a determinação periódica , normalmente semanal , do Status de cada pacote de trabalho em termos de percentagem completada.

A avaliação do desempenho é a comparação entre o progresso e os gastos reais com relação ao planejado . O desempenho serve para identificar desvios do planejado e determinar soluções para corrigir estes desvios .

A produtividade é definida como a produção por HomemHora , seja em termos de linhas de código , documentos ou qualquer outra unidade de produto.

As avaliações de progresso , desempenho e produtividade foram integralmente implementadas no modelo proposto.

Tarefas :

- a) - Estabelecer o sistema de monitoração e relatórios .
- b) - Determinar o tipo , frequência , fonte e destino dos relatórios do projeto.
- c) - Estabelecer um sistema de gerenciamento da configuração do software .
- d) - Estabelecer ferramentas de status que deem visibilidade sobre o progresso e não apenas recursos utilizados .
- e) - Selecionar ferramentas de software , técnicas e índices para monitorar a produção , i.e. PERT , CPM , cartas de GANTT , etc .
- f) - Coletar os dados do projeto para desenvolver métodos de medição e estimativas mais corretos .

Medir resultados

Esta atividade está implementada no modelo proposto.

Tarefas :

- a) - Medir e auditar gastos reais , progresso e status de um projeto .
- b) - Medir e avaliar a qualidade e quantidade dos produtos de software .
- c) - Determinar os desvios do plano do projeto , padrões ou requisitos .

Iniciar ações corretivas

Esta atividade não está diretamente implementada no modelo proposto.

Tarefas :

- a) - Iniciar ações corretivas se os planos , padrões ou requisitos não estão sendo atendidos .
- b) - Determinar a melhor abordagem para iniciar as ações corretivas .

Recompensar e disciplinar

Esta atividade não está diretamente implementada no modelo proposto.

Tarefas :

- a) - Elogiar e recompensar aqueles que contribuíram para o sucesso das metas do projeto .
- b) - Tomar ações disciplinares contra aqueles que poderiam mas não contribuíram para o projeto .

Documentar os métodos de controle

A maior parte desta documentação está contida no modelo proposto.

Tarefas :

- a) - Documentar os padrões , procedimentos de controle e métodos de medida .
- b) - Publicar relatórios periódicos sobre os desvios entre o planejado e os resultados reais .
- c) - Documentar as ações tomadas para corrigir os desvios do planejado .

Baseado no modelo clássico de gerenciamento de projetos de engenharia de software detalhado neste capítulo , é apresentado no próximo capítulo o modelo proposto para o GEPES . No próximo capítulo , na seção 4.4 - O funcionamento do modelo proposto , ficará claro o relacionamento das atividades do modelo clássico de gerenciamento de projetos de engenharia de software com as funções do modelo proposto .

CAPÍTULO 4 - MODELO PROPOSTO

Neste capítulo é apresentado o modelo proposto para um sistema de gerenciamento genérico baseado no modelo clássico de gerenciamento que : pode ser automatizado , cobre parte das falhas apresentadas pelos atualmente disponíveis , e é independente do modelo de desenvolvimento adotado para o projeto .

Praticamente todas as funções apresentadas no modelo proposto estão implementadas no protótipo .

4.1 - Informações Principais

Um sistema de gerenciamento de projetos (não apenas de software) engloba 3 grupos principais de informações :

- 1 - O projeto
- 2 - Os recursos
- 3 - Os procedimentos e metodologias adotados pela empresa.

Nenhuma das ferramentas existentes leva em consideração o terceiro grupo de informações , ainda mais quando referem-se à engenharia de software , ficando o mesmo implícito na forma como o gerente planeja o projeto .

Nossa intenção é que o modelo proposto , assim como a ferramenta automatizada implementada , permitam a um gerente de projetos configurar o ambiente de engenharia de software onde os projetos serão gerenciados tendo , portanto , acesso aos grupos 2 e 3 de informações .

Estes dois últimos grupos de informações formarão o "ambiente de gerenciamento" e o primeiro grupo formará o "ambiente de projetos".

Adotando o MER de Chen [CHE76] , podemos representar estes 3 grupos da seguinte forma :

4.1.1 - O Projeto

O projeto é uma entidade que possui atividades e atributos próprios . As atividades por sua vez também possuem vários atributos , conforme a Figura 7.

O conjunto de atributos associados ao projeto e às atividades é o mínimo necessário para que possam ser implementadas as tarefas do modelo clássico de gerenciamento .

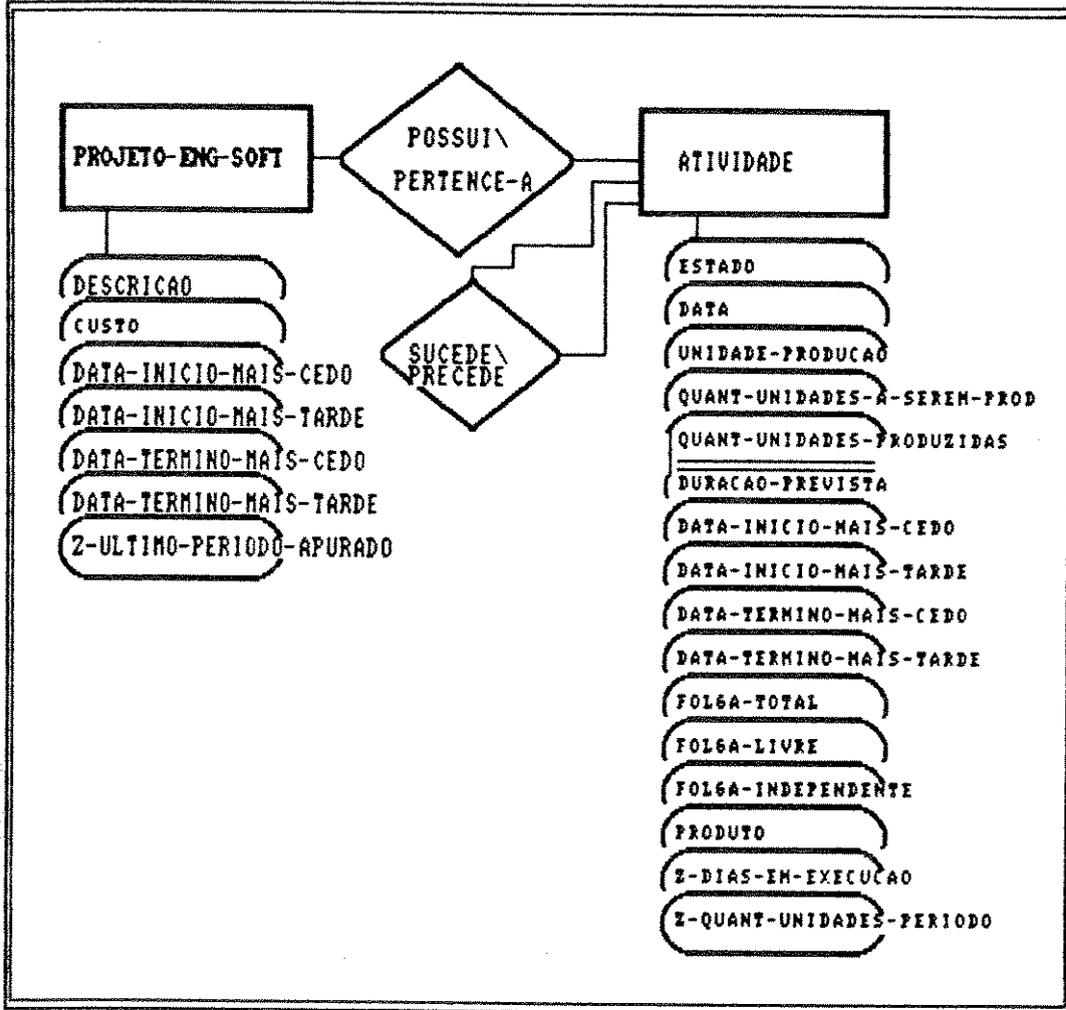


Fig. 7 - 0 Projeto

4.1.2 - OS RECURSOS

Os recursos são os consumíveis do projeto, e podem ser classificados em dois grupos: os recursos materiais (CATEGORIAS RECURSOS MATERIAIS) e os recursos humanos (CATEGORIAS RECURSOS HUMANOS).

Cada categoria de recursos humanos (analistas, programadores, etc.) possui uma ou mais pessoas a ela alocada, e cada pessoa poderá estar subordinada a um ou mais gerentes conforme o tipo de estrutura funcional adotada.

A Figura 8 representa este grupo de informações.

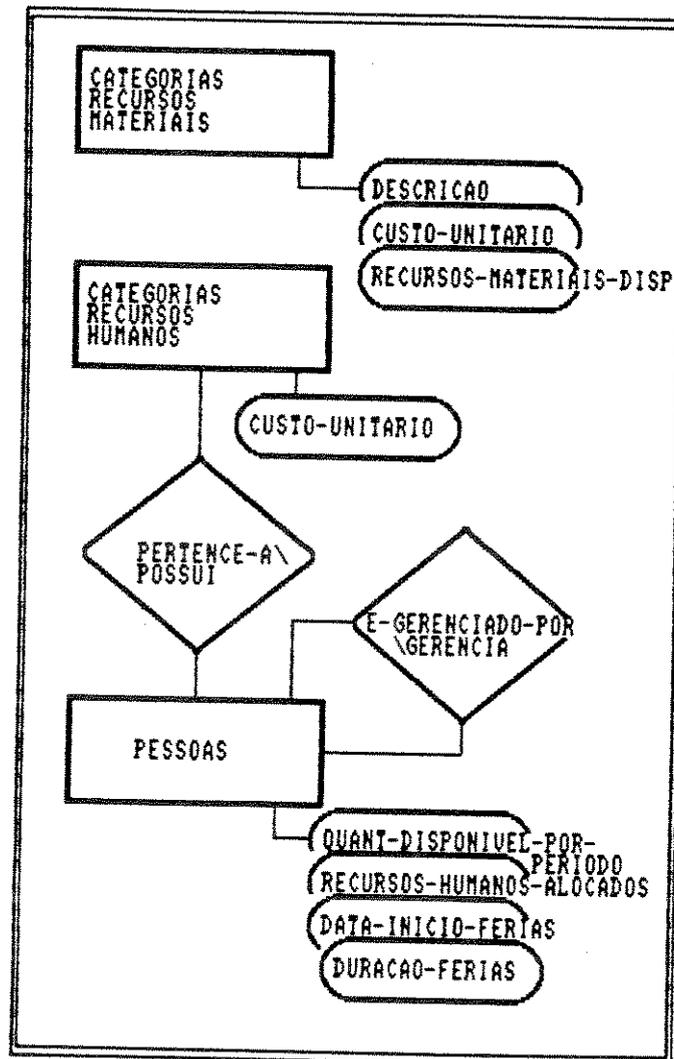


Figura 8 - Os Recursos

4.1.3 - OS PROCEDIMENTOS E METODOLOGIAS

Cada produto possui um ciclo próprio de desenvolvimento . Para o software , convencionou-se chamar o mesmo de "Ciclo de Vida do Software" . Independentemente do tipo de ciclo de vida adotado , ele certamente será definido por fases , que por sua vez poderão ser constituídas de sub-fases . Como item final de ciclo de evolução de fases em sub-fases e assim por diante, temos os tipos de atividades comportados pelas subfases do nível mais baixo da hierarquia .

As fases ou sub-fases produzirão um produto , e tanto as fases como os tipos de atividades estão associados com uma metodologia, conforme a Figura 9, o que força e normaliza a adoção de metodologias no projeto de engenharia de software, e permite que seja utilizado qualquer modelo de desenvolvimento para o software.

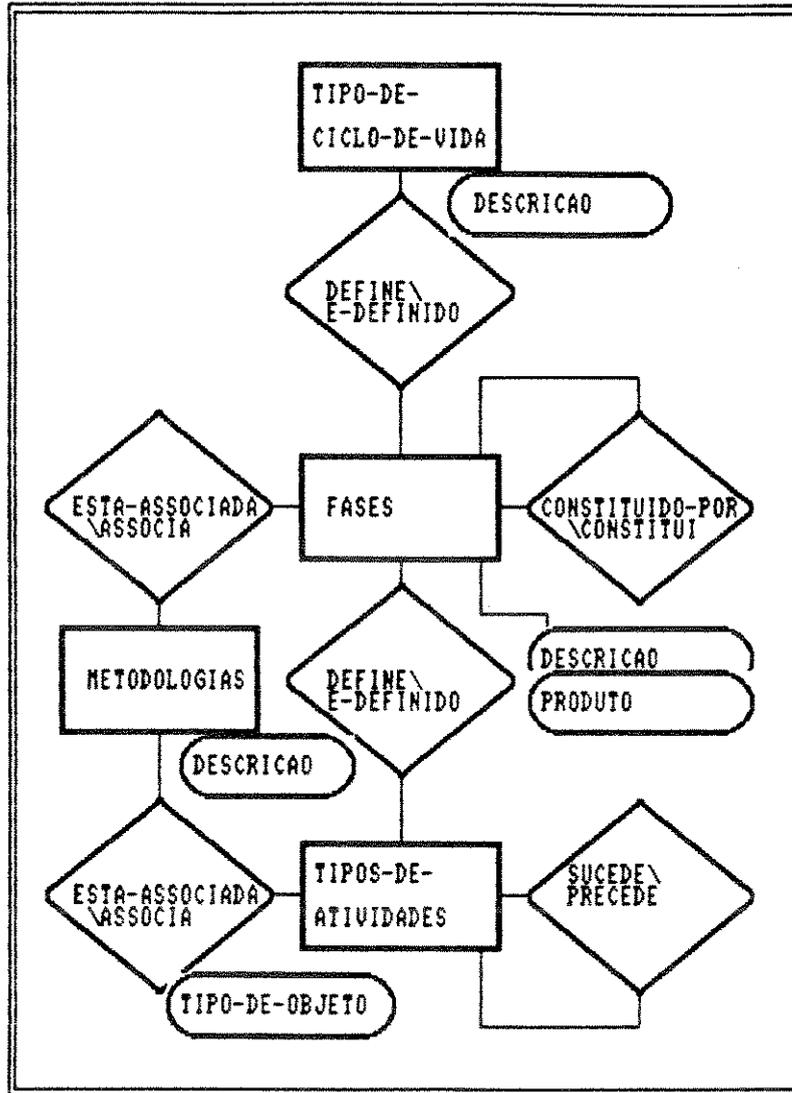


Figura 9 - Procedimentos e Metodologias

4.2 - O Ambiente de Gerenciamento

A integração dos dois primeiros grupos de informações, Figura 10, cria o que denominamos um "ambiente de gerenciamento". Cada empresa, ou setores diferentes de uma mesma empresa podem possuir ambientes de gerenciamento diferentes, conforme o ciclo de vida do produto a ser produzido, ou os tipos de recursos utilizados. Cada um destes ambientes de gerenciamento diferentes, com seus recursos, procedimentos e métodos podem ser configurados pelo gerente de projetos e coexistirem simultaneamente na ferramenta, o que mais ainda caracteriza seu uso genérico.

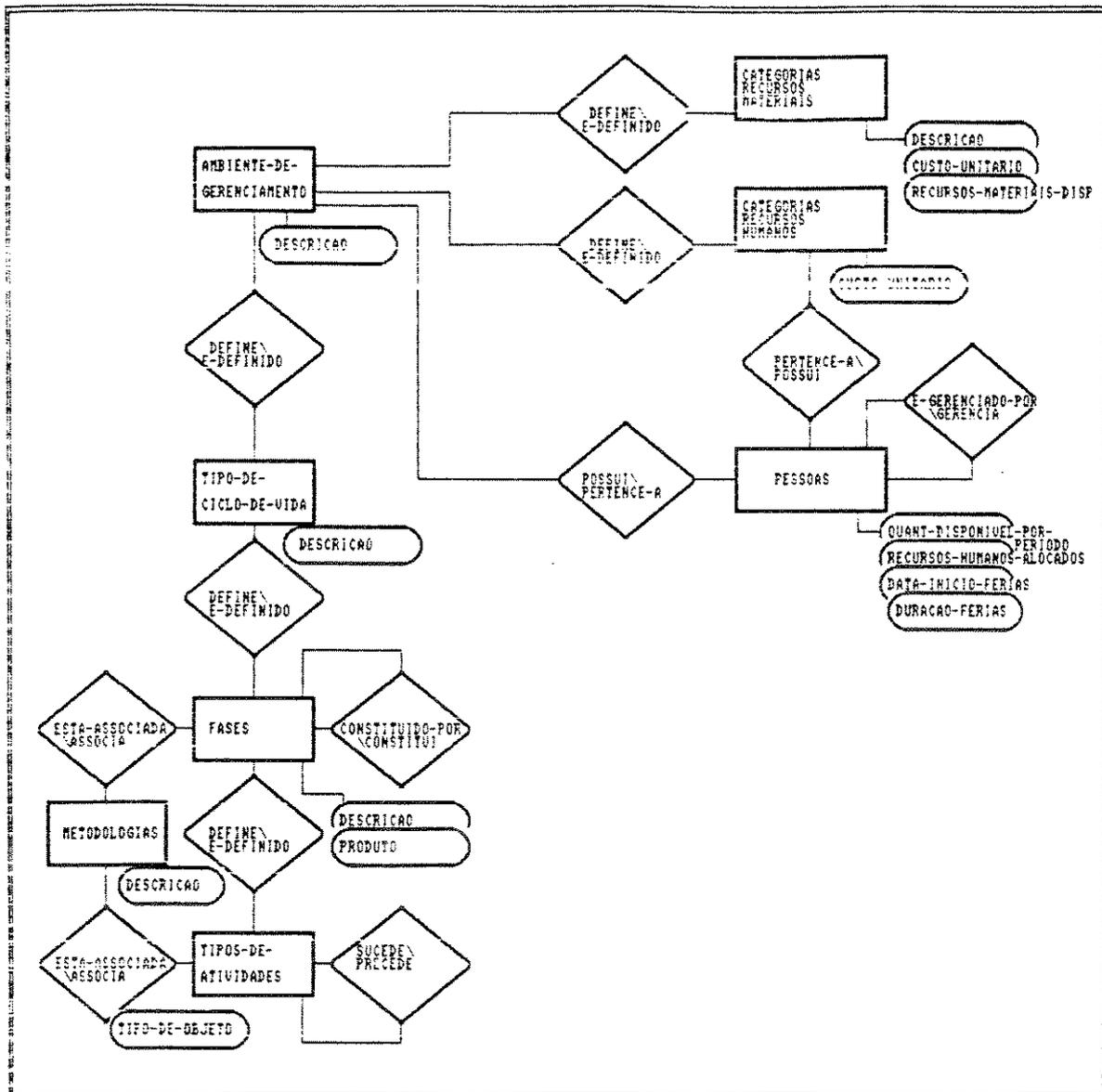


Fig. 10 - Ambiente de Gerenciamento

4.3 - Diagrama de Entidades e Relacionamentos do Modelo Proposto

Os diversos projetos inserem-se num ambiente de gerenciamento, através de seus relacionamentos com os recursos e os procedimentos da empresa, completando o modelo proposto, conforme finalmente mostrado na Figura 11.

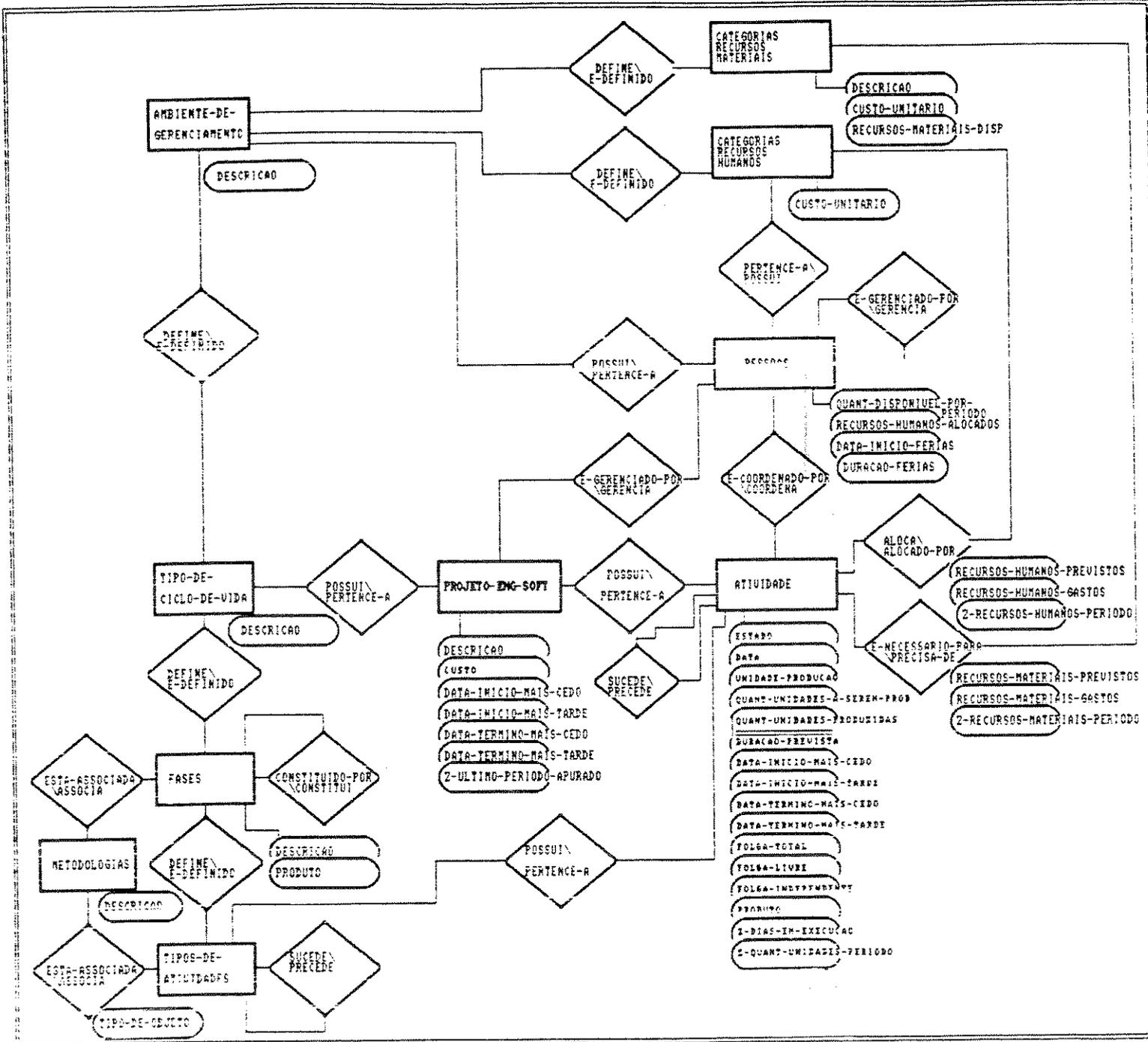


Fig. 11 - MER do Modelo Proposto .

4.4 - O Funcionamento do Modelo Proposto

Na Figura 10 , temos o ambiente de gerenciamento, que é um dos dois grupos principais de informações do modelo (o outro são os projetos).

O ambiente de gerenciamento é a alma do modelo, pois é nele que estão contidas as informações de Engenharia de Software.

Para utilização da ferramenta automatizada , o GP, o responsável por metodologias, ou qualquer outra pessoa com o conhecimento e a responsabilidade para tal , deverá *configurar* o ambiente de gerenciamento, que consiste no fornecimento de todas as informações que aparecem na Figura 10 . Mais adiante explicaremos qual a opção escolhida para o preenchimento destas informações.

Uma vez configurado o ambiente , poderá ter início o gerenciamento dos projetos de engenharia de software .

Para fins de explanação do funcionamento do gerenciamento de projetos de engenharia de software , utilizando o modelo proposto na Figura 11 , concordamos com Howes [HOWB4] de que ajuda bastante imaginar o GEPES como sendo constituído de duas partes separadas porém relacionadas :

- o planejamento do projeto e
- a execução do projeto .

Estendemos esta idéia e dizemos que o GEPES é constituído de 3 partes separadas porém relacionadas :

- o planejamento
- a execução e
- o controle de qualidade do projeto .

O relacionamento entre estas partes é representado pela Figura 12.

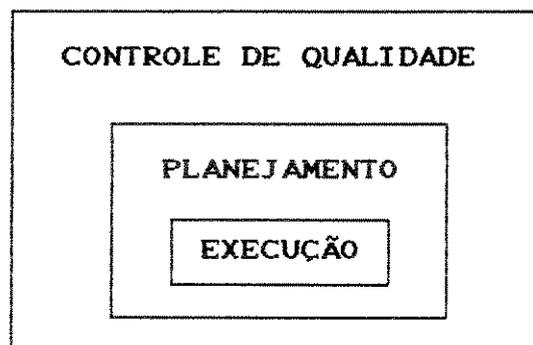


Fig. 12 - Relacionamento entre as Partes Principais do GEPES

O planejamento engloba as funções de *Planejamento*, *Organização* e *Alocação de Pessoal* do modelo clássico de gerenciamento de projetos . A execução engloba as funções de *Direcionamento* e *Controle* também do modelo clássico de gerenciamento de projetos .

A Figura 12 mostra que o controle de qualidade é uma atividade básica do GEPES e que engloba os outros dois componentes do GEPES. Não adianta nada terminar um projeto dentro dos prazos e custos , sem estar conforme as especificações de requisitos .O controle de qualidade poderia ser encarado como a parte mais técnica do GEPES como a utilização de análise e projetos estruturados , revisões e walkthroughs , gerenciamento de configuração , etc .

Neste trabalho não abordaremos a parte referente ao controle de qualidade do software , restringindo-nos às atividades gerenciais e menos técnicas do GEPES , que seriam os componentes do planejamento e da execução do projeto .

Segundo Howes os componentes do PLANEJAMENTO são :

- subdivisão do trabalho
- quantificação
- sequenciamento do trabalho
- orçamentos
- cronogramas
- montagem da equipe

Os componentes da EXECUÇÃO do projeto são :

- apuração dos gastos reais
- medição do progresso
- avaliação da performance
- medição da produtividade
- controle de mudanças
- estimativas

As tarefas principais de cada atividade do modelo clássico de GEPES já foram citadas anteriormente . O GP deverá escolher um modelo de Ciclo de Vida do Software que mais se adapte ao projeto em questão e dar toda ênfase ao controle da qualidade do software.

Agora restringiremo-nos às tarefas específicas do GEPES , modelando os relacionamentos e atributos de cada entidade que definiremos para o planejamento e execução do projeto .

4.4.1 - PLANEJAMENTO DO PROJETO DE ENGENHARIA DE SOFTWARE SEGUNDO O MODELO PROPOSTO

4.4.1.1 - SUBDIVISÃO DO TRABALHO

A subdivisão do trabalho é a decomposição de um projeto em pedaços gerenciáveis, referidos como "Pacotes" ou "Fases" de trabalho (daqui por diante nos referiremos indiferentemente a pacotes ou fases).

Os Pacotes devem ser constituídos de apenas um tipo de trabalho, podendo ter várias ATIVIDADES. Estas atividades devem ter curta duração, em geral não mais de 40hh ou 5 dias úteis, devem ser logicamente relacionadas com como o trabalho será feito, e deve ser possível alocar a responsabilidade de execução da atividade a uma pessoa.

A decomposição do projeto deve ser feita de uma forma hierárquica, com refinamentos sucessivos dos níveis de detalhes. Isto é feito automaticamente em nosso modelo, ao associarem-se os projetos a tipos de ciclos de vida, dentro dos quais está implícita esta decomposição hierárquica em fases ou pacotes, sendo as atividades do projeto associadas aos tipos de atividades que as fases do ciclo de vida possuem. Uma decomposição assim estruturada é a nossa já definida EST.

Temos pois uma entidade PROJETO que é subdividida ou composta por PACOTES e que pode ter mais de uma ATIVIDADE. Uma forma de incluirmos o Controle de Qualidade do Software nos componentes de Planejamento e Execução do projeto, seria alocar atividades ao Controle de Qualidade, por exemplo uma atividade poderia ser a "inspeção da especificação de requisitos".

4.4.1.2 - QUANTIFICAÇÃO

A quantificação consiste em 7 passos:

- 1° passo - identificar para cada atividade dos pacotes, a sua unidade de medida, que pode ser por exemplo linhas de código, relatórios, gráficos, fluxogramas, diagramas, documentos, etc.
- 2° passo - alocar quantidades para cada atividade. Se a unidade de medida for fluxogramas por exemplo, indicar o número de fluxogramas que serão gerados pela atividade.
- 3° passo - alocar o número de homens-horas necessários para cada atividade. Este número deve ser alocado por classes de tipos de homens-horas. Sugerimos no mínimo 3 classes:
 - Classe 1 - Gerência (Supervisão)
 - Classe 2 - Mão de obra direta (Programadores, Analistas,
 - Classe 3 - Auxiliares (Datilografia, arquivistas, etc.)

- 4° passo - Alocar a quantidade de recursos materiais (por exemplo horas de computador necessárias), também classificando , como por exemplo
 Classe 1 - Independente da máquina alvo , pequena capacidade .
 Classe 2 - idem , grande capacidade .
 Classe 3 - na máquina alvo .
- 5° passo - Alocar para cada atividade um custo fixo .
- 6° passo - Alocar para cada atividade um percentual de overhead , que seriam os custos indiretos da atividade (cópias , diária de hotel , etc.)
- 7° passo - Calcular o custo de cada atividade , que seria dado por :

Custo Atividade =

$$\begin{aligned}
 & \left[\sum \{ (\text{Classe HH}) * (\$ \text{ da classe de HH}) \} \right. \\
 & \qquad \qquad \qquad + \\
 & \qquad \qquad \qquad \sum \{ (\text{Classe computador}) * (\$ \text{ da hora de máquina da classe}) \} \\
 & \qquad \qquad \qquad + \\
 & \qquad \qquad \qquad \left. \text{Custo fixo da atividade} \right] \\
 & \qquad \qquad \qquad * \\
 & \qquad \qquad \qquad \left[1 + (\% \text{ custos indiretos da atividade}) \right]
 \end{aligned}$$

Para fins de simplificação , podem-se suprimir os passos 5 e 6 , suprimindo-os também do cálculo do sétimo passo .

Além disto é importante guardar a época em que foi feito este cálculo do custo da atividade .

Assim temos para cada atividade mais as seguintes informações que portanto foram incluídas na Figura 11:

- Quantidade de unidades a serem produzidas
- Classes e quantidades de HH
- Classes e quantidades de recursos materiais
- Custo total (redundante)
- data do cálculo do custo.

4.4.1.3 - SEQUENCIAMENTO DO TRABALHO

Após a quantificação dos Pacotes o GP deve determinar a sequência em que eles serão executados. Este sequenciamento fornece uma visão da ordem em que o trabalho será executado. O sequenciamento é feito ao nível das atividades, informando-se as sucessoras e predecessoras de cada atividade. Deve ser verificado se este sequenciamento é válido, de acordo com as relações de precedências definidas para os tipos de atividades do ambiente de gerenciamento, conforme ainda a Figura 11.

A medida em que vão sendo definidas as sequências, a EST (Pag.28) pode necessitar ser revista, para garantir que a subdivisão do trabalho é compatível com a sequência em que ele deve ser executado. Pode ser necessário mudar ou criar novas atividades, para melhor definir a forma e a ordem de como o trabalho será realizado. Como as atividades possuem curta duração, o início de uma atividade sempre estará amarrado ao término de outra, não podendo iniciar-se uma atividade quando por exemplo 50 % da anterior já estiver concluída. Entretanto o início pode ser em qualquer época após o término da atividade anterior, e não necessariamente imediatamente após o término desta. Portanto para cada atividade no nível mais inferior da hierarquia teremos

- atividades precedentes
- atividades sucessoras.

4.4.1.4 - ORÇAMENTOS

Uma vez feita a quantificação dos pacotes o GP disporá de uma estimativa de custos para o projeto. Esta estimativa deverá ser aprovada pelo nível hierárquico superior do gerenciamento. Se aprovada, esta estimativa será então o ORÇAMENTO do projeto, e que poderá ser apresentado sob várias formas, como custos de HH, custos de equipamentos, custos diretos, overheads, etc, e em qualquer dos níveis hierárquicos em que foi feita a EST.

Caso a estimativa não seja aprovada, todo o trabalho deverá ser revisto, até que se consiga chegar a um orçamento compatível com os recursos disponíveis, ou mostrar que o projeto é inexecutável com estes mesmos recursos.

Lembramos aqui que é normalmente mais importante cumprir o cronograma do que o orçamento.

4.4.1.5 - CRONOGRAMAS

O custo total do projeto será fixo . A distribuição deste custo dependerá do cronograma do projeto . A confecção do cronograma é normalmente bastante trabalhosa se feita manualmente . O processo de iteração para aprovação do orçamento faz com que o cronograma seja constantemente modificado . É imprescindível a automação deste procedimento. Para projetos pequenos , podem-se usar diversas técnicas , inclusive manualmente . Para grandes projetos a técnica mais recomendada (já comentada anteriormente) é o PERT .

Para confecção do cronograma nos basearemos nas necessidades x recursos disponíveis de hh e equipamentos . Necessitaremos portanto saber para cada classe de HH ou de máquinas , a quantidade de horas disponíveis. Estas informações são fornecidas quando da montagem da equipe .

O GP deverá poder simular diversas situações , fixando ou deixando livres as datas de início e/ou término de fases ou atividades em qualquer nível hierárquico , ou aumentando ou diminuindo as quantidades de horas de máquinas ou pessoas disponíveis , analisando os casos pessimistas , mais prováveis e otimistas.

À medida em que o projeto for sendo executado , este cronograma será constantemente revisado , o que mais uma vez justifica sua automação .

4.4.1.6 - MONTAGEM DA EQUIPE

Uma vez definido o projeto , orçamentos e cronogramas , deve-se montar a equipe de projeto . Independentemente de qual será a estrutura organizacional do projeto , essa equipe será formada por PESSOAS .

Cada PESSOA possuirá :

- um NOME
- um SUPERIOR
- uma CLASSIFICAÇÃO (de HH)
- época e duração de FÉRIAS
- os pacotes/atividades em que está ALOCADA
- e poderá ainda ser RESPONSÁVEL por uma atividade , ou mesmo gerenciar o projeto .

Com a montagem da equipe será feito um cruzamento com as informações do orçamento e do cronograma para verificar em quais pacotes existe excesso/falta de pessoal (material) alocado ou responsável , e quais pessoas (recursos materiais) estão sub ou sobrecarregadas de trabalho . Conforme o caso , poderá ser necessário proceder a um nivelamento dos recursos , para compatibilizar os mesmos , retornando-se ao item 5.4.1.1. Uma vez compatibilizadas as necessidades e os recursos , teremos terminado a parte referente ao PLANEJAMENTO .

Chama-se o conjunto do ORÇAMENTO E CRONOGRAMA como o **PLANO** ou **LINHA BASE** do projeto .

Agora iniciaremos a parte referente à EXECUÇÃO do projeto de ES , que consiste em controlar o esforço de desenvolvimento de acordo com o PLANO do projeto .

Para a EXECUÇÃO teremos então :

4.4.2 - A EXECUÇÃO DO PROJETO DE ENGENHARIA DE SOFTWARE SEGUNDO O MODELO PROPOSTO

4.4.2.1 - APURAÇÃO DOS GASTOS REAIS

Para apuração dos gastos reais , utilizam-se os métodos clássicos de contabilidade , alocando uma conta para cada pacote , e debitando nesta conta os gastos reais .

Os gastos de HH podem ser debitados diretamente pelo sistema automatizado , devendo-se informar apenas as exceções (ausências de funcionários) . Os gastos de HH devem ser debitados semanalmente . Os gastos monetários podem ser debitados semanal ou mensalmente .

Um gráfico apresentando os gastos reais com relação à linha base é bastante útil para o GP .

4.4.2.2 - MEDIÇÃO DO PROGRESSO

A medição do progresso consiste em determinar periodicamente (usualmente semanalmente) o STATUS de cada pacote . O STATUS é medido em termos de percentagem completada .

Com a percentagem completada calcularemos a produtividade e a performance .

A medição do progresso é a pedra angular do gerenciamento de projetos . Ainda mais do projeto de engenharia de software, como veremos abaixo .

O nível mais baixo da medição do progresso é o das atividades. Para uma atividade comum da engenharia de produção, como por exemplo fabricar chupetas, se numa atividade está prevista a fabricação de 9832 chupetas e já foram fabricadas 2458 chupetas, pode-se afirmar, sem muito receio de errar, que 25% da atividade está completada, e a partir deste número, calcular a produtividade e performance do projeto "chupetas".

Se entretanto estivermos lidando com uma atividade de um projeto de engenharia de software, para o qual tenhamos estimado produzir 1000 módulos com 400 hh, e que já tenhamos produzido 300 módulos e gasto 200 hh, qual será a % completada desta atividade?

Primeiramente uma premissa, que nem sempre é verdadeira mas que com a evolução da disciplina de engenharia de software tenderá cada vez mais a ser correta, de que o gerente (ou sua equipe) foi bastante cuidadoso em suas estimativas, e o resultado final da atividade serão realmente 1000 módulos à 400 hh.

Aqui configura-se um problema: existem módulos que são mais trabalhosos do que outros, não sendo portanto uniforme o esforço ao longo da atividade. Permanece portanto a pergunta: Como calcular a percentagem completada desta atividade?

Podemos calcular diretamente de duas formas:

pelo **produzido** :
$$\frac{300 \text{ módulos produzidos}}{1000 \text{ módulos estimados}} = 30 \% \text{ completada}$$

ou

pelo **gasto** :
$$\frac{200 \text{ hh gastos}}{400 \text{ hh previstos}} = 50 \% \text{ completada}$$

Os que leram [BR075] "The Mythical Man-month" devem estar escandalizados com a segunda opção, pois 200 hh gastos em 400 hh previstos não significam 50% do projeto concluído. Entretanto estando as estimativas razoavelmente corretas, este valor de 50% não é tão mítico, e é **MUITO MAIS CORRETO** do que pelas unidades produzidas (obtendo 30%) que os autores de trabalhos sobre gerenciamento de projetos preconizam.

Porém como sabemos que o esforço não é uniforme, nota-se que este valor calculado apenas com os hh também não é correto. Tentaremos achar uma forma de correlacionar os quatro valores disponíveis, simultaneamente, para determinar a percentagem completada, e não 2 a 2 como fizemos anteriormente.

Considerando uma distribuição uniforme do esforço durante a realização da atividade, que não é real, mas é compensada por um menor esforço em algum período da atividade no caso de haver um maior esforço durante algum outro período da atividade, podemos traçar uma curva de distribuição do esforço.

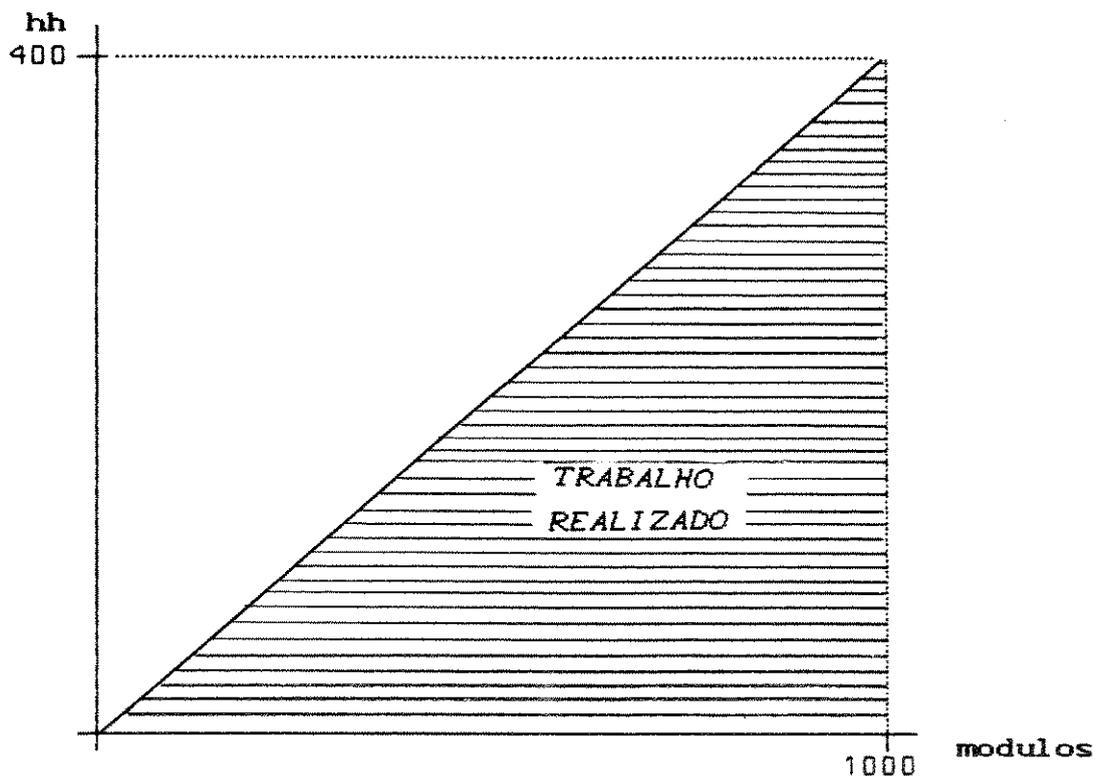


Fig. 13 - Distribuição do Esforço

A área embaixo da curva, que chamaremos de *TRABALHO REALIZADO*, possuirá um valor máximo ao término da atividade. Podemos dizer que a percentagem completada em qualquer instante da atividade será o trabalho realizado até aquele instante, dividido pelo trabalho total da atividade.

Ou de outra forma, podemos traçar a curva

*% completada X (recursos * unidades) :*

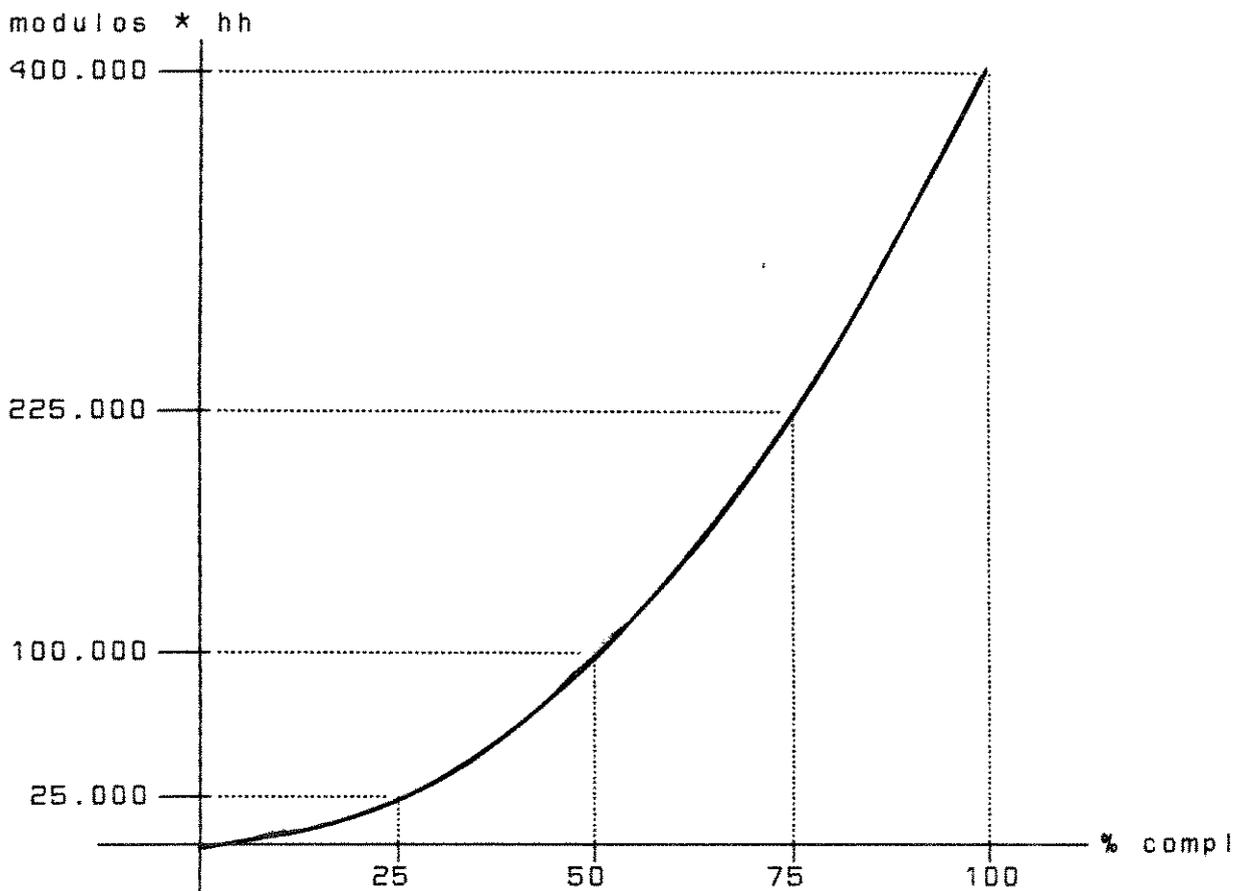


Fig. 14 - Curva % COMPLETADA X (RECURSOS*UNIDADES)

Em qualquer um dos dois casos , extraímos a fórmula que correlaciona simultaneamente os quatro valores :

$$\% \text{ completada atividade} = \sqrt{\frac{\text{unidades produzidas} * \text{recursos gastos}}{\text{unidades previstas} * \text{recursos previstos}}}$$

para nosso exemplo em questão , obteríamos

$$\% \text{ completada} = \sqrt{\frac{300 * 200}{1000 * 400}} = 38.7 \%$$

que é um valor bem mais razoável do que 30 % ou 50 % .

e agora que as atividades possuem as mesmas unidades , podemos calcular a percentagem completada do conjunto , que será dada por :

$$\% \text{ completada } (1+2) = \sqrt{\frac{(12 + 16) * (200 + 300)}{(40 + 80) * (400 + 800)}} = 31,2 \%$$

Existe entretanto uma forma mais fácil para se calcular a percentagem completada de um conjunto de atividades , sem ser necessário reduzi-las a um mesmo conjunto de unidades e que conduz ao mesmo resultado , sendo dada por :

$$\% \text{ Compl. Conjunto Ativ} = \frac{\sum_{i=1}^n (\% \text{ comp. ativ}_i * \text{hh orçado ativ}_i)}{\sum_{i=1}^n \text{hh orçado}_i}$$

onde n é o número total de atividades do conjunto de atividades considerado para cálculo da percentagem completada .

Esta é a fórmula que utilizaremos para qualquer nível hierárquico ou fase do projeto (inclusive para todo o projeto) , bastando por exemplo considerar as atividades que pertencem à uma dada fase , para obtermos a percentagem completada daquela fase . Se considerarmos *todas* as atividades , teremos a percentagem completada do projeto .

Para o exemplo anterior das 2 atividades teríamos :

$$\% \text{ completada}_{1+2} = \frac{(38.7\% * 400) + (27.38\% * 800)}{400 + 800} = 31.2 \%$$

4.4.2.3 - AVALIAÇÃO DA PERFORMANCE

A medição do progresso diz ao GP se o projeto está sendo executado dentro do prazo previsto ou não, ou seja, se o conteúdo do trabalho está sendo executado ou não conforme o planejado.

Pode acontecer de o projeto estar sendo executado conforme o planejado, porém com seus custos de HH e máquinas acima ou mesmo abaixo do planejado. Esta informação é obtida com a avaliação da performance.

A avaliação da performance dentro de um período de tempo qualquer, utiliza tres informações disponíveis para o GP:

- a - o CUSTO REAL de cada pacote no período.
- b - o CUSTO ORÇADO para cada pacote no período, ou seja, a linha base do período.
- c - o CUSTO TEÓRICO do pacote no período, que é obtido multiplicando-se a percentagem completada do pacote pelo total de HH do pacote.

Este custo pode ser subdividido em seus componentes isolados para uma análise mais cuidadosa.

A avaliação da performance é feita traçando-se as curvas dos tres itens a, b e c.

Se a curva do item c acompanha a curva do item b o projeto está progredindo conforme planejado.

Se a curva do item c acompanha a curva do item a, a produtividade está conforme o planejado.

Quaisquer desvios devem ser cuidadosamente analisados pelo GP.

4.4.2.4 - MEDIÇÃO DA PRODUTIVIDADE

Se houver desvios entre as curvas anteriormente citadas, existe motivo para preocupação. O GP deverá tomar providências para sanar o problema. Antes porém de sanar o problema o GP deverá primeiramente isolá-lo.

Suponhamos que a curva c esteja acompanhando a linha base, mas desvie violentamente da curva a, com os custos reais da curva a acima da curva c.

Isto significa (conforme já dito) que o conteúdo do trabalho está sendo executado conforme planejado (cronogramado), mas que o custo está significativamente acima do planejado. A conclusão óbvia é BAIXA PRODUTIVIDADE (ou má estimativa).

Para determinar quais pacotes estão causando baixa produtividade e poder corrigir esta tendencia num tempo que não prejudique o andamento do projeto , o GP deve possuir um relatório pelo menos semanal da produtividade .

O cálculo da produtividade de um pacote básico é feita como abaixo :

Primeiramente o pacote básico recebe uma unidade qualquer de medida e a quantidade desta unidade que deverá ser produzida , e que não necessita ser a mesma de nenhuma de suas atividades componentes . Já possuímos o custo orçado do pacote , que seria o somatório de suas atividades . Dividindo-se o custo do pacote pelo número de unidades , teremos o CUSTO UNITÁRIO ORÇADO DE UM PACOTE . Dividindo o custo real pelo número de unidades produzidas teremos o CUSTO UNITÁRIO REAL DE UM PACOTE , e para calcular o CUSTO UNITÁRIO TEÓRICO DE UM PACOTE usa-se a fórmula

$$\text{CUSTO UNITÁRIO TEÓRICO} = \frac{\text{Custo real do pacote}}{(\% \text{ compl.pacote}) * (\text{total de unidades do pacote})}$$

Onde a fórmula para percentagem completada do pacote já foi anteriormente detalhada . Similarmente pode-se calcular o custo unitário orçado e real de qualquer pacote em qualquer nível hierarquico da EST usando o mesmo procedimento e as mesmas fórmulas dos pacotes básicos.

Para calcular a produtividade do projeto , que será a mesma de qualquer nível hierárquico , usamos as fórmulas

$$\text{CUSTO UNITÁRIO ORÇADO DO PROJETO / NÍVEL HIERARQUICO} = \frac{\text{Custo total do projeto (nível hier.)}}{\text{Total de unidades do projeto (nível hier.)}}$$

$$\text{CUSTO UNITÁRIO REAL DO PROJETO / NÍVEL HIERÁRQUICO} = \frac{\text{Custo real do projeto (nível hier.)}}{\text{Unidades produzidas do projeto (nível hier.)}}$$

$$\text{CUSTO UNITÁRIO TEÓRICO DO PROJETO / NÍVEL HIERARQUICO} = \frac{\text{Custo real do projeto (nível hier.)}}{(\% \text{comp.projeto(n.h.)}) * (\text{Total unidades projeto(n.h.)})}$$

De posse deste relatório semanal e do gráfico com as curvas a, b e c, o GP pode determinar tendências e isolar os pacotes problemáticos.

É também necessário consultar o cronograma do projeto para determinar problemas, pois ainda que a produtividade seja satisfatória, os pacotes podem não estar iniciando ou terminando conforme planejado. Isto seria representado pela curva c acompanhando a curva a, mas desviando significativamente da curva b.

Existem três causas pelas quais o projeto não progride conforme o planejado, que são:

- 1 - mudanças no escopo do trabalho
- 2 - desvios nas quantidades
- 3 - desvios na produtividade

Mudanças no escopo do trabalho são redefinições dos requisitos originais, que podem ser causados por desde uma mudança nos procedimentos do usuário, até uma melhor alternativa de projeto.

Desvios nas quantidades são causados por erros no processo de quantificação.

Desvios na produtividade são causados por não se conseguir executar o trabalho no custo unitário orçado.

É importante que o GP note bem estes 3 tipos de desvios. Se o trabalho não está evoluindo conforme planejado por causa de baixa produtividade, pode-se exercer pressão para aumentar a produtividade. Por outro lado, exercer pressão quando a produtividade alcança ou excede os valores unitários planejados pode ser contraprodutivo. Os programadores e analistas devem ser recompensados por exceder a produtividade planejada, ainda que o trabalho não esteja progredindo por outras razões. Não o fazendo, o GP pode introduzir problemas de produtividade onde antes não os havia.

4.4.2.5 - CONTROLE DE MUDANÇAS

O controle de mudanças consiste em manter a linha base do projeto atualizada . Esta linha base pode ser afetada de duas formas : a primeira por uma mudança do escopo do projeto solicitada pelo cliente e a segunda por um desvio da quantidade .

Deve-se manter documentadas estas mudanças para futuras referências .

É importante manter a linha base atualizada , pois caso contrário a percentagem completada não estaria correta , influenciando todos os cálculos anteriormente explicados .

4.4.2.6 - ESTIMATIVAS

Existem três tipos principais de estimativas .

O primeiro tipo de estimativas são as feitas na fase inicial do projeto , onde são estimados principalmente os recursos necessários para o projeto e a duração do projeto . Este tipo de estimativa não foi contemplado no modelo proposto; entretanto , com a utilização da ferramenta que implementa o modelo proposto , após alguns projetos executados teremos todos os dados históricos necessários para estimarmos novos projetos ou calibrarmos uma ferramenta automática para estimativa de projetos , como a proposta e implementada por Manoel Neto [NET90] .

O segundo tipo de estimativas são também realizadas na fase inicial do projeto e consistem em efetuar a EST e estimar os recursos e duração de cada atividade , em seguida sequenciar o trabalho e proceder ao nivelamento dos recursos necessários, obtendo as datas de início e término de cada atividade.

A obtenção destas estimativas é um processo altamente iterativo, sendo indispensável a utilização de uma ferramenta para cálculo de redes de PERT . Como existem várias ferramentas no mercado que fazem o cálculo de redes de PERT, este cálculo não foi implementado no protótipo. Como temos todos os dados necessários, bastará gerar um arquivo com estes dados e submeter este arquivo a uma das ferramentas para cálculo de redes de PERT .

O terceiro tipo de estimativas são as efetuadas durante a execução do projeto de engenharia de software . Elas são baseadas nas informações obtidas das atividades já terminadas, podendo-se calcular para cada tipo diferente de atividade , quais as médias e desvios padrões entre o estimado e o real da duração e dos recursos necessários. Com estas informações o GP pode reestimar as atividades do mesmo tipo que ainda não foram terminadas , obtendo novas estimativas, provavelmente mais reais, dos custos e duração das atividades/projeto. Mais uma vez será necessário utilizar uma das ferramentas para cálculo da rede de PERT . Este procedimento também não foi automatizado na ferramenta, podendo entretanto ser facilmente implementado .

O próximo capítulo detalha a implementação do protótipo da ferramenta automatizada que suporta o modelo de gerenciamento de projetos de engenharia de software proposto .

CAPÍTULO 5 - IMPLEMENTAÇÃO DO PROTÓTIPO

5.1 - Considerações Iniciais

O protótipo de uma ferramenta automatizada para o GEPES que implementamos, contempla as funções de planejamento e execução de projetos, já detalhadas nos itens 5.4.1.1 a 5.4.2.4. As funções referentes aos itens 5.4.2.5 - Controle de Mudanças, e 5.4.2.6 - Estimativas, não foram implementadas. Conforme dito anteriormente, estas funções podem ser implementadas através da geração de um arquivo com todos os dados relevantes, e submetendo-se este arquivo a uma das várias ferramentas para análise de redes de PERT existentes no mercado.

Para que o leitor possa melhor entender como foi implementado o protótipo, é necessário algum conhecimento sobre o SIPS e o MRO. Tentamos nos restringir ao mínimo necessário e fornecemos este conhecimento nos dois itens seguintes, antes de realmente detalharmos a implementação do protótipo.

5.2 - O SIPS

5.2.1 - Considerações Iniciais

Para a implementação do protótipo, optamos pela execução do mesmo no SIPS. Abaixo apresentaremos uma descrição do que é o SIPS, e de várias razões técnicas que nos levaram a optar pelo SIPS e que ficarão claras na descrição do mesmo.

A razão principal que levou-nos a utilizar o SIPS é a de que ele é uma excelente plataforma para experimentação de metodologias, tendo sua utilização agilizado em muito a implementação da ferramenta para o GEPES, pois poupou-nos todo o trabalho de projetar uma base de dados e codificar as rotinas necessárias para o manuseio das informações contidas na base de dados.

A descrição do SIPS aqui apresentada baseia-se na nossa pequena experiência com o mesmo, e nas informações contidas nos documentos [LOP89], [CAP88] e [JIN88].

5.2.2 - A Arquitetura do SIPS

O SIPS - Sistema Integrado para Produção de Software, em desenvolvimento no IA-CTI Instituto de Automação do Centro Tecnológico para Informática em Campinas-SP é um sistema expansível para apoio à produção de software. Seu objetivo é dar suporte ao desenvolvimento integrado de software, abrangendo todas as fases do ciclo de vida, através de ferramentas automatizadas que contribuam para uma maior agilidade no processo de desenvolvimento de sistemas, aumento de produtividade e melhor qualidade do software produzido.

A filosofia de desenvolvimento do SIPS consiste na construção de ferramentas independentes entre si, porém com funções concatenadas, de maneira a atender as necessidades de desenvolvimento de um projeto. A integração entre ferramentas é conseguida através da padronização das informações armazenadas e manipuladas pelas ferramentas, as quais se apóiam em um sistema de gerenciamento de bases de dados especialmente construído. Além disso, todas as ferramentas empregam uma mesma interface de comunicação com o usuário, mantendo assim a coerência entre os comandos das várias ferramentas, o que facilita a interação com o usuário.

A existência do núcleo comum de gerenciamento de informações para todas as ferramentas e a interface única de comunicação com o usuário fazem do SIPS um sistema integrado para a produção de software; ao mesmo tempo, a independência entre as ferramentas permite a incorporação gradativa de novas ferramentas, tornando-o expansível e capaz de atender novas necessidades de produção de software.

Para que uma metodologia seja automatizada, ela deve ser representada na ferramenta que a automatiza e, para isso, ela deve ser formalizada. O processo de formalização de uma metodologia impõe uma definição rígida do escopo de abrangência dos casos tratados pela metodologia, bem como da maneira como ela é utilizada. No SIPS, metodologias podem ainda ser integradas entre si para propiciar o suporte integrado às várias fases do processo de produção do software.

O SIPS pode ser considerado como sendo composto de dois subsistemas : **SIPS-META** , que permite a definição de Linguagens de Descrição de Metodologias pelo especialista em metodologias ; e **SIPS-SISTEMA** , que permite o uso do SIPS pelos projetistas , empregando as metodologias já integradas para o desenvolvimento de sistemas de software , aqui denominados alvos .

O SIPS armazena todas as informações do sistema alvo em uma única base de dados , usando o Modelo de Representação de Objetos MRO vide [TRAB6] e [TRAB8] , independentemente das metodologias empregadas para descrever estes sistemas . Esta base de dados é denominada Base Sistema .

As descrições das metodologias são mantidas em uma outra base de dados , a **BASE META** , cujo conteúdo define como a Base Sistema deve ser usada . A Base Meta é inicializada e mantida pelos especialistas em metodologias . Estes especialistas analisam as metodologias a serem integradas ao SIPS , descrevendo-as com uma Meta Linguagem específica do SIPS . A descrição é interpretada pelo interpretador da Meta Linguagem , que armazena essas informações na Base Meta .

A Figura 15 mostra a arquitetura do SIPS , onde aparecem os vários componentes do ambiente . As ferramentas desenvolvidas para o SIPS são integradas ao ambiente . Qualquer conjunto de ferramentas desenvolvidas para o SIPS pode ter a aparência de um ambiente integrado , obtida através do supervisor . O supervisor é responsável pelo monitoramento e pela ativação das ferramentas com as quais o usuário interage . As ferramentas acessam apenas no modo leitura a Base Meta , a qual contém a definição de como a Base Sistema deve ser usada ; a Base Sistema é acessada para armazenar descrições de sistemas alvo . Os acessos às bases de dados do SIPS são feitos através de primitivas que constituem o Gerenciador da Base Meta e o Gerenciador da Base Sistema . Esses acessos às bases de dados são monitorados pelo Gerenciador de Projeto e Configuração de Software . As ferramentas possuem uma interface padrão de comunicação com o usuário , que é obtida através das primitivas que constituem o Gerenciador da Interface Homem/Máquina .

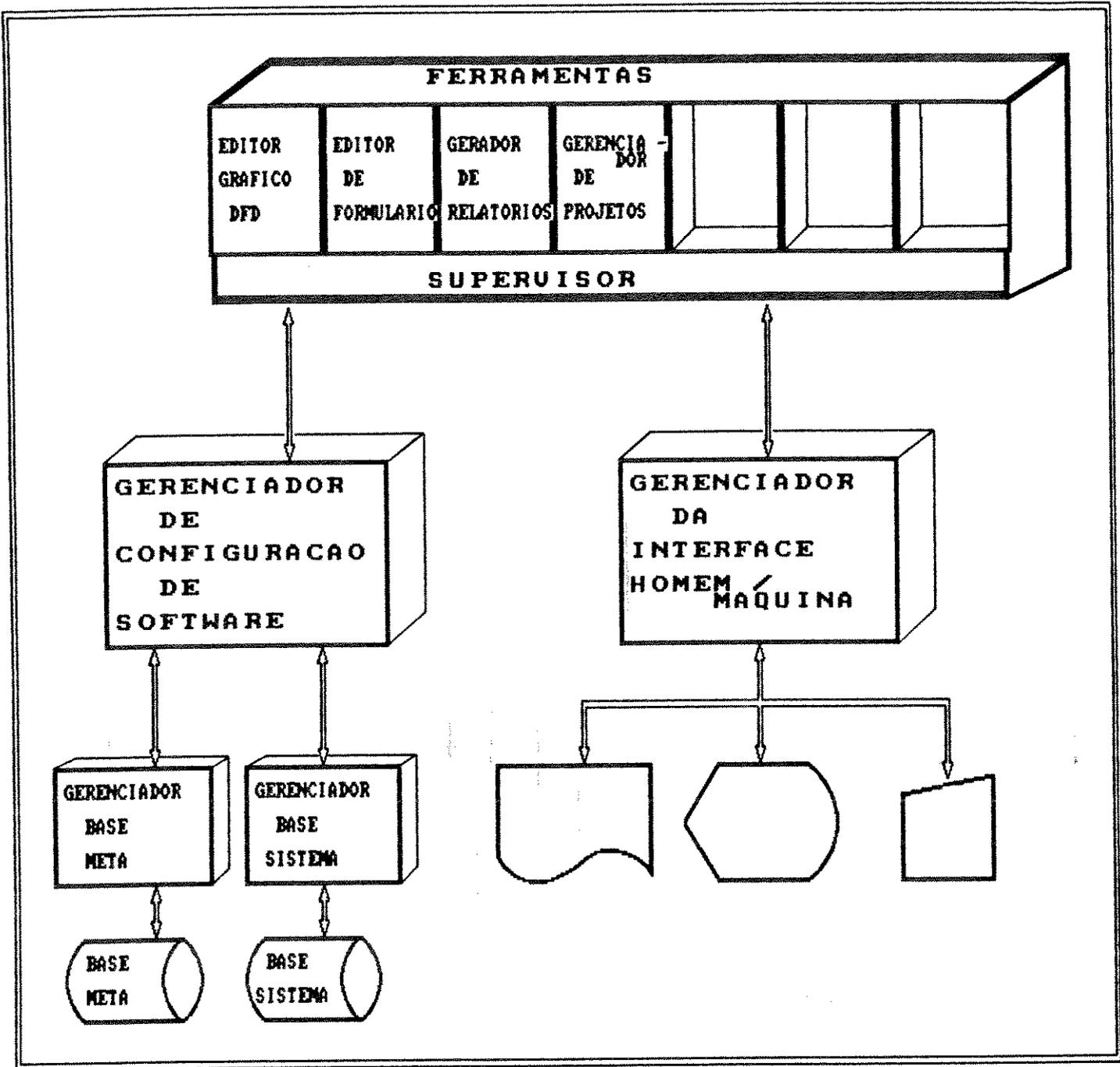


Fig. 15 - Arquitetura do SIPS

5.2.3 - Classes do Sips

O núcleo do SIPS consiste de primitivas que permitem às ferramentas acessar as Bases Meta e Sistema . Uma ferramenta pode somente ler os dados da Base Meta , uma vez que ela contém as informações que descrevem a metodologia que será usada para descrever o sistema alvo . A Base Sistema pode ser acessada pelas ferramentas para leitura , inclusão ou alteração de dados do sistema alvo em desenvolvimento .

Quando se procura acessar um objeto armazenado na Base Sistema , se for conhecido o seu código (identificador interno) então o acesso é imediato , uma vez que o código identifica unívocamente um objeto entre todos os objetos da Base Sistema . No entanto , quando a identificação é feita pelo nome (ou por um sinônimo) essa identificação não é unívoca , uma vez que pode existir mais de um objeto com um mesmo nome . Para que se possa identificar um objeto pelo seu nome unívocamente , é necessário que se especifique o contexto onde este nome é definido . Essa especificação é feita através da partição dos identificadores existentes na Base Sistema em classes , de maneira que um nome esteja definido somente no domínio de uma classe .

A identificação de objetos pelo seu código é normalmente efetuada pelas ferramentas ; porém , o usuário nunca tem acesso direto a esses códigos , sendo que a sua interação com o SIPS sempre se dá através dos nomes e sinônimos dos objetos envolvidos . Assim , é desejável que o modelo de dados utilizado pelo seu gerenciador de base de dados esteja bastante próximo do que o usuário "sente" que seja a informação que a Base de Dados manipula . Pode-se dizer que um sistema , quando visto de maneira global , apresenta um pequeno grau de detalhe . Quando o interesse recai em um de seus componentes , o grau de detalhe aumenta , porém toda a informação que então se torna disponível refere-se diretamente àquele componente que está sendo observado . Pode-se então pensar que quando uma ferramenta efetua uma comunicação com o usuário , ela realiza a interpretação dos identificadores dos objetos envolvidos , levando em consideração somente o conjunto de informações que está sendo examinado , e exigindo que qualquer referência a outras informações tenha um teor de identificação maior .

Pode-se dotar o modelo de dados do SIPS dessa capacidade considerando que , quando se manipula a base de dados de um ponto de vista *global* , são poucos os objetos acessíveis . Indicando-se ao sistema que se pretende considerar a seguir um desses objetos , indica-se uma região na base de dados em que se pretende obter ou manipular informações . Dessa forma tornam-se acessíveis novos objetos . Esse procedimento pode ser repetido , indicando-se para consideração um dos novos objetos tornados disponíveis , e dessa forma aumentando-se sucessivamente o nível de detalhe e , correspondentemente , a quantidade de objetos a que se tem acesso diretamente através de seus nomes ou sinônimos .

Cada vez que se coloca em consideração um novo objeto , aumenta-se um nível de detalhe ; porém , não são todos os objetos desse nível que se tornam disponíveis , mas apenas aqueles que detalham mais o objeto considerado .

O modelo de dados do SIPS pode então contar com essa capacidade usando-se os conceitos de **Classe de Objetos** , **Objeto Considerado** e **Classe Hospedeira** .

Uma das classes terá de estar permanentemente disponível e será denominada **Classe Global** . Os objetos da Classe Global são os que estão disponíveis quando se está no nível menos detalhado de acesso ao sistema e permanecem sempre disponíveis . Correspondentemente existe uma Classe Hospedeira Global , a qual contém os tipos de objetos que existem na Classe Global . Dessa forma , todos os objetos dos tipos que pertencem à classe hospedeira global estarão permanentemente disponíveis .

Portanto o conceito de classes permite que os objetos de uma base de dados do SIPS passem a ser acessíveis em blocos , sendo que cada bloco refere-se não somente a um nível específico de detalhe de informação , mas também a um determinado aspecto de informação .

Para isso , os objetos de uma base de dados devem ser classificados segundo diferentes aspectos , sendo essa classificação feita usando os diferentes tipos de objetos definidos na Base Meta . Dessa forma , cada objeto pertence a apenas um bloco acessível , que é chamado **Classe de Objetos** , que é determinada através do seu tipo de objeto .

Para a existência de Classes de Objetos , é necessário que se divida os tipos de objetos contidos na Base Meta nos vários aspectos de interesse . Pode-se considerar então que o conceito de Classes de Objetos estabelece uma partição no conjunto de tipos de objetos da Base Meta , sendo cada partição denominada **Classe Hospedeira** . Os objetos da Base Sistema estarão também particionados segundo os seus tipos , em uma correspondência com a partição que as classes hospedeiras estabelecem sobre os tipos de objetos da Base Meta .

A Figura 16 ilustra o conceito de classes descrito acima . A classe hospedeira **global** , neste exemplo , é composta de *PAÍSES* , que por sua vez são compostos dentre outros objetos , de *ESTADOS* , que são compostos por *Cidades* . Os *PAÍSES* estão sempre acessíveis , e não existem dois *PAÍSES* com o mesmo nome . Dentro da classe hospedeira global , existe a classe hospedeira *PAÍSES* que contém os vários *ESTADOS* ; e dentro da classe hospedeira *PAÍSES* existe a classe hospedeira *ESTADOS* , que contém as várias *Cidades* . Um *PAÍS* não possui dois *ESTADOS* com o mesmo nome , mas pode haver dentro do global , mais de um *ESTADO* com o mesmo nome , mas em *PAÍSES* diferentes . Assim também , um *ESTADO* não possui duas *Cidades* com o mesmo nome , mas podem haver *Cidades* com o mesmo nome , desde que em *ESTADOS* diferentes . Então , ao identificarmos o contexto de um determinado *PAÍS* e de um determinado *ESTADO* , quando estivermos referenciando uma *Cidade* , esta pertencerá ao contexto especificado .

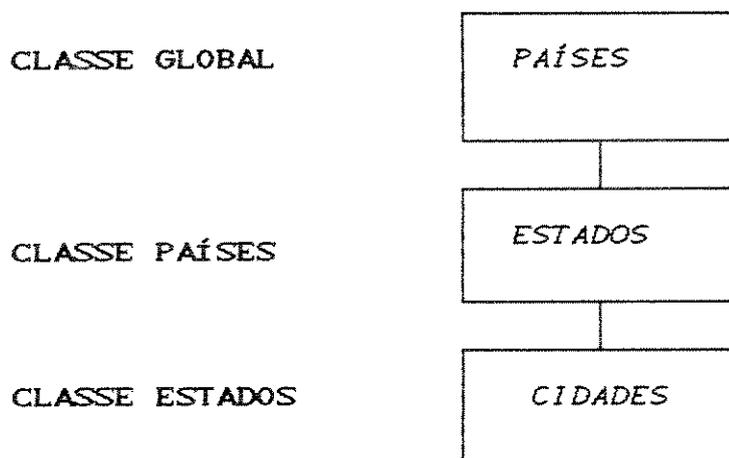


Fig. 16 - Exemplo de Classes do SIPS

5.2.4 - O MRO (Modelo de Representação de Objetos)

O modelo de dados em que o SGBD do SIPS se apóia é o MRO , que procura modelar as situações do mundo real, representando as características desejáveis dos objetos que existem no mundo real .

A fundamentação matemática do MRO é a teoria dos grafos , por ser mais adequada para modelos de dados voltados à representação semântica intensa . Nesse modelo , os nós de um dígrafo representam os objetos que existem no mundo real e os arcos representam as associações que existem entre eles , as quais são denominadas relacionamentos . Os objetos são identificados através de nomes e são representados através de códigos (identificadores internos) .

A categorização dos nós de um dígrafo classifica os objetos através de um tipo que é atribuído a cada objeto . O mesmo é feito quanto aos relacionamentos entre os objetos , aos quais se atribuem igualmente tipos de relacionamentos . Assim , os objetos e os relacionamentos no MRO são classificados segundo seus respectivos tipos .

Cada relacionamento no MRO indica que os objetos estão associados um ao outro , ou seja , no dígrafo correspondente devem existir dois arcos , um em cada sentido . Cada arco representa um relacionamento oposto ao representado por seu par ; no MRO todo relacionamento tem o seu oposto correspondente .

Representam-se também as *propriedades* dos objetos que é uma denominação genérica dada aos tipos abstratos **comentário** , **atributo** e **sinônimo** (nome alternativo de objetos) , bem como os atributos de relacionamentos .

Aos objetos , além das propriedades , podem ainda ser associados alguns atributos comuns a todos os objetos da base de dados , denominados *atributos intrínsecos* . Como atributos intrínsecos de um objeto podemos ter , por exemplo , o nome do objeto e a data de sua criação .

A Figura 17 apresenta um exemplo de MRO para o caso de um escritor que escreve livros.

Nesta figura temos 2 objetos : escritor e livro.

O escritor pertence à classe *global* e possui dois atributos : nome e endereço .

O livro pertence à classe *livros* e possui também dois atributos : título e editora.

O escritor e o livro estão relacionados pelos relacionamentos *escreve/e-escrito*.

A seta no relacionamento indica o sentido principal,

ou seja : escritor escreve livro

e o relacionamento oposto é

livro e-escrito escritor

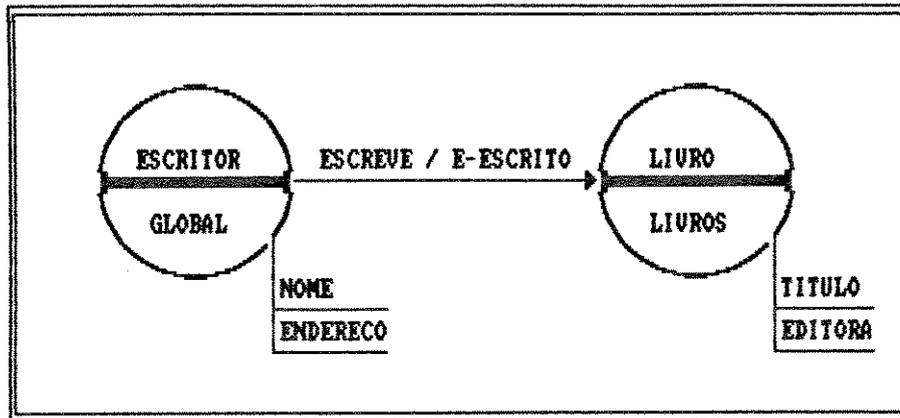


Fig. 17 - Exemplo de MRO

5.2.5 - LINGUAGEM DE DESCRIÇÃO DA BASE META

A incorporação de uma metodologia no SIPS é feita por um especialista da metodologia, que tenha dela um profundo conhecimento e, além disso, que tenha bom conhecimento de como essa tarefa é realizada com o SIPS.

Para que uma metodologia possa ser suportada pelo SIPS, esta deve ser modelada no MRO e essa modelagem submetida a um interpretador de linguagem de descrição de metodologias do SIPS, denominado Interpretador Meta. A descrição obtida deste interpretador é armazenada na Base Meta, como já dissemos anteriormente.

O processo de modelagem de uma metodologia pode ser resumidamente descrito pelo seguinte procedimento, que compreende 9 passos:

- 1 - Identificar as palavras chaves da metodologia. Para isso, poder-se-ia analisar um texto que descreva a metodologia e dele extrair todas as palavras que se referem a conceitos da metodologia;
- 2 - Separar essas palavras chaves em tipos de objetos e tipos de associações. As palavras que representam elementos da metodologia devem ser identificadas e separadas em tipos de objetos. As palavras que correspondem a associações são as que identificam propriedades de objetos e relacionamentos entre objetos;
- 3 - Analisar os tipos de associações. Devem ser classificados em: tipos de relacionamentos - aquelas que associam dois ou mais objetos; tipos de atributos - aquelas que se referem a valores que se associam a tipos de objetos ou a tipos de relacionamentos; tipos de sinônimos - aquelas que se referem a tipos de identificadores alternativos para um mesmo objeto e tipos de comentários - aquelas que se referem a tipos de textos que podem ser associados a objetos;
- 4 - Identificar as associações entre objetos e relacionamentos. A partir desse passo, deve ser identificado como os vários tipos podem se associar, armazenando essas informações em tabelas. Neste passo, identificam-se os tipos de relacionamentos que podem ser associados a cada tipo de objeto;

- 5 - Identificar os tipos de relacionamentos opostos a cada relacionamento . Como todo tipo de relacionamento tem um tipo de relacionamento oposto , este tem que ser identificado . A seguir , completam-se as tabelas de quais relacionamentos são associados a cada objeto , usando a confrontação com os relacionamentos opostos ;
- 6 - Identificar as associações entre atributos e objetos e entre atributos e relacionamentos ;
- 7 - Identificar as associações entre objetos e comentários e entre objetos e sinônimos . Neste ponto teríamos todas as informações necessárias para a modelagem utilizando o MRO , resultando na figura 11 já apresentada ;
- 8 - Preparar a descrição da modelagem obtida , na Linguagem de Descrição de Metodologias do SIPS . Esta linguagem informa ao Interpretador Meta , quais os objetos da metodologia , os relacionamentos , os atributos e seus interrelacionamentos .

Para nosso exemplo do escritor/livro , esta linguagem teria o seguinte aspecto :

```

>classes
  global , livros
>objetos
  escritor , global
  livro , livros
>relacionamentos
  escreve op e-escrito
>atributos
  nome
  endereco
  titulo
  editora
>sintaxe-rel
  objeto escritor
    escreve livro
  objeto livro
    e-escrito escritor
>sintaxe-atr
  objeto escritor
    nome
    endereco
  objeto livro
    titulo
    editora
>fim-nomes

```

9 - Submeter esta descrição ao Interpretador Meta para que ela possa ser armazenada na Base Meta ; a descrição armazenada será usada a partir de então para a descrição dos sistemas do usuário .

5.2.6 - FORMULÁRIOS PARA ENTRADA DE DADOS

Com relação às ferramentas , dois tipos podem ser gerados dentro do ambiente SIPS : ferramentas genéricas e ferramentas específicas.

Ferramentas específicas são construídas para automatizar metodologias específicas . Deste tipo é o gerenciamento de projetos que implementamos . Ferramentas genéricas atuam sob uma descrição meta que descreve como as ocorrências do modelo meta da ferramenta estão associados ao modelo meta da metodologia . Estas informações estão armazenadas na Base Meta .

Definir um formulário compreende : modelar o formulário com base em um **Modelo Geral de Formulários** e em seguida descrever essa modelagem utilizando o Editor de Descrições , que é outra ferramenta genérica do SIPS .

A descrição do Modelo Geral de Formulários estará armazenada na Base Meta do SIPS , e a definição do formulário com base no modelo será armazenada na Base Sistema do SIPS . Sendo assim , o especialista de metodologias cada vez que quiser inserir um novo tipo de formulário no SIPS deverá inicialmente modelar este formulário através do Modelo e depois inserir esta modelagem através do Editor de Descrições . Esta inserção é um pouco trabalhosa quando os formulários são complexos ou em grande número. Porém uma vez inserida a modelagem do formulário na Base Sistema , a utilização do Editor de Formulários é muito simples e eficiente .

Um formulário é identificado pelo tipo de objeto da metodologia que ele representa e pode consistir dos seguintes elementos :

- Bloco;
- Condição;
- Item Elementar ;
- Item de Menu;
- Rodapé;
- Cabeçalho .

Os principais elementos são :

Bloco

Um bloco é um tipo abstrato de campo que pode ser precedido por um texto e deve consistir de outros blocos ou de itens elementares. A presença de um bloco no formulário pode ser condicionada a uma determinada condição , e além disso , um bloco pode se repetir em um formulário , com base em um indicador de repetição .

Item Elementar

Um item elementar caracteriza o tipo mais elementar de campo que pode aparecer em um formulário . O item elementar pode fazer parte de um bloco ou não , pode ser preenchido diretamente ou através de seleção sobre uma lista de itens pré-estabelecidos , ou apenas o conteúdo do item aparece na tela e o item não fica habilitado para escrita . O conteúdo do item elementar pode ser um atributo, comentário, sinônimo , e relacionamento do objeto a que o formulário se refere ; pode determinar o tipo que outro item vai ser , como por exemplo o tipo de um objeto destino de um relacionamento , ou insere uma informação (comentário, atributo, sinônimo, relacionamento) referente a outro objeto que não é o objeto a que o formulário se refere , mas que está definido em outro item .

São definidos para o item elementar : tamanho e posição no formulário , texto precedente e se o item ocorre uma única vez ou se se repete .

Quando o preenchimento de um item elementar deve ser feito através de seleção sobre uma lista de itens pré-estabelecidos , estes itens são definidos como itens de escolha . O uso de itens de escolha se restringe ao caso em que o conteúdo do item elementar é um atributo do objeto à que o formulário se refere ou o conteúdo determina o tipo de uma palavra especial .

Ítem de Menu

Os itens de menu correspondem às opções de menu do formulário , através dos quais serão selecionados os campos do formulário para edição . Cada item elementar ou bloco está associado a um item de menu correspondente, através do qual , via seleção , o campo poderá ser acessado .

São definidos para o item de menu : o texto que aparecerá no menu, a posição no formulário e o texto help que esclarece o significado do campo .

Para fins de clareza, suponhamos que desejamos criar um formulário para informar os livros que o escritor de nosso exemplo anterior escreveu, baseados no exemplo de MRO e da listagem de descrição que apresentamos.

O formulário que chamaremos de FORMULARIO-EXEMPLO poderia ter a forma

Nome do escritor
LIVROS ESCRITOS
Titulo do livro
Editora

Fig. 18 - Modelo de um Formulário

Ele será composto de um item elementar que será o nome do escritor (item-elementar-1) e de um bloco (bloco-1) que conterà dois outros itens elementares, o titulo do livro (item-elementar-2) e a editora (item-elementar-3) e que poderá ter repetições, pois um escritor escreve normalmente mais de um único livro.

Cada um destes itens possuirá um item de menu, que possibilitará o acesso ao item específico. Teremos assim :

- item-menu-1 para o nome do escritor,
- item-menu-2 para o titulo do livro,
- item-menu-3 para a editora e um
- item-menu-bloco-1 para o bloco de títulos e editoras.

Um item elementar especial é o 10 , que é o nome do autor, ao qual os outros dois itens estão relacionados. Este item especial será o objeto-corrente ao qual referir-se-ão os outros itens.

Finalmente a listagem de descrição do formulário terá o aspecto apresentado na Fig. 19 .

```

FORMULARIO-EXEMPLO
*-----*
ITEM-ELEMENTAR-1          (ITEM-ELEMENTAR)
*-----*
--- Relacionamentos do Objeto ---
ACESSADO-PELO
    item-menu-1
CONSISTE
    formulario-exemplo
--- Atributos do Objeto ---
TIPO-CAMPO
    objeto-corrente
--- Comentarios do Objeto ---
NOME-TIPO
    nome
TEXTO-PRECEDENTE
    Nome do Escritor

*-----*
bloco-1                    (BLOCO)
*-----*
--- Relacionamentos do Objeto ---
ACESSADO-PELO
    item-menu-bloco-1
CONSISTE
    formulario-exemplo
CONSISTE-DE
    item-elementar-2
    item-elementar-3

--- Atributos do Objeto ---
REPETICAO
    sim
--- Comentarios do Objeto ---
TEXTO-PRECEDENTE
    LIVROS ESCRITOS

```

Figura 19 - Listagem de Descrição do Formulário

```

*-----*
item-elementar-2          (ITEM-ELEMENTAR)
*-----*
--- Relacionamentos do Objeto ---
ACESSADO-PELO
    item-menu-2
CONSISTE
    bloco-1
FAZ-REFERENCIA-A
    item-elementar-1
--- Atributos do Objeto ---
TIPO-CAMPO
    relacionamento
--- Comentarios do Objeto ---
NOME-TIPO
    escreve
TEXTO-PRECEDENTE
    Titulo do Livro

*-----*
item-elementar-3          (ITEM-ELEMENTAR)
*-----*
--- Relacionamentos do Objeto ---
ACESSADO-PELO
    item-menu-3
CONSISTE
    bloco-1
FAZ-REFERENCIA-A
    item-elementar-2
--- Atributos do Objeto ---
TIPO-CAMPO
    atributo
--- Comentarios do Objeto ---
NOME-TIPO
    editora
TEXTO-PRECEDENTE
    Editora

*-----*
item-menu-1              (ITEM-DE-MENU)
*-----*
AGESSA
    item-elementar-1
TEXTO-DE-MENU
    Nome do Editor :

```

Figura 19 - Listagem de Descrição do Formulário (continuação)

```

*-----*
item-menu-2                (ITEM-DE-MENU)
*-----*
ACESSA
    item-elementar-2
TEXTO-DE-MENU
    Titulo do Livro :
*-----*
item-menu-3                (ITEM-DE-MENU)
*-----*
ACESSA
    item-elementar-3
TEXTO-DE-MENU
    Editora :

*-----*
item-menu-bloco-1         (ITEM-DE-MENU)
*-----*
ACESSA
    bloco-1
TEXTO-DE-MENU
    LIVROS ESCRITOS

```

Figura 19 - Listagem de Descrição do Formulário (continuação)

5.3 - Classes do Protótipo

Conforme descrito no item 6.2.3 , o SIPS permite a utilização de diferentes classes de objetos.

Optamos pela utilização de 3 classes : a classe **global** com o objeto *ambiente-de-gerenciamento* ; a classe **gerenciamento** que possui vários objetos , dentre eles o *projeto-eng-soft* que restringe a última classe , a classe **projeto** , onde estão as *atividades* .

A razão para a escolha destas classes é a de que tendo-se uma classe global "ambiente-de-gerenciamento", podemos configurar diferentes ambientes de gerenciamento. Como cada ambiente possui uma classe *gerenciamento* onde estão os projetos, é possível termos projetos de mesmo nome, porém em ambientes diferentes. E a razão de escolhermos uma terceira classe, a classe *projeto*, é a de que isto possibilita que projetos diferentes possuam atividades com o mesmo nome, uma situação bastante comum no gerenciamento de vários projetos.

A Figura 20 ilustra o conceito de classes descrito acima. A classe hospedeira **global** , neste exemplo , é composta de *ambientes de gerenciamento* , que por sua vez são compostos dentre outros objetos , de *projetos* , que são compostos por *atividades* . Os ambientes-de-gerenciamento estão sempre acessíveis , e não existem dois ambientes com o mesmo nome . Dentro da classe hospedeira global , existe a classe hospedeira **gerenciamento** que contém os vários *projetos* ; e dentro da classe hospedeira gerenciamento existe a classe hospedeira **projeto** , que contém as várias *atividades* . Um ambiente não possui dois projetos com o mesmo nome , mas pode haver dentro do global , mais de um projeto com o mesmo nome , mas em ambientes diferentes . Assim também , um projeto não possui duas atividades com o mesmo nome , mas pode haver atividades com o mesmo nome , desde que em projetos diferente. Então, ao identificarmos o contexto de um determinado ambiente e de um determinado projeto , quando estivermos referenciando uma atividade , esta pertencerá ao contexto especificado .

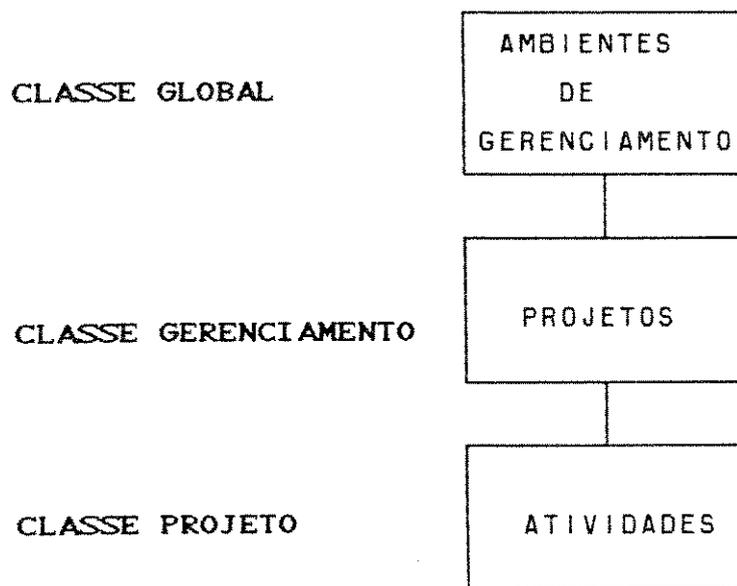


Fig. 20 - Classes do Protótipo

5.5 - Descrição da Base Meta do Protótipo

Conforme dito no item 5.2.5, para que uma metodologia seja suportada pelo SIPS, esta deve ser modelada no MRO, conforme feito no item 6.4 e esta modelagem submetida a um interpretador de linguagem de descrição de metodologias do SIPS.

No anexo 1 está a linguagem de descrição do modelo proposto.

5.6 - Formulários Utilizados no Protótipo

No protótipo , optamos pela utilização de uma ferramenta genérica, no caso o Editor de Formulários , para a entrada de todos os dados da classe gerenciamento , que são os dados dos ambientes de gerenciamento , ou seja , os recursos humanos e materiais , os ciclos de vida e as metodologias , através da definição de formulários específicos para cada conjunto de dados .

A razão que nos levou a optar pela utilização dos formulários para o fornecimento dos dados do ambiente de gerenciamento, é a de que a configuração do ambiente não é um processo tão frequente e iterativo como o planejamento e a execução dos projetos, além do volume de dados ser muito menor . Por exemplo, para fornecermos um novo tipo de recurso humano, basta selecionar o formulário correspondente e informar o tipo de recurso e seu custo unitário.

Os formulários só não se aplicam às informações que precisam ser cruzadas , como por exemplo ao informarmos a sequencia de execução das atividades, necessitamos verificar se as antecessoras e sucessoras existem, e se os seus tipos de atividades são compatíveis com as relações de precedencias entre tipos de atividades.

Os formulários criados para utilização em nosso protótipo foram quatro :

- Ambiente de Gerenciamento ;
- Categorias de Recursos Humanos ;
- Tipos de Ciclos de Vida ;
- Fases .

As figuras a seguir reproduzem a tela do sistema quando são apresentados os formulários para edição (preenchimento) . Nas figuras ficam claros os itens que fazem parte de cada formulário.

A listagem do anexo 2 foi obtida com o uso de outra ferramenta de uso genérico do SIPS , o Listador de Descrições. Esta listagem reproduz as informações inseridas na Base Sistema com o Editor de Descrições para cada um dos formulários criados, e tem a forma do exemplo apresentado no item 5.2.6.

S I P S	SUPER AMBIENTE	Ver: 2.11
AMBIENTE DE GERENCIAMENTO DESCRICAO DO AMBIENTE ***** RECURSOS MATERIAIS *** CATEGORIA REC.MATERIAL DESCRICAO REC.MATERIAL CUSTO UNITARIO REC.MAT. DISPONIBILIDADE REC.MAT. ***** RECURSOS HUMANOS ***** CATEGORIA REC.HUMANO CUSTO UNITARIO REC.HUM. ***** CICLOS DE VIDA ***** TIPO DE CICLO DE VIDA DESCRICAO CICLO DE VIDA ***** PESSOAS ***** PESSOA DISPONIBILIDADE POR PERIODO ALOCACAO INICIO DE FERIAS		AMBIENTE SUPER AMBIENTE
EDITA AMBIENTE-DE-GERENCIAMENTO		PREENCHE

Fig. 22 - Formulário "AMBIENTE-DE-GERENCIAMENTO"

S I P S	SUPER AMBIENTE	Ver: 2.11
CATEGORIA REC. HUMANO CUSTO UNITARIO REC.HUM. ***** PESSOAS NESTA CATEGORIA PESSOA DISPONIBILIDADE POR PERIODO ALOCACAO INICIO DAS FERIAS DURACAO DAS FERIAS NOME DO GERENTE		RECURSOS HUMANOS Analistas
EDITA CATEGORIAS-RECURSOS-HUMANOS		PREENCHE

Fig. 23 - Formulário "CATEGORIAS-RECURSOS-HUMANOS"

S I P S	SUPER AMBIENTE	Jan. 2.11
<div style="border: 1px solid black; padding: 5px;"> <p>TIPO-DE-CICLO-DE-VIDA</p> <p>DESCRICAO DO CICLO DE VIDA</p> <p>***** FASES DO CICLO DE VIDA</p> <p>FASE</p> <p>DESCRICAO DE FASE</p> <p>PRODUTO DA FASE</p> <p>FASE DE QUE FAZ PARTE</p> <p>METODOLOGIA ASSOCIADA</p> </div>		TIPO-DE-VIDA Classico
<p>EDITA TIPO-DE-CICLO-DE-VIDA PREENCHE</p>		

Fig. 24 - Formulário "TIPO-DE-CICLO-DE-VIDA"

S I P S	SUPER AMBIENTE	Jan. 2.11
<div style="border: 1px solid black; padding: 5px;"> <p>FASES</p> <p>DESCRICAO DA FASE</p> <p>PRODUTO DA FASE</p> <p>FASE DE QUE FAZ PARTE</p> <p>TIPO ATIVIDADES DA FASE ****</p> <p>TIPOS ATIVIDADES DA FASE</p> <p>TIPO ATIVIDADE PREDECESSORA</p> <p>TIPO ATIVIDADE SUCESSORA</p> <p>METODOLOGIA ASSOCIADA</p> <p>TIPO OBJETO ASSOCIADO</p> </div>		FASES analise
<p>DD-GERENCIA EDITA FASES PREENCHE</p>		

Fig. 25 - Formulário "FASES"

5.7 - ARQUITETURA DO PROTÓTIPO

Para implementação da ferramenta de uso específico para o gerenciamento de projetos, optamos pela arquitetura apresentada na Figura 25.

Nesta figura cada linha corresponde a um nível hierárquico. Por exemplo o Módulo 0, através de subcomandos, ativa os Módulos 1,2,3,4 ou 5. Assim também o Módulo 5.2 ativa os Módulos 5.2.1 ou 5.2.2.

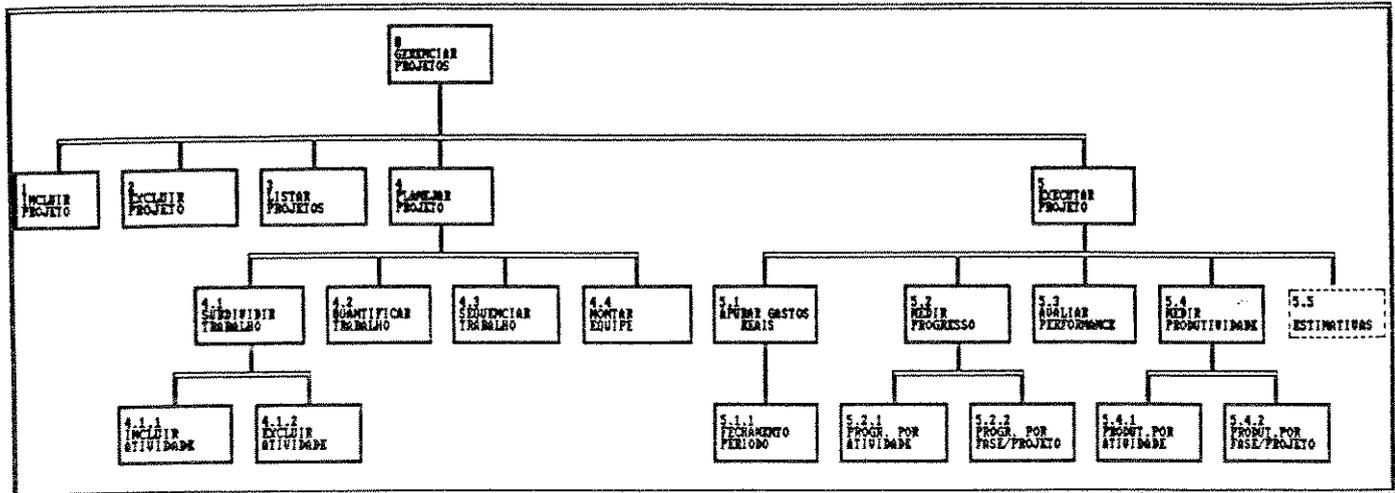


Fig. 25 - Arquitetura do Protótipo

As funções de cada um dos módulos do protótipo estão apresentadas no item 5.8 logo abaixo.

Além destes módulos, foram criadas 6 rotinas de uso geral, que não aparecem na figura 25, mas estão descritas no item 5.9.

A linguagem escolhida para codificação dos módulos foi C, em primeiro lugar porque todas as rotinas do núcleo do SIPS estão codificadas nesta linguagem, e em segundo lugar por C tratar-se de uma linguagem de propósitos gerais, com características de expressão, controle de fluxo e estruturas de dados econômicas. Sua generalidade torna-a, para muitas tarefas, mais conveniente e eficiente do que outras linguagens supostamente mais poderosas.

5.8 - ESPECIFICAÇÃO DOS MÓDULOS

Os algoritmos dos módulos do gerenciamento de projetos , já apresentados na arquitetura da ferramenta , estão descritos abaixo .

MÓDULO 0 - GERENCIAR PROJETOS

Descrição :

Este módulo inicia o gerenciamento, criando todas as tabelas que serão utilizadas pela ferramenta e ativando o módulo correspondente à opção escolhida pelo usuário.

Pseudo-Código :

```
escrever cabeçalho.
ler nome do arquivo para mensagens de erro e consistencia.
MONTA TABELA de ambientes de gerenciamento.
selecionar o ambiente de gerenciamento.
MONTA TABELA de projetos.
MONTA TABELA de ciclos de vida.
MONTA TABELA de pessoas.
MONTA TABELA de recursos humanos.
MONTA TABELA de recursos materiais.
CASO opção for
  "Incluir novo projeto" ativa MODULO 1;
  "Excluir projeto"      ativa MODULO 2;
  "Listar projetos"     ativa MODULO 3;
  "Planejar um projeto" ativa MODULO 4;
  "Executar um projeto" ativa MODULO 5;
  "Menu anterior"      RETORNA .
```

MODULO 1 - INCLUIR NOVO PROJETO

Descrição :

Este módulo procede à inclusão de um projeto que não exista na Base Sistema . Para tal ele solicita o nome do projeto , sua descrição , o tipo de ciclo de vida e qual pessoa será o gerente.

Pseudo-Código :

```
escrever cabeçalho.
ler nome do projeto para incluir.
SE projeto não EXISTE
  ler descrição,ciclo de vida , gerente;
  incluir novo projeto na base sistema;
  incluir novo projeto na tabela de projetos;
  zerar o ultimo período apurado;
  criar a classe projeto .
```

MODULO 2 - EXCLUIR PROJETO

Descrição :

Este módulo exclui um projeto da Base Sistema, sem entretanto verificar se suas atividades estão terminadas ou não.

Pseudo-Código :

```
escrever cabeçalho e ler nome do projeto para excluir.
SE projeto EXISTE
  excluir da tabela de projetos;
  excluir da base sistema .
```

MODULO 3 - LISTAR PROJETOS

Descrição :

Este módulo lista os nomes dos projetos existentes na Base Sistema.

Pseudo-Código :

escrever cabeçalho.
listar tabela de projetos.

MODULO 4 - PLANEJAR PROJETO

Descrição :

Este módulo seleciona um projeto para planejar. Baseado no tipo de ciclo de vida do projeto, monta a tabela das fases de nível hierárquico mais alto e ativa um dos módulos correspondente à opção escolhida pelo usuário.

Pseudo-Código :

escrever cabeçalho.
selecionar o projeto para planejar.
MONTA TABELA de fases nível zero.
abrir a classe projeto.
MONTA TABELA de atividades.
CASO opção seja
 "Subdividir trabalho" ativa MODULO 4.1;
 "Quantificar trabalho" ativa MODULO 4.2;
 "Sequenciar trabalho" ativa MODULO 4.3;
 "Montar equipe" ativa MODULO 4.4;
 "Menu anterior" RETORNA.

MODULO 4.1 - SUBDIVIDIR TRABALHO

Descrição :

A subdivisão do trabalho consiste em incluir, excluir ou listar atividades de um projeto. Este módulo ativa o módulo que executa uma destas tres opções.

Pseudo-Código :

escrever cabeçalho.
CASO opção seja
 "Incluir atividade" ativa MODULO 4.1.1 ;
 "Excluir atividade" ativa MODULO 4.1.2 ;
 "Listar atividades" exibe a tabela das atividades ;
 "Menu anterior" RETORNA .

MODULO 4.1.1 - INCLUIR ATIVIDADE

Descrição :

Este módulo inclui uma atividade que ainda não exista em um projeto. A atividade deve ser identificada pela rotina IDENTIFICA ATIVIDADE.

Pseudo-Código :

escrever cabeçalho .
IDENTIFICA ATIVIDADE para incluir .
SE atividade não EXISTE
 incluir na base sistema ;
 incluir na tabela de atividades .

MODULO 4.1.2 - EXCLUIR ATIVIDADE

Descrição :

Este módulo exclui uma atividade de um projeto , sem verificar se a atividade está terminada ou não .

Pseudo-Código :

escrever cabeçalho.
IDENTIFICA ATIVIDADE para excluir.
SE atividade EXISTE
 exclui da tabela de atividades;
 exclui da base sistema .

MODULO 4.2 - QUANTIFICAR TRABALHO

Descrição :

Para uma atividade identificada , este módulo lê a unidade e quantidade de unidades a serem produzidas , e para cada um dos recursos materiais e humanos , lê a quantidade necessária caso o recurso seja utilizado na atividade .

Pseudo-Código :

escrever cabeçalho.
IDENTIFICA ATIVIDADE para quantificar.
SE atividade EXISTE
 ler unidade e quantidade à produzir ;
 incluir unidade e quantidade na base sistema;
 PARA todos os recursos humanos utilizados
 ler a quantidade de hh necessários;
 incluir na base .
 PARA todos os recursos materiais necessários
 ler a quantidade ;
 incluir na base sistema .

MODULO 4.3 - SEQUENCIAR TRABALHO

Descrição :

Este módulo monta a rede de precedencias , guardando para uma atividade identificada , quais são as suas atividades antecessoras e sucessoras .

Pseudo-Código :

escrever cabeçalho .
IDENTIFICA ATIVIDADE para sequenciar .
ler duração.
incluir na base sistema.
PARA todas as atividades predecessoras
 ler nome da atividade;
 incluir na base sistema.
PARA todas atividades sucessoras
 ler nome da atividade ;
 incluir na base sistema.

MODULO 4.4 - MONTAR EQUIPE

Descrição :

Este módulo simplesmente lê o nome do coordenador da atividade .
As pessoas que trabalharão na atividade são automaticamente
alocadas por meio do tipo de recurso humano utilizado pela
atividade identificada .

Pseudo-Código :

escrever cabeçalho.
IDENTIFICA ATIVIDADE para montar equipe.
ler coordenador da atividade.
incluir na base sistema.

MODULO 5 - EXECUTAR PROJETO

Descrição :

Este módulo , semelhante ao módulo 4 , serve simplesmente para
selecionar um projeto e ativar o módulo correspondente à opção do
usuário .

Pseudo-Código :

escrever cabeçalho.
selecionar o projeto à executar.
MONTA FASES de nível zero.
abrir a classe projeto.
MONTA TABELA de atividades.
CASO opção seja
 "Apurar gastos reais" ativa MODULO 5.1 ;
 "Medir progresso" ativa MODULO 5.2 ;
 "Avaliar performance" ativa MODULO 5.3 ;
 "Medir produtividade" ativa MODULO 5.4 ;
 "Estimativas" ativa MODULO 5.5 ;
 "Menu anterior" RETORNA .

MODULO 5.1 - APURAR GASTOS REAIS

Descrição :

Este módulo lê o número de unidades de tempo do período a que se
refere a apuração e para cada uma das atividades identificadas ,
lê o número de unidades produzidas , e para os recursos humanos e
materiais que estão alocados aquela atividade , lê as quantidades
utilizadas . Se a atividade estiver terminada , muda seu status
para terminada .

Pseudo-Código :

escrever cabeçalho.
ler quantidade de unidades de tempo do período apurado.
ENQUANTO houverem atividades para apurar
 IDENTIFICA ATIVIDADE para apurar;
 ler quantidade produzida no período;
 guardar na base sistema;
 incrementar a quantidade de unidades de tempo em execução;
SE duração prevista < tempo em execução
 erro de consistência.
PARA todos os recursos humanos alocados à atividade
 ler hh utilizados no período ;
 gravar na base sistema.

PARA todos os recursos materiais necessários
ler quantidade utilizada no período;
gravar na base sistema.
SE unidades produzidas \geq unidades previstas
SE atividade está terminada
muda status para terminada;
SENÃO
erro de consistência.
SE todas atividades do projeto executadas no período tiverem sido
apuradas , ativa MODULO 5.1.1 .

MODULO 5.1.1 - CRIAR ARQUIVO SEQUENCIA OU FECHAMENTO DO PERÍODO

Descrição :

Se todas as atividades de um certo período houverem sido apuradas,
este módulo lê todas as atividades do projeto , e para as que
houverem sido executadas durante aquele período , grava-as no
arquivo do período correspondente, para que posteriormente
possamos traçar as curvas de performance .

Pseudo-Código :

escrever cabeçalho e avisar que o arquivo PERÍODO é apagado.

SE pode continuar

ler o número do último período apurado;

gravar o número do período atual;

criar o arquivo PERÍODO ;

PARA todas as atividade do projeto

ler estado;

ler duração;

ler dias em execução;

ler unidades previstas;

ler unidades produzidas;

ler unidades produzidas no período;

zerar o período e incrementar unidades produzidas;

PARA todos os recursos humanos

ler quantidade prevista ;

ler quantidade utilizada ;

ler quantidade utilizada no período ;

zerar o período e incrementar utilizadas.

SE produzidas no período > 0 ou

quantidade utilizada no período > 0

gravar todos os dados no arquivo do período.

MODULO 5.2 - MEDIR PROGRESSO

Descrição :

Este módulo simplesmente ativa um de seus módulos subordinados ,
conforme a opção do usuário .

Pseudo-Código :

escrever cabeçalho.

CASO opção seja

"Por atividade" ativa MODULO 5.2.1 ;

"Por fase" ativa MODULO 5.2.2(0) ;

"De todo Projeto" ativa MODULO 5.2.2(1) ;

"Menu anterior" RETORNA

MODULO 5.2.1 - MEDIR PROGRESSO POR ATIVIDADE

Descrição :

Após identificar uma atividade , este módulo calcula a % completada da atividade de acordo com o item 4.4.2.2 .

Pseudo-Código :

escrever cabeçalho.
IDENTIFICA ATIVIDADE para medir progresso .
ler quantidade de unidades previstas.
ler quantidade de unidades produzidas.
ler unidades produzidas no período e incrementar produzidas.
PARA todos os recursos humanos
 acumular recursos humanos previstos ;
 acumular recursos humanos gastos .
calcula e exibe a % completada .

MODULO 5.2.2 - MEDIR PROGRESSO POR FASE OU MEDIR PROGRESSO DE TODO PROJETO

Descrição :

Este módulo é idêntico ao 5.2.1 , apenas ao invés de calcular o progresso para uma única atividade , calcula para o conjunto de atividades que pertencem a uma dada fase, ou de todas as atividades se for de todo o projeto . Este cálculo também é baseado no item 4.4.2.2 .

Pseudo-Código :

escrever cabeçalho.
SE for de todo o projeto
 copiar todas as atividades para a tabela de nomes.
SE for por fase
 SELECIONA as atividades e guarda na tabela de nomes.
PARA todas as atividades na tabela de nomes
 acumula hh orçados;
 acumula hh gastos ;
 acumula quant. de unidades previstas ;
 acumula quant. de unidades produzidas.
calcular e exibir a % completada.

MODULO 5.3 - AVALIAR PERFORMANCE

Descrição :

Este módulo traça as curvas a,b e c citadas no item 4.4.3.2 . O gráfico cobre um intervalo de 10 períodos . O usuário necessita apenas informar qual o número do período inicial do gráfico .

Pseudo-Código :

escreve cabeçalho.
SELECIONA atividades para o gráfico.
ler período inicial do gráfico.
calcular período final do gráfico.

PARA período inicial ao final
 zerar os acumuladores do período;
 abrir o arquivo do período;
 PARA todas as atividades no arquivo do período
 SE atividade pertence ao conjunto selecionado
 guarda os dados do período para gráfico.
 criar janela para custos.
 criar janela para períodos.
 criar janela para definições.
 preencher as janelas de custos e definições.
 traçar o grid do gráfico.
 PARA período inicial ao final
 traçar as 3 curvas do período ;
 colocar o numero do período na janela dos períodos.

MODULO 5.4 - MEDIR PRODUTIVIDADE

Descrição :

A medição da produtividade consiste em calcular os custos unitários real, orçado e teórico, conforme detalhado no item 4.4.2.4 . Este módulo apenas ativa o módulo que calculará os custos unitários conforme a opção escolhida pelo usuário.

Pseudo-Código :

escrever cabeçalho.
 CASO opção seja
 "Por atividade" ativa MODULO 5.4.1 ;
 "Por fase" ativa MODULO 5.4.2(0) ;
 "De todo projeto" ativa MODULO 5.4.2(1) ;
 "Menu anterior" RETORNA.

MODULO 5.4.1 - MEDIR PRODUTIVIDADE POR ATIVIDADE

Descrição :

Para uma atividade identificada , este módulo calcula seus custos unitários.

Pseudo-Código :

escrever cabeçalho.
 IDENTIFICA ATIVIDADE para medir produtividade.
 calcular % completada da atividade.
 PARA cada recurso humano alocado
 calcular e exibir : custo unitário real
 custo unitário orçado
 custo unitário teórico
 PARA cada recurso material utilizado
 calcular e exibir : custo unitário real
 custo unitário orçado
 custo unitário teórico.

**MODULO 5.4.2 - MEDIR PRODUTIVIDADE POR FASE OU
MEDIR PRODUTIVIDADE DE TODO PROJETO**

Descrição :

Este módulo calcula os custos unitários para um conjunto de atividades . Por tratar-se de um conjunto de atividades , podem existir atividades que já estejam terminadas . Assim é possível calcular além dos custos real,orçado e teórico , também o custo unitário histórico , baseado nas atividades já terminadas .

Pseudo-Código :

escrever cabeçalho.

SE for de todo o projeto

 copiar todas as atividades para a tabela de nomes.

SE for por fases

 SELECIONA as atividades e guarda na tabela de nomes.

PARA todas as atividades na tabela de nomes

 acumula dados para calcular a % completada.

calcular a % completada da fase ou do projeto.

PARA todos os recursos humanos

 PARA todas as atividades na tabela de nomes

 acumula os dados do tipo de recurso humano.

 calcula e exhibe : custo unitário real

 custo unitário orçado

 custo unitário teórico

 custo unitário histórico do recurso humano

PARA todos os recursos materiais

 PARA todas as atividades na tabela de nomes

 acumula os dados do tipo de recurso material.

 calcula e exhibe : custo unitário real

 custo unitário orçado

 custo unitário teórico

 custo unitário histórico do recurso material

5.9 - Especificação das Rotinas de Uso Geral

Além dos módulos anteriormente apresentados , foram criadas 8 rotinas de uso geral , que passamos a descrever :

1 - BUSCA-RELACIONAMENTOS

FINALIDADE : Dado um objeto origem , e um tipo de relacionamento , retorna a identidade de todos os objetos que possuem este relacionamento com o objeto origem .

PSEUDO-CÓDIGO :

tornar o objeto origem corrente.

obtem o código do relacionamento.

posiciona no 1º relacionamento.

ENQUANTO código do relacionamento posicionado =

 código do relacionamento desejado

 obtém dados do relacionamento;

 guarda na tabela de identificadores;

 avança para o próximo relacionamento.

2 - EXISTE

FINALIDADE : Dado um nome de objeto , verifica se ele existe na tabela de objetos daquele tipo .

PSEUDO-CÓDIGO :

PARA toda a tabela de objetos do tipo

 SE nome = nome da tabela RETORNA TRUE.

RETORNA FALSE.

3 - IDENTIFICA-ATIVIDADE

FINALIDADE : Identificar unívocamente uma atividade dentre as existentes , para posterior processamento .

PSEUDO-CÓDIGO :

SELECIONA as atividades de interesse e coloca na tabela nomes.

exibe a tabela de nomes.

escolhe um nome ou digita um novo nome.

4 - LIMPA

FINALIDADE : Preencher com " " uma string .

PSEUDO-CÓDIGO : Colocar " " em todas as posições da string.

5 - MONTA-FASES

FINALIDADE : Montar uma tabela apenas com as fases de nível zero (nível superior das fases do ciclo de vida).

PSEUDO-CÓDIGO :

ler o tipo de ciclo de vida do projeto.

tornar o ciclo de vida corrente.

posicionar na primeira fase.

ENQUANTO não houver erro

 obtém os dados da fase ;

 coloca na tabela de fases ;

 avança para a próxima fase .

PARA todas as fases da tabela

 SE fase não for de nível zero

 retira fase da tabela .

6 - MONTA-TABELA

FINALIDADE : Dado um tipo de objeto , retorna uma tabela com os nomes de todos os objetos daquele tipo .

PSEUDO-CÓDIGO :

obtem o código do tipo de objeto.

posiciona no primeiro objeto.

ENQUANTO não houver erro

 obtem dados do objeto ;

 coloca na tabela .

7 - RETIRA-ATIVIDADES

FINALIDADE : Retira da tabela de nomes de atividades , todas as atividades que dentro de um nível hierárquico específico , não pertençam ao tipo de fase selecionada .

PSEUDO-CÓDIGO :

PARA todas as atividades na tabela de nomes

 SE atividade não pertence ao tipo selecionado de fase

 retira atividade da tabela de nomes .

8 - SELECIONA

FINALIDADE : Retornar uma tabela com o nome das atividades que compõe um nível hierárquico .

PSEUDO-CÓDIGO :

copiar todas as atividades para a tabela de nomes.

copiar tabela de fases do nível zero para tabela de fases.

ENQUANTO tabela de fases não estiver vazia AND desejar descer um nível na hierarquia

 escolhe uma das fases da tabela de fases ;

 RETIRA ATIVIDADES que não pertencem àquela fase ;

 apaga a tabela de fases.

 BUSCA RELACIONAMENTOS de subfases da fase selecionada ;

 copiar para a tabela de fases as subfases encontradas ;

SE desejar descer mais um nível

 apaga a tabela de fases ;

 BUSCA RELACIONAMENTOS de tipos de atividades da última fase selecionada ;

 copiar para a tabela de fases os tipos de atividades encontrados ;

 escolhe um tipo de atividade ;

 RETIRA ATIVIDADES que não pertencem àquele tipo de ativ.;

CAPÍTULO 6 - UTILIZAÇÃO DO PROTÓTIPO

Neste capítulo apresentamos um exemplo de utilização da ferramenta automatizada que foi implementada para dar suporte ao modelo proposto para o GEPES .

Para fins de demonstração , criamos um ambiente de gerenciamento que possui :

Recursos Materiais : Recurso Material 1
 Recurso Material 2
 Recurso Material 3

Recursos Humanos : Analistas
 Gerentes
 Programadores

Tipos de Ciclos de Vida : Clássico
 Prototipagem

Tipos de Atividades : Tipo 1
 Tipo 2

Pessoas : Evandro
 Baptista
 Carlos
 Rubens

Fases e tipos de atividades do ciclo de vida clássico :

Análise	Subfase Análise 1	Subfase Análise 1 1	Tipo 1
			Tipo 2
		Subfase Análise 1 2	Tipo 1
			Tipo 2
	Subfase Análise 2	Tipo 1	
		Tipo 2	
Projeto			
Codificação			
Testes			
Manutenção			

Os nomes utilizados neste ambiente de gerenciamento foram escolhidos de forma a facilitar a apresentação e entendimento do exemplo de utilização do protótipo pois , por exemplo , onde aparecer "Tipo 1" durante a apresentação do exemplo de utilização, ficará imediatamente mais claro que estamos lidando com um tipo de atividade do que se o tipo de atividade se chamasse "Estudo Preliminar" .

Apenas os recursos materiais poderiam ter nomes mais familiares , como fizemos para os recursos humanos , mas como é apenas um exemplo de utilização da ferramenta, os nomes dados não são tão importantes .

A utilização da ferramenta pode ser considerada sob dois enfoques principais : a configuração do ambiente de gerenciamento e o gerenciamento dos projetos .

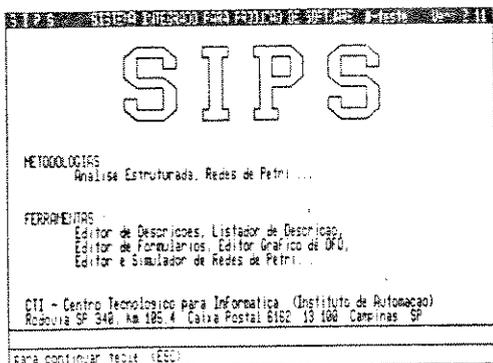
6.1 - Configuração do ambiente de gerenciamento

Configurar o ambiente de gerenciamento consiste em fornecer todas as informações que constam da Fig.10 e que são principalmente o nome do ambiente de gerenciamento , as fases e sub-fases do ciclo de vida adotado , as metodologias e produtos associados com cada fase e os recursos humanos e materiais disponíveis .

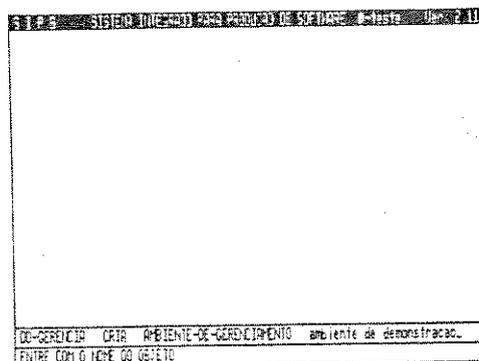
Conforme exposto no capítulo 5 , optamos pela utilização de uma ferramenta de uso genérico do SIPS , o editor de dicionário de dados , em conjunto com os quatro formulários AMBIENTE-DE-GERENCIAMENTO , TIPO-DE-CICLO-DE-VIDA , CATEGORIAS-RECURSOS-HUMANOS e FASES para o fornecimento dos dados do ambiente de gerenciamento .

A Fig.27 apresenta a criação de um ambiente de gerenciamento .

A Fig.28 apresenta o fornecimento de parte dos dados básicos do ambiente de gerenciamento através do uso do formulário AMBIENTE-DE-GERENCIAMENTO



A



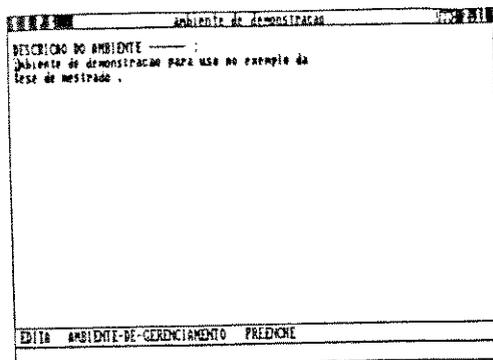
B

Figura 27 - Criação de um ambiente de gerenciamento

(A) - Ativação do sistema SIPS

(B) - Fornecimento do nome do ambiente de gerenciamento

(C) - Fornecimento da descrição do ambiente de gerenciamento



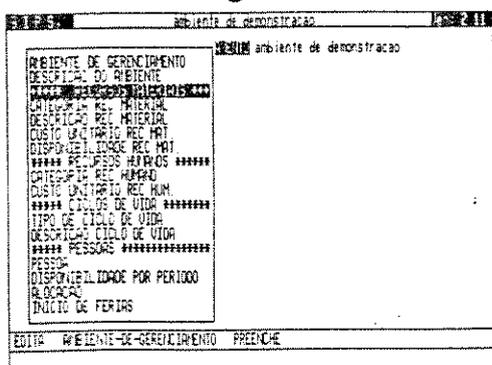
C

Figura 27 - Criação de um ambiente de gerenciamento (cont.)

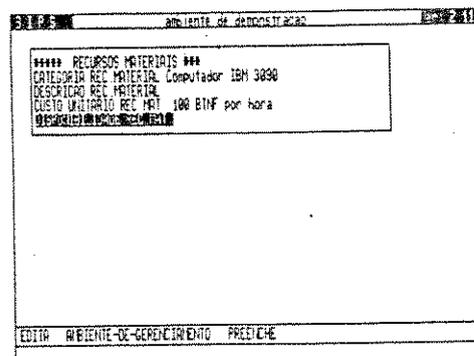
(A) - Ativação do sistema SIPS

(B) - Fornecimento do nome do ambiente de gerenciamento

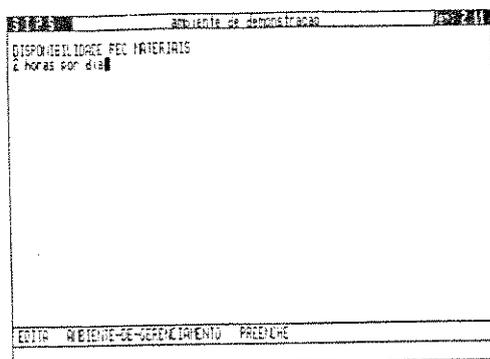
(C) - Fornecimento da descrição do ambiente de gerenciamento



A



B



C

Figura 28 - Fornecimento de dados do ambiente de gerenciamento

(A, B, C) - Recursos materiais

(D, E) - Recursos humanos

(F, G) - Ciclos de vida

(H, I) - Pessoas

ambiente de demonstracao

ambiente de demonstracao

AMBIENTE DE GERENCIAMENTO
 DESCRICAO DO AMBIENTE
 ***** RECURSOS MATERIAIS *****
 CATEGORIA REC. MATERIAL
 DESCRICAO REC. MATERIAL
 CUSTO UNITARIO REC. MAT.
 DISPONIBILIDADE REC. MAT.
 ***** RECURSOS HUMANOS *****
 CATEGORIA REC. HUMANO
 CUSTO UNITARIO REC. HUM.
 ***** CICLOS DE VIDA *****
 TIPO DE CICLO DE VIDA
 DESCRICAO CICLO DE VIDA
 ***** PESSOAS *****
 PESSOA
 DISPONIBILIDADE POR PERIODO
 ALOCACAO
 INICIO DE FERIAS

EDITA AMBIENTE-DE-GERENCIAMENTO PREENCHE

ambiente de demonstracao

***** RECURSOS HUMANOS *****
 CATEGORIA REC. HUMANO Analistas
 CUSTO UNITARIO REC. HUM. 150 Cr\$/h

EDITA AMBIENTE-DE-GERENCIAMENTO PREENCHE

D

ambiente de demonstracao

ambiente de demonstracao

AMBIENTE DE GERENCIAMENTO
 DESCRICAO DO AMBIENTE
 ***** RECURSOS MATERIAIS *****
 CATEGORIA REC. MATERIAL
 DESCRICAO REC. MATERIAL
 CUSTO UNITARIO REC. MAT.
 DISPONIBILIDADE REC. MAT.
 ***** RECURSOS HUMANOS *****
 CATEGORIA REC. HUMANO
 CUSTO UNITARIO REC. HUM.
 ***** CICLOS DE VIDA *****
 TIPO DE CICLO DE VIDA
 DESCRICAO CICLO DE VIDA
 ***** PESSOAS *****
 PESSOA
 DISPONIBILIDADE POR PERIODO
 ALOCACAO
 INICIO DE FERIAS

EDITA AMBIENTE-DE-GERENCIAMENTO PREENCHE

E

ambiente de demonstracao

***** CICLOS DE VIDA *****
 TIPO DE CICLO DE VIDA classico
 DESCRICAO CICLO DE VIDA

EDITA AMBIENTE-DE-GERENCIAMENTO PREENCHE

F

ambiente de demonstracao

ambiente de demonstracao

AMBIENTE DE GERENCIAMENTO
 DESCRICAO DO AMBIENTE
 ***** RECURSOS MATERIAIS *****
 CATEGORIA REC. MATERIAL
 DESCRICAO REC. MATERIAL
 CUSTO UNITARIO REC. MAT.
 DISPONIBILIDADE REC. MAT.
 ***** RECURSOS HUMANOS *****
 CATEGORIA REC. HUMANO
 CUSTO UNITARIO REC. HUM.
 ***** CICLOS DE VIDA *****
 TIPO DE CICLO DE VIDA
 DESCRICAO CICLO DE VIDA
 ***** PESSOAS *****
 PESSOA
 DISPONIBILIDADE POR PERIODO
 ALOCACAO
 INICIO DE FERIAS

EDITA AMBIENTE-DE-GERENCIAMENTO PREENCHE

G

ambiente de demonstracao

***** PESSOAS *****
 PESSOA exatino
 DISPONIBILIDADE POR PERIODO 40 h
 ALOCACAO
 INICIO DE FERIAS 22/05/91
 DURACAO DAS FERIAS 28 dias
 NOME DO GERENTE Jino

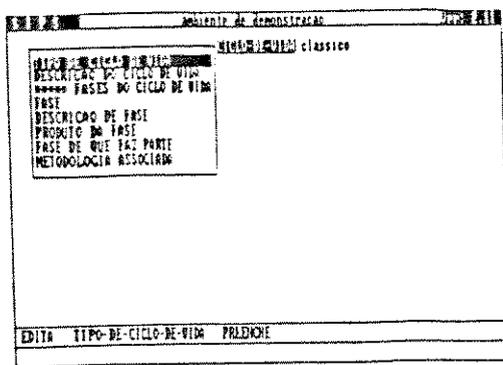
EDITA AMBIENTE-DE-GERENCIAMENTO PREENCHE
 Novo objeto inserido na base (ESC)

H

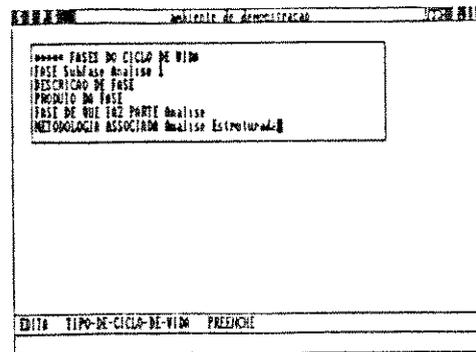
I

Figura 28 - Fornecimento de dados do ambiente de gerenciamento
 (A, B, C) - Recursos materiais (cont.)
 (D, E) - Recursos humanos
 (F, G) - Ciclos de vida
 (H, I) - Pessoas

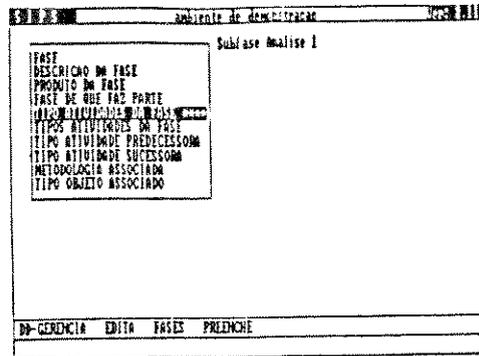
A Fig.29 apresenta a caracterizaco de parte do ciclo de vida clssico adotado neste exemplo , atravs da utilizaco dos formulrios TIPO-DE-CICLO-DE-VIDA e FASES .



A



B



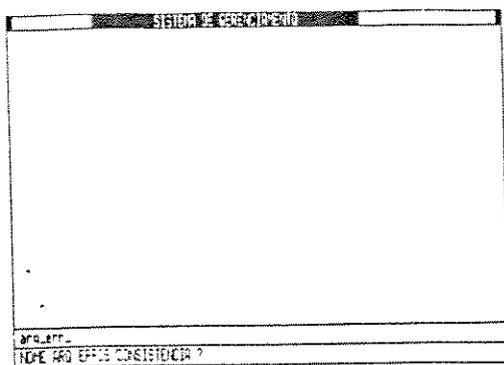
C

Figura 29 - Caracterização de parte do ciclo de vida clássico
 (A) - Identificação do tipo de ciclo de vida
 (B) - Fornecimento das fases e metodologias
 (C) - Detalhamento de cada fase ou subfase

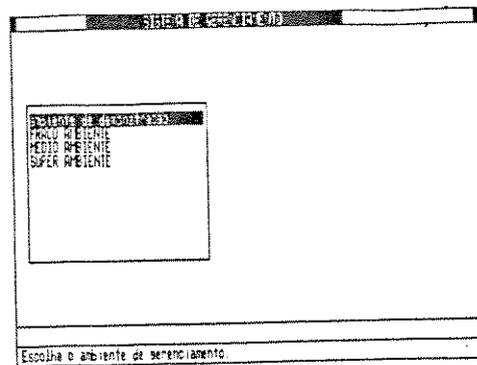
Uma vez fornecidas todas as informações que compõe o ambiente de gerenciamento, através das telas geradas com o editor de formulários, podemos iniciar realmente o gerenciamento de projetos, que utiliza a ferramenta específica para o gerenciamento de projetos que foi implementada baseada no modelo proposto no capítulo 4. Isto significa que as próximas telas são as geradas pela própria ferramenta específica para o gerenciamento.

6.2 - GERENCIAMENTO DE PROJETOS

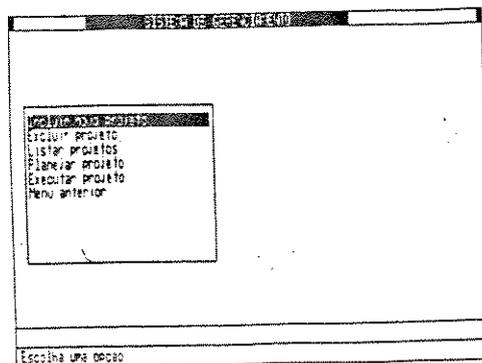
O gerenciamento de projetos está subdividido em duas etapas principais, conforme exposto na seção 4.4 - O Funcionamento do Modelo Proposto, e que são: o planejamento e a execução do projeto de engenharia de software. A Fig.30 apresenta a ativação da ferramenta para o gerenciamento. A primeira pergunta da ferramenta é o nome do arquivo para mensagens de erro e consistência. Dá-se um nome qualquer. Listando-se este arquivo ao final do processamento é possível verificar e tentar corrigir erros de consistência das informações contidas na Base Sistema, como por exemplo uma dada atividade não possuir quantidades previstas para produzir. Em seguida escolhe-se o ambiente de gerenciamento, e é apresentado o menu principal.



A



B



C

Figura 30 - Ativar a ferramenta para o GEPES

- (A) - Nome do arquivo para mensagens de erro
- (B) - Escolha do ambiente de gerenciamento
- (C) - Apresentação do Menu Principal

O menu principal (Fig.30 C) indica os tres grupos principais de operações que podem ser realizadas :

- | | |
|--|--|
| <ul style="list-style-type: none"> 1 - Manuseio de Projetos 2 - Planejar Projetos 3 - Executar Projetos . | <ul style="list-style-type: none"> Inclusão de novo projeto Excluir projeto Listar projetos |
|--|--|

As atividades dos grupos 2 e 3 são as já descritas anteriormente quando da apresentação do modelo proposto no capítulo 4 .

A primeira coisa que se deve fazer para gerenciar projetos é incluir um projeto caso o mesmo ainda não tenha sido incluído, e fornecer suas informações, conforme a Fig.31. Uma das informações mais importantes quando incluímos um novo projeto é qual o tipo de ciclo de vida associado com este projeto .

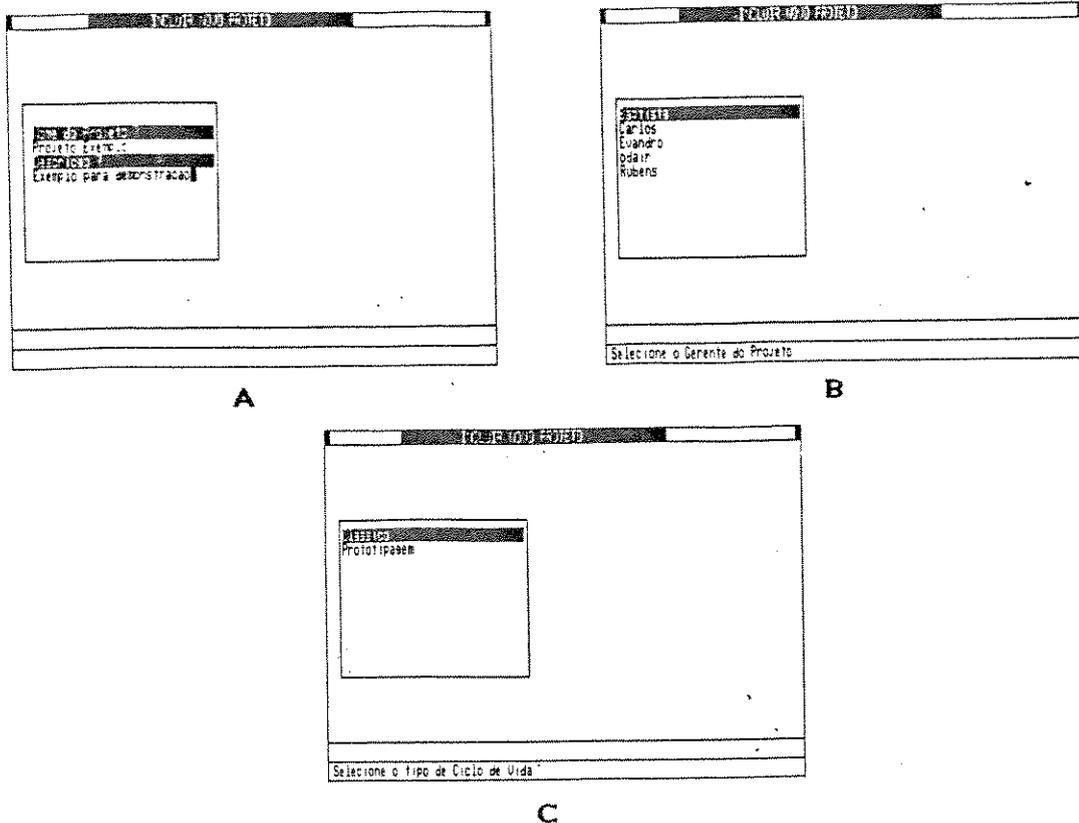


Figura 31 - Inclusão de um novo projeto
 (A) - Nome e descrição do projeto
 (B) - Gerente do projeto
 (C) - Tipo de ciclo de vida adotado para o projeto

Uma vez que o ambiente de gerenciamento esteja configurado e já exista algum projeto incluído na base de dados, podemos iniciar o Planejamento do Projeto .

6.2.1 - O Planejamento do Projeto de Engenharia de Software

O planejamento de projetos de engenharia de software é ativado escolhendo-se a opção "Planejar Projeto" no menu principal . Deve-se em seguida selecionar o projeto que se deseja planejar , sendo apresentada pela ferramenta o menu do planejamento de projetos , conforme mostrado na Fig.32 .

O planejamento do projeto possui 4 atividades principais (Fig.32 C) : Subdividir Trabalho , Quantificar Trabalho , Sequenciar Trabalho e Montar Equipe . Estas 4 atividades cobrem as funções das 6 atividades do modelo proposto apresentadas na seção 4.4.1 - O Planejamento do Projeto de Engenharia de Software segundo o Modelo Proposto.

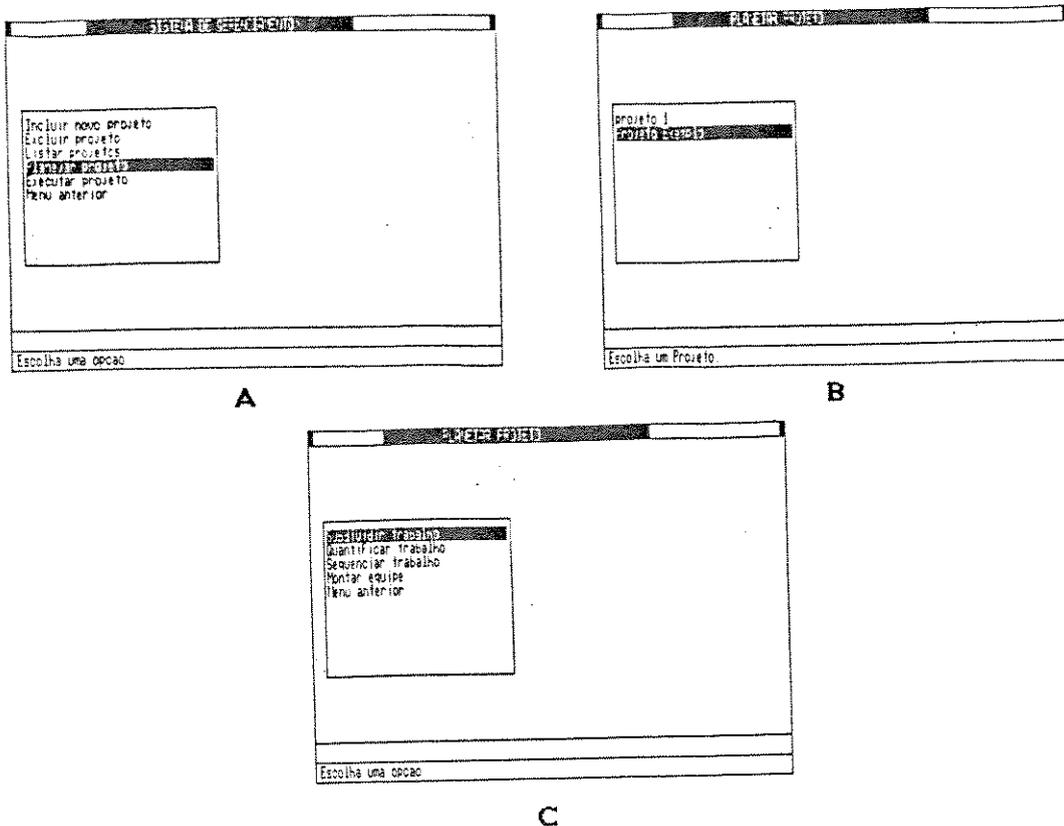


Figura 32 - Planejamento do projeto

(A) - Ativar o planejamento

(B) - Escolher o projeto para planejar

(C) - Menu das atividades principais do planejamento

A subdivisão do trabalho, dentro do Planejamento do Projeto, consiste em informar as atividades que compõe o projeto. A Fig.33 apresenta a inclusão da atividade do tipo 1, "Atividade Exemplo" na Sub Fase 1.1 da Análise do projeto exemplo.

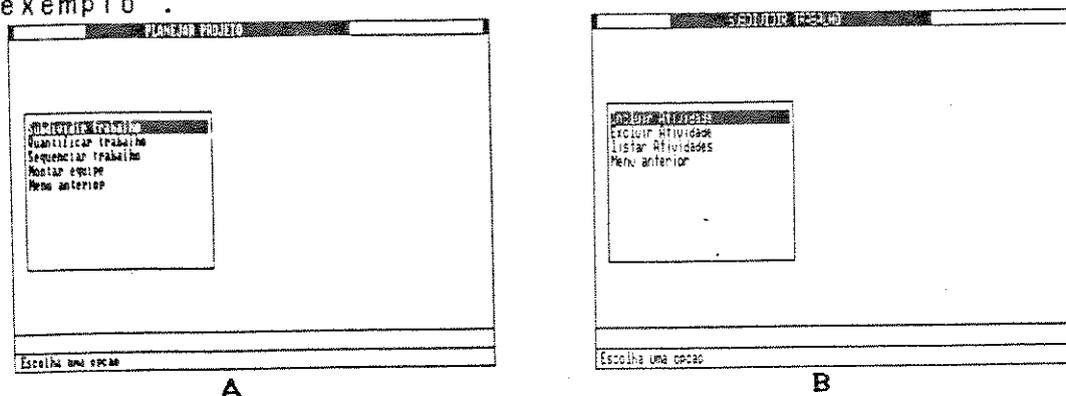


Figura 33 - A subdivisão do trabalho

(A) - Escolha da atividade de subdivisão do trabalho

(B) - Escolha de inclusão de atividade

(C, D, E, F) - Identificação do posicionamento no ciclo de vida, da atividade que será incluída

(G, H) - Nome da nova atividade

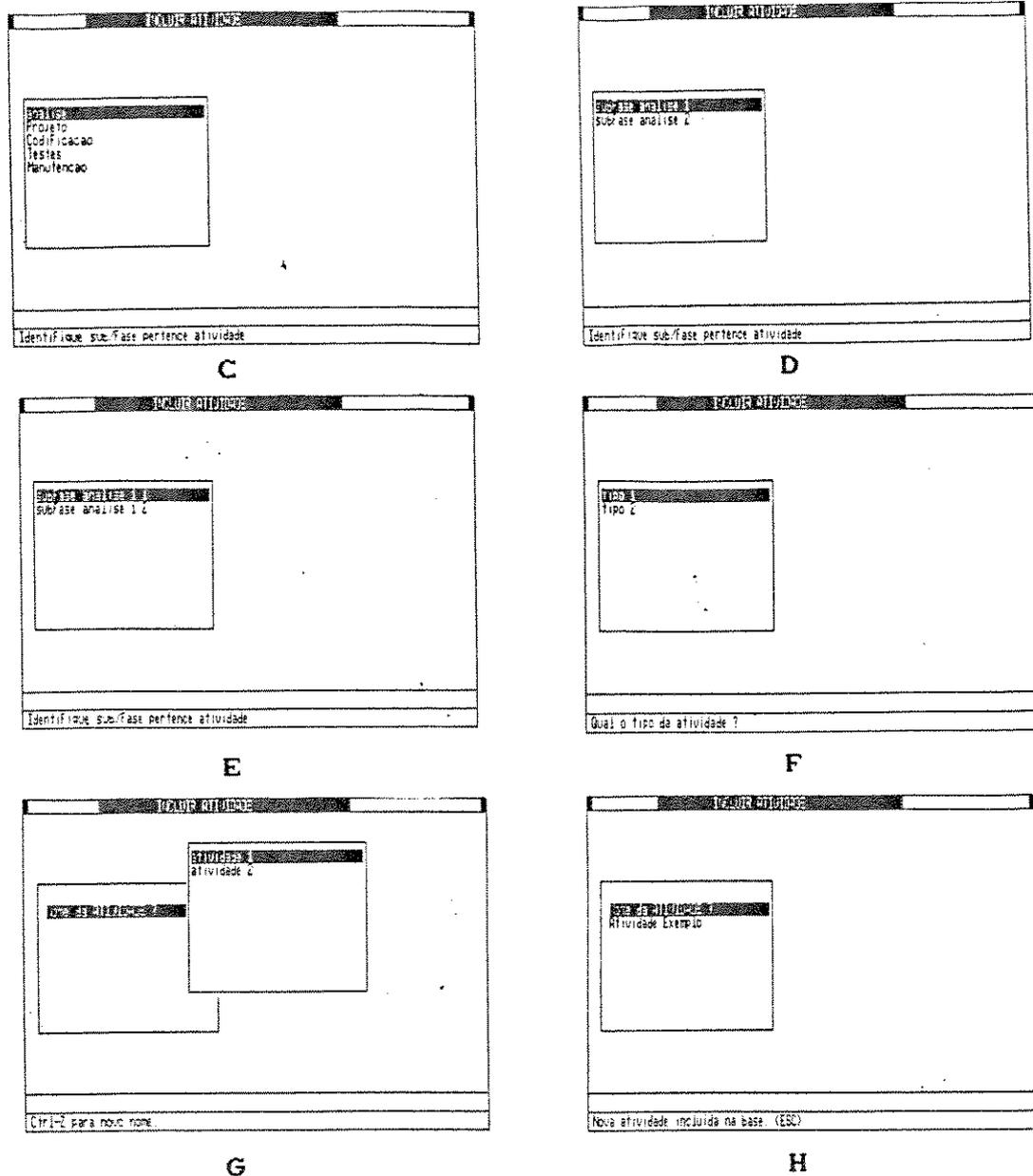


Figura 33 - A subdivisão do trabalho (cont.)

- (A) - Escolha da atividade de subdivisão do trabalho
- (B) - Escolha de inclusão de atividade
- (C,D,E,F) - Identificação do posicionamento no ciclo de vida , da atividade que será incluída
- (G,H) - Nome da nova atividade

Para o restante das atividades do Planejamento e as da Execução do Projeto de Engenharia de Software , será sempre necessário selecionar uma atividade ou fase do ciclo de vida do projeto que estiver sendo planejado ou executado . A Fig. 34 apresenta a sequencia de telas para seleção de uma atividade do projeto. Esta sequencia não mais será exibida durante o exemplo , porém sempre que nos referenciarmos a selecionar uma atividade do projeto , estaremos nos referindo a Fig. 34 .

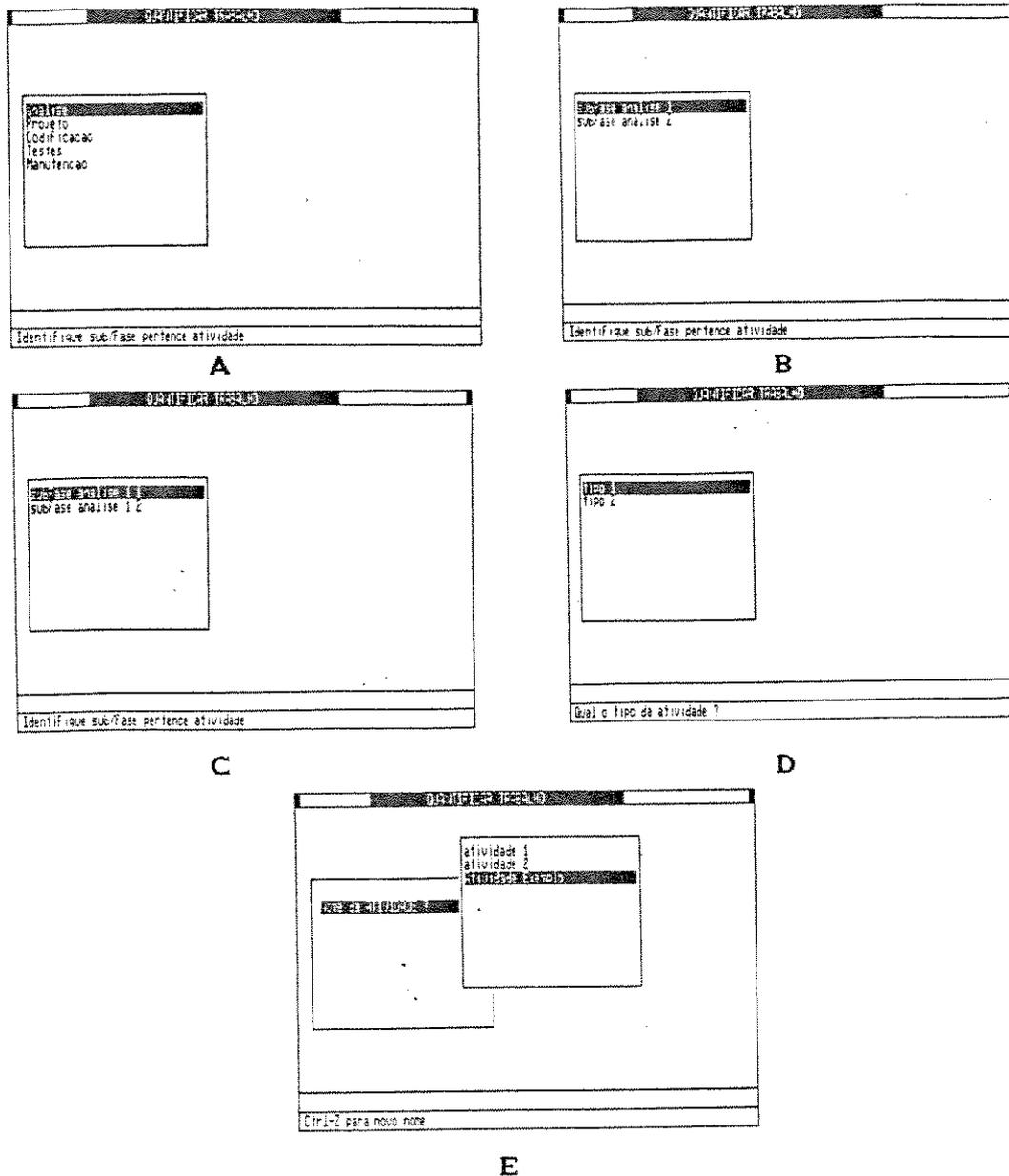


Figura 34 (A, B, C, D, E) - Selecionar uma atividade de um projeto.

Dentro da fase de Planejamento , QUANTIFICAR uma atividade do projeto significa , após selecionar a atividade , informar qual a unidade de seu produto e quantas unidades do mesmo serão produzidas . Em seguida para todos os recursos humanos e materiais necessários para esta atividade , seleciona-se e informa-se a quantidade de cada recurso . No exemplo da Fig. 35 apresentamos apenas o tipo de recursos humanos "analistas" .

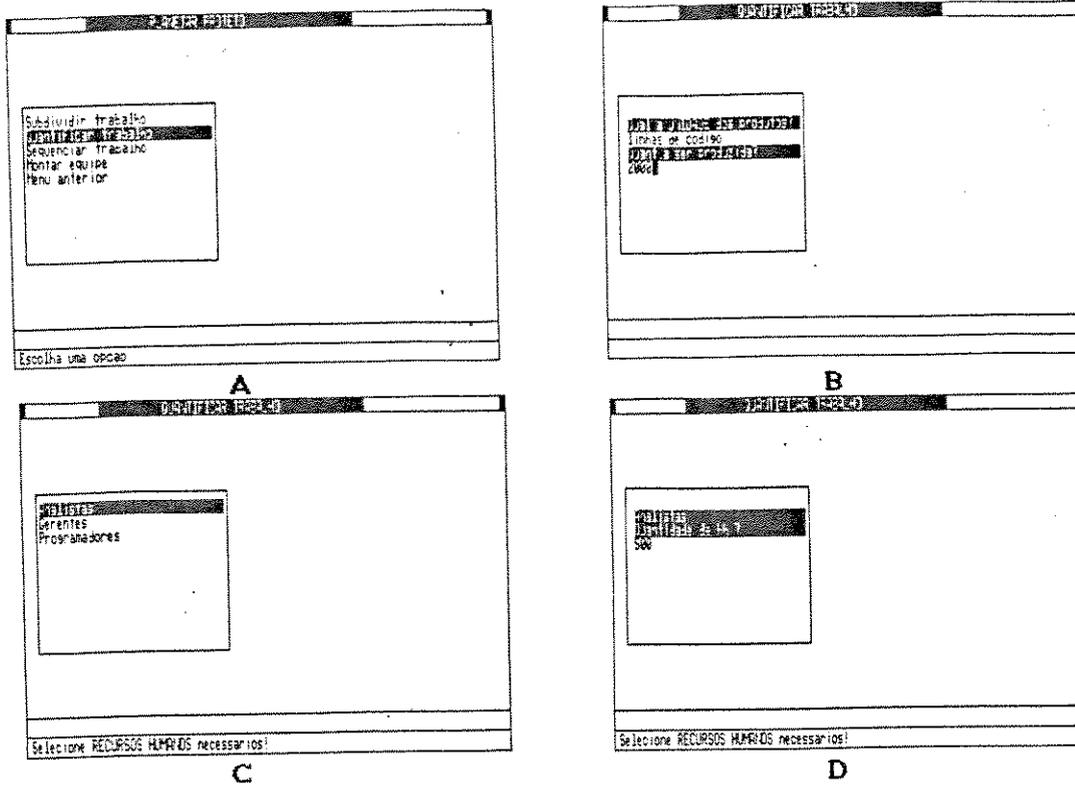
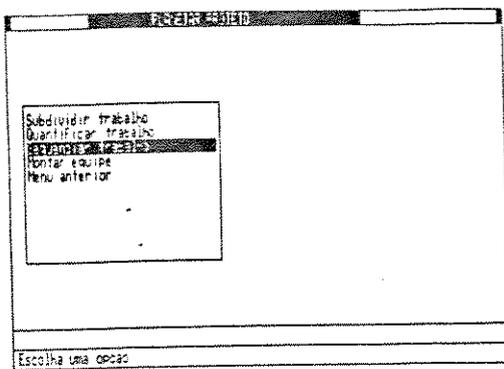


Figura 35 - Quantificar atividades do projeto

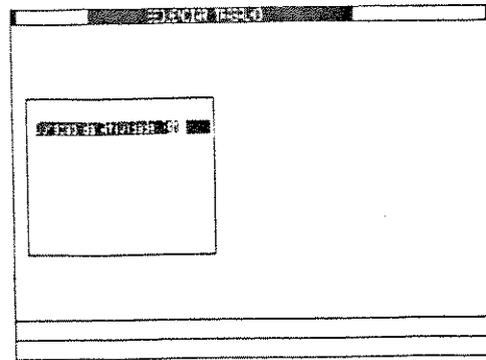
- (A) - Escolher a quantificação de atividades
- (B) - Fornecer unidade e quantidade do produto da atividade
- (C) - Selecionar o recurso humano necessário
- (D) - Informar quantidade necessária do recurso selecionado

Ainda na fase de planejamento do projeto , uma vez quantificadas as atividades do projeto , podemos realizar o sequenciamento das atividades do projeto .

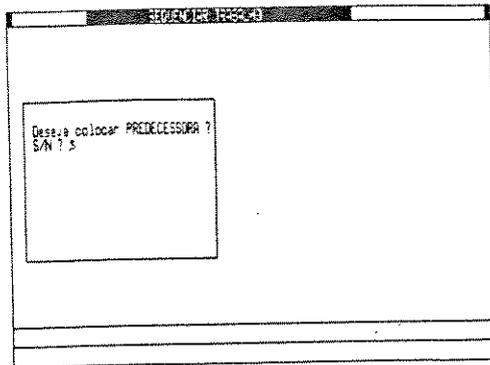
O sequenciamento consiste em informarmos a duração de cada atividade selecionada e suas predecessoras e sucessoras , conforme mostrado na Fig. 36 .



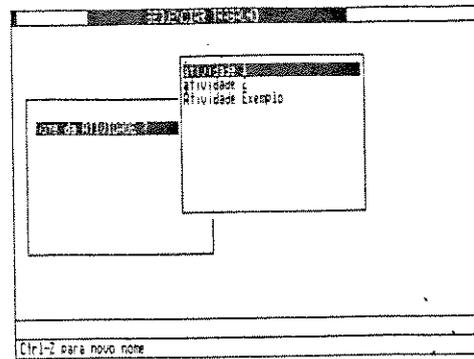
A



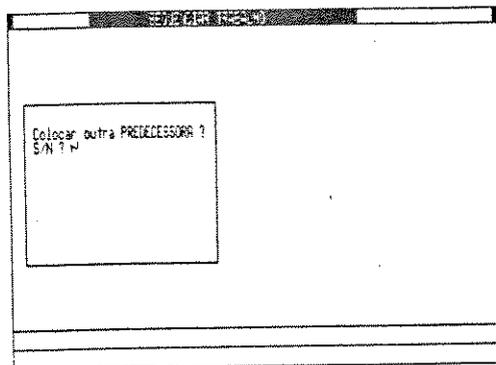
B



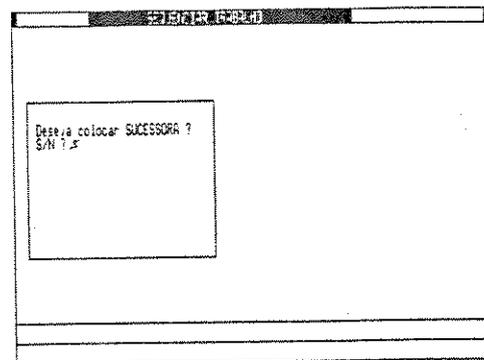
C



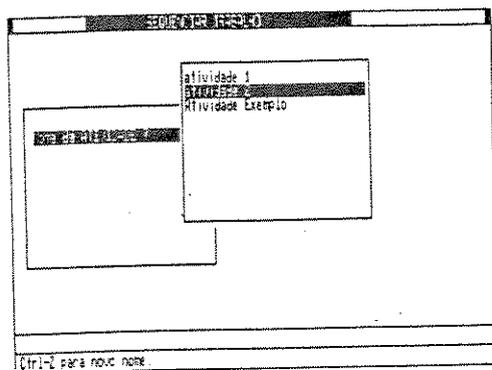
D



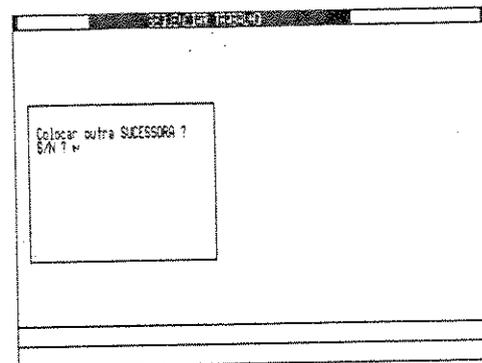
E



F



G



H

Figura 36 - Sequenciamento das atividades do projeto
 (A) - Ativar o sequenciamento
 (B) - Informar a duração da atividade
 (C, D, E) - Informar predecessoras
 (F, G, H) - Informar sucessoras

A montagem da equipe consiste simplesmente em fornecer o nome do responsável por cada atividade, que omitiremos.

Neste ponto termina o Planejamento do Projeto e tem início a EXECUÇÃO do Projeto .

6.2.2 - A EXECUÇÃO DO PROJETO DE ENGENHARIA DE SOFTWARE

Após escolhermos a execução de projetos no menu principal e escolhermos um projeto para ser executado , no caso o projeto Exemplo, podemos escolher uma das atividades compreendidas na execução do projeto e que estão apresentadas no menu da execução de projetos , Fig. 37 C . A atividade ESTIMATIVAS não foi implementada .

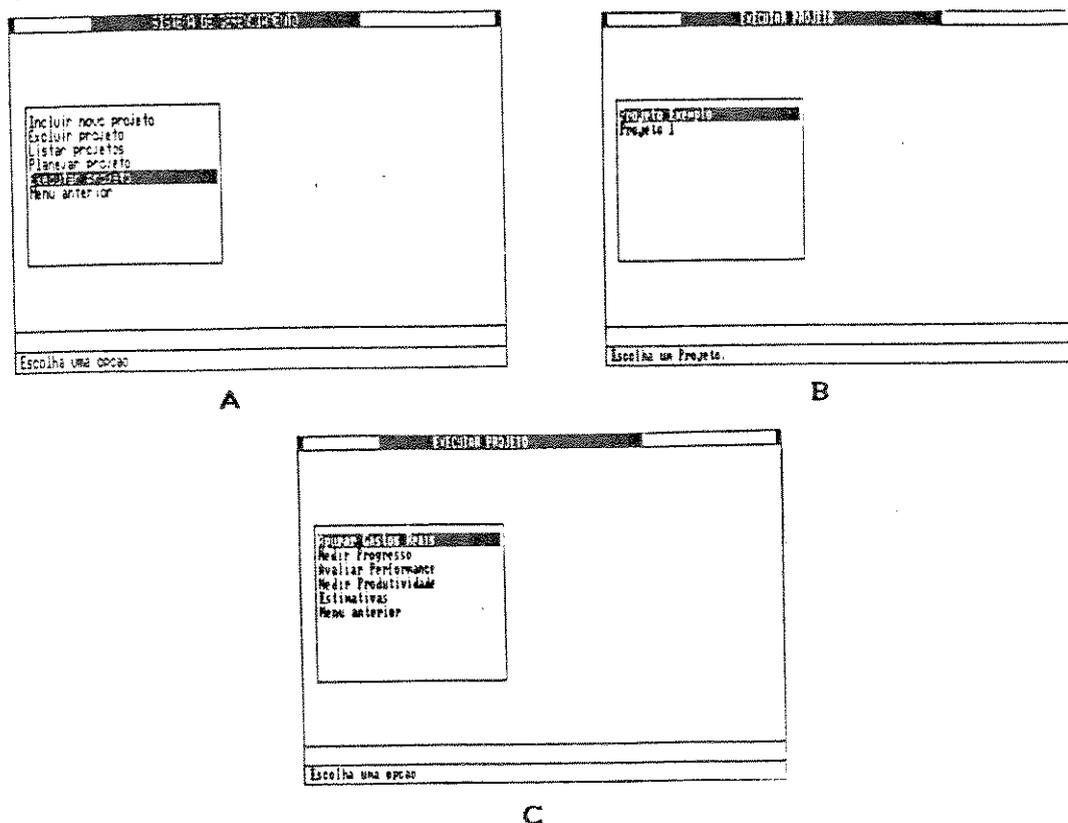


Figura 37 - A execução do projeto de engenharia de software
(A) - Ativar a execução de projetos
(B) - Selecionar o projeto a ser executado
(C) - Menu principal da execução de projetos

Apurar gastos reais , Fig. 38 , consiste em informar periodicamente, quanto foi consumido de cada recurso e quanto foi produzido pela atividade do projeto ..

Primeiramente informamos o número de unidades de tempo a que referem-se as quantidades apuradas e em seguida identificamos cada atividade e informamos as quantidades produzidas e consumidas .

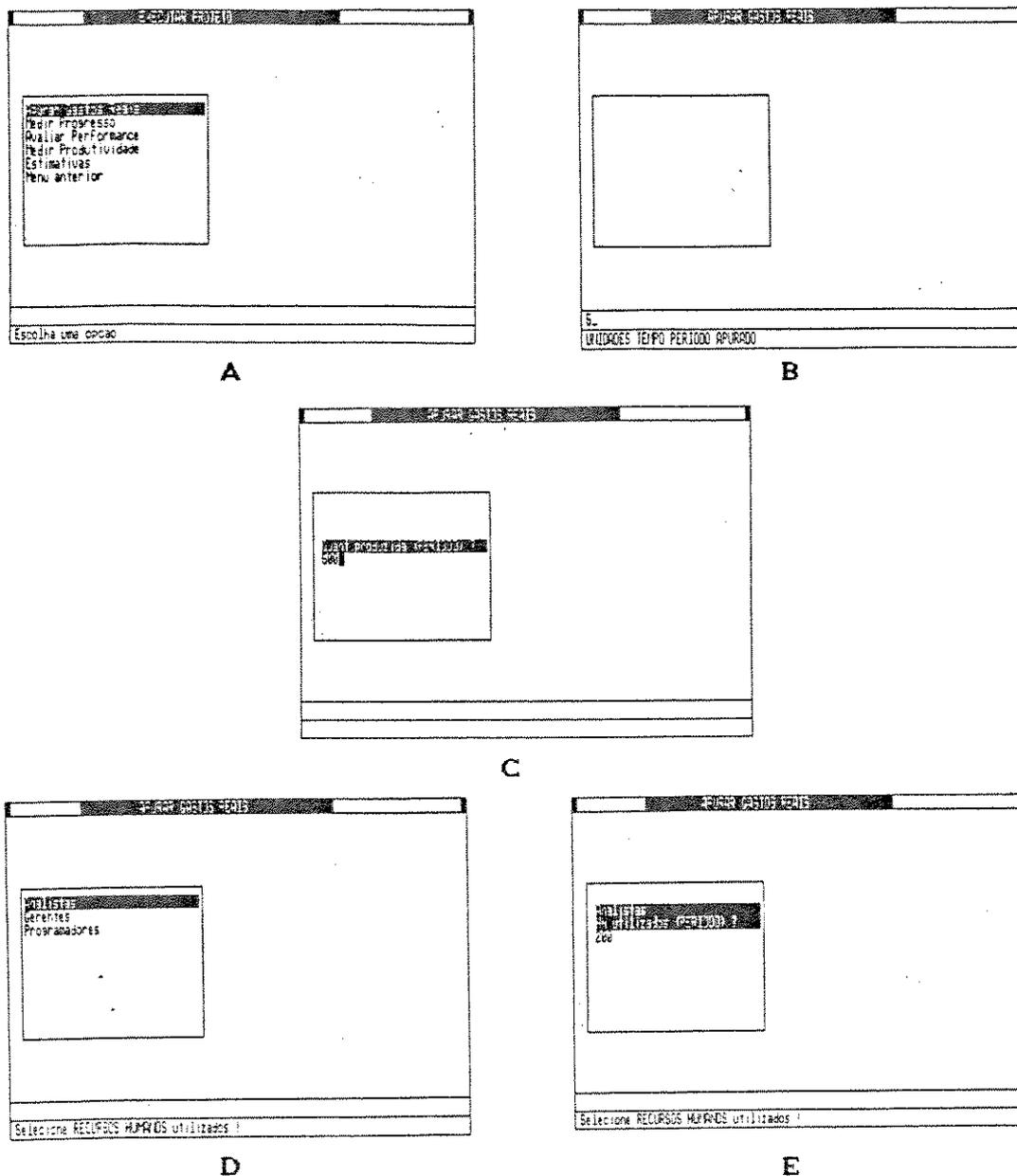


Figura 38 - Apurar gastos reais

- (A) - Selecionar apuração de gastos reais
- (B) - Informar número de unidades de tempo do período apurado
- (C) - Informar quantidade de unidades produzidas pela atividade durante este período
- (D,E) - Informar quantidades utilizadas de cada recurso

Ainda dentro da apuração dos gastos reais , se tivermos apurado TODAS atividades que foram executadas durante este período , o período poderá ser arquivado para que possamos fazer os calculos de desempenho (Fig. 39) .

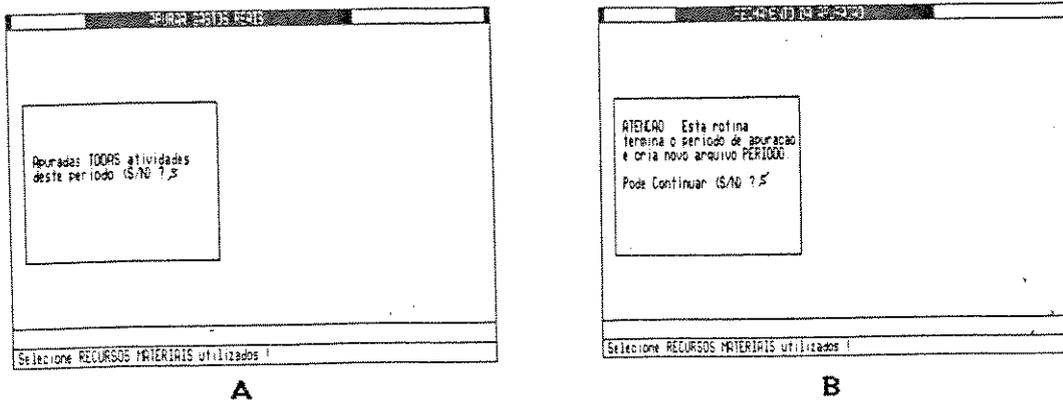
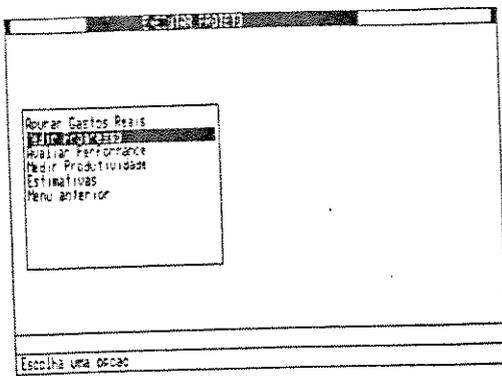


Figura 39 (A,B) - Fim da apuração de todas atividades de um período

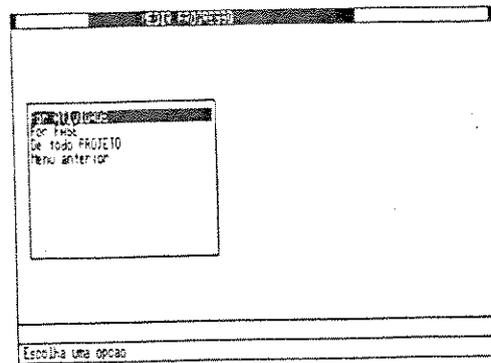
As 3 funções restantes da execução do projeto : Medir o Progresso , Avaliar Performance (Desempenho) e Medir Produtividade , são as funções principais do *Controle* do projeto . Estas funções já foram descritas no modelo proposto.

Tanto a medição do progresso como a da produtividade podem ser realizadas : para atividades isoladas do projeto , para fases específicas do projeto ou para todo o projeto .

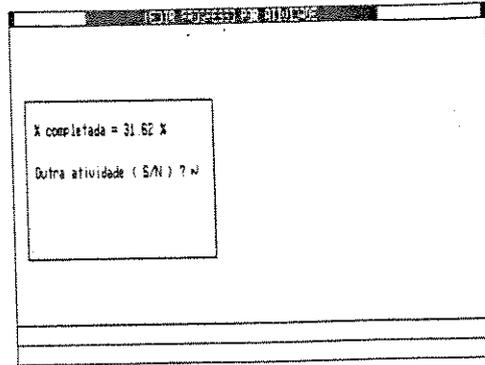
Na Fig.40 apresentamos a medição do progresso para uma atividade selecionada .



A



B



C

Figura 40 - Medição do Progresso

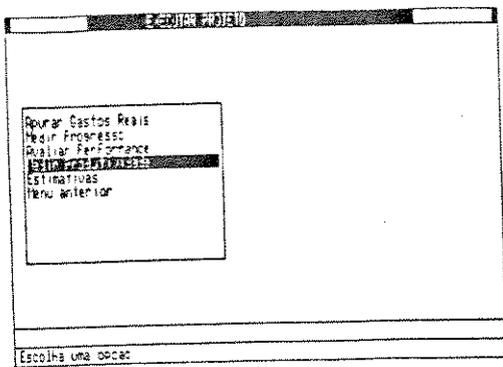
(A) - Seleção da medição do progresso

(B) - Escolha da opção por atividade

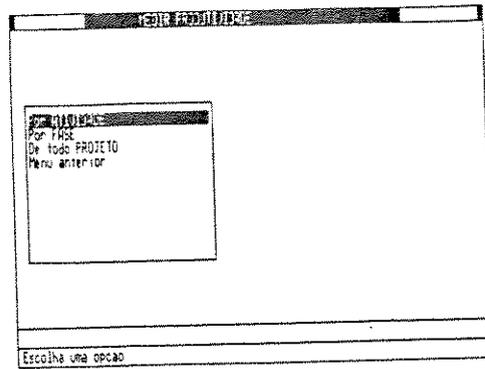
(C) - Medição do progresso da atividade selecionada

A medição da produtividade (custos unitários) é similar a medição do progresso e nos informa os custos unitários real , orçado e teórico para cada um dos tipos de recursos humanos e materiais utilizadas por uma dada atividade , fase do projeto ou de todo o projeto .

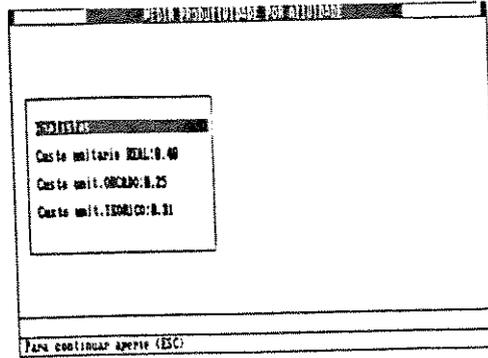
A Fig. 41 apresenta o exemplo da medição da produtividade dos analistas de uma atividade selecionada .



A



B



C

Figura 41 - Medição da Produtividade

(A) - Ativar a medição de produtividade

(B) - Medir produtividade de uma atividade que será selecionada

(C) - Produtividade do recurso humano ANALISTAS da atividade selecionada

Finalmente temos a avaliação da performance .

A performance é avaliada por meio das curvas de custo real, orçado e teórico, conforme já detalhado anteriormente. Estas curvas também podem ser traçadas por atividades, fases ou de todo o projeto. Informando-se o período inicial do gráfico, as 3 curvas são exibidas numa faixa de tempo que cobre do período inicial até no máximo 10 períodos à frente, conforme exibido na Fig. 42.

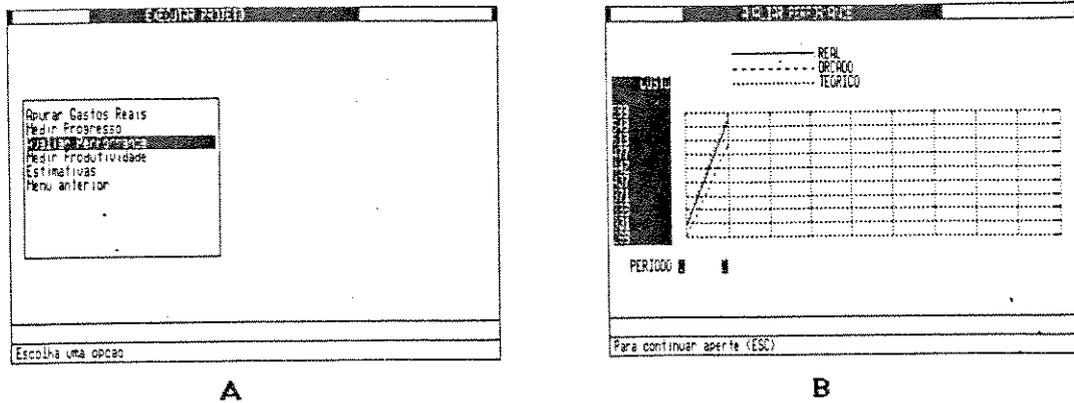


Figura 42 - Avaliação da Performance
 (A) - Ativar a avaliação da performance
 (B) - Avaliação da performance de uma atividade selecionada .

CAPÍTULO 7 - CONCLUSÕES

Neste trabalho apresentamos alguns modelos de gerenciamento de projetos e algumas ferramentas automatizadas para o gerenciamento de projetos , comercialmente disponíveis .

Como as ferramentas não são específicas para projetos de engenharia de software , propusemos um modelo que integrasse as informações da engenharia de software com o gerenciamento de projetos.

Este modelo possui duas partes principais : o ambiente de gerenciamento , onde estão todas as informações referentes à engenharia de software, tais como os ciclos de vida , suas fases, as metodologias e seus produtos, e os recursos humanos e materiais disponíveis ; e o ambiente de projetos , onde temos os projetos com suas atividades.

Baseados neste modelo, implementamos uma ferramenta automatizada para o gerenciamento de projetos . Esta ferramenta foi implementada no SIPS , por razões já expostas anteriormente , e visa auxiliar o GP nas fases de planejamento e execução do projeto, inclusive propondo uma nova forma para o cálculo do progresso do projeto (vide ítem 5.4.2.2).

Infelizmente a ferramenta ainda está incompleta, podendo-se citar como possíveis melhoramentos :

- 1 - A ferramenta não faz o cálculo da rede de precedências (PERT). Porém como todas as informações para o cálculo da rede estão disponíveis na Base Sistema , cremos que com um pequeno esforço será possível fazer o cálculo em nosso próprio protótipo auxiliando ainda mais o gerente . Além disso , se a opção for submeter o arquivo a uma das ferramentas existentes para cálculo de redes de PERT , será necessário criar um módulo que gere este arquivo , e outro módulo que leia um arquivo gerado pela ferramenta externa para colocar automaticamente os resultados da rede em nossa Base Sistema . Provavelmente estes dois módulos tomarão o mesmo tempo e esforço que um módulo para cálculo da rede de PERT.

- 2 - A ferramenta não faz a compatibilização (nivelamento) de recursos necessários X recursos disponíveis . Esta informação é também muito importante para o GP , principalmente em ambientes onde existam vários projetos , em que os recursos sejam distribuídos entre estes projetos .
Mais uma vez dizemos que como as informações estão todas disponíveis na Base Sistema , pode-se criar um módulo que faça esta compatibilização em conjunto com o gerente .
- 3 - A ferramenta não está integrada com as metodologias existentes no SIPS .
Isto significa que ao dizer por exemplo que a metodologia associada com uma certa fase é a análise estruturada e que o produto de uma atividade é um DFD , esta informação fica simplesmente guardada na Base Sistema .
Seria interessante que fosse dado o nome do DFD , e que a ferramenta verificasse se o DFD já existe ou não e , caso negativo , criasse-o e que a atividade só fosse dada como encerrada , no momento em que o DFD fosse considerado um item de software congelado .
Isto não é trivial , ainda que todas as informações necessárias estejam disponíveis . Achamos que uma boa abordagem para este tipo de problema , que na verdade seria a integração de um ambiente de engenharia de software , seria a criação de um sistema especialista em engenharia de software , que coordenasse e controlasse toda a troca de informações da Base Sistema .
- 4 - Ainda que não seja bem uma falha da ferramenta , seria interessante que os projetos terminados pudessem constituir uma base de dados histórica , onde seriam guardadas as informações referentes a : produtividade obtida para os diversos tipos de atividades , erros de estimativas , problemas ocorridos , etc. . Estas informações poderiam ser usadas , idealmente automaticamente , para verificarmos a correteza das informações fornecidas para um novo projeto , ou mesmo apenas para fins de estimativas de esforço e duração de novos projetos , apenas para citar algumas das possíveis utilizações .
- 5 - O número de atividades , recursos , pessoas , projetos , fases, ciclos de vida , etc. estão limitados a 100 de cada . Esta é uma limitação apenas de protótipo , e o limite real será a quantidade de memória disponível .
- 6 - Ainda por limitação de memória apenas , o número de níveis hierárquicos do projeto é de 5 .

Por mais completa que fosse a ferramenta que implementamos , sempre faltaria alguma coisa . Porém implementando-se os melhoramentos anteriormente destacados , a ferramenta poderá ser de grande utilidade para o GP .

O modelo nos parece bom apesar das falhas , possuindo as informações básicas necessárias ao gerenciamento dos projetos de engenharia de software , destacando-se como ponto principal do mesmo , a separação do ambiente de gerenciamento do ambiente dos projetos , estando estes últimos firmemente amarrados ao ambiente de gerenciamento , o que de certa forma força a utilização dos procedimentos de engenharia de software que estão contidos no ambiente de gerenciamento .

CAPÍTULO 8 - BIBLIOGRAFIA

- [AGR86] Agresti , W.W. - "What are the new paradigms." - New Paradigms for Software Development : Tutorial. Washington DC : IEEE Computer Society , 1986 , pp. 6-10
- [BAK72] Baker , F.T. - "Chief Programmer Team Management of Production Programming" - IBM System Journal 1 , 1972, pags.57-73
- [BR075] Frederick P. Brooks Jr. - "The Mythical Man-Month." - Addison-Wesley Publishing Company Inc. - 1975
- [CAP88] Miriam Akemi Manabe Capretz - "Automatização do Gerenciamento de Configuração de Software" - Dissertação apresentada à FEC-UNICAMP para obtenção do título de Mestre , Agosto 1988
- [CHE76] P.P.Chen - "The Entity Relationship Model : Towards a Unified View of Data" - ACM Transactions on Database Systems , 1 , 1 , Março 1976
- [CO083] J.E.Coolahan Jr. e N.Roussopoulos - "Timing Requirements for Time-driven Systems Using Augmented Petri Nets." - IEEE Transactions on Soft.Eng., vol.SE-9 .n.5 , pp 603-616 Set.1983
- [COR85] Cori , K.A. - "Fundamentals of Master Scheduling for the Project Manager" - Project Management Journal , Junho 1985 pags 78-89 .
- [CUR87] B.Curtis , H.Krasner , V.Shen e N.Iscoe - "On Buiding Software Process Models Under the Lamspot" - Proc.9th Int.Conf. Software Engineering , Monterey , CA , 30 Março a 2 Abril 1987 , pp 96-103
- [DRU64] Drucker , P.F. - "Managing for Results" - New York : Harper & Row 1964
- [FAR89] Paulo Cesar Faria - "Escolha seu Software para Grenciar Projetos." - Jornal Automação e Industria , pp. 16-18, Maio 1989
- [FER87] Fersko , Weiss H. - "Master Plans : Project Management Software" - Revista PC Magazine , pp. 153 a 209 , Set.1987
- [HER59] Herzberg , F. ; Mausner , B. e Snyderman , B. B. - "The Motivation to Work" - John Wiley & Sons , NY 1959
- [HOW84] Howes ,N.R. - "Managing Software Development Projects for Maximum Productivity" - IEEE ,Transactions on Software Engineering ,Janeiro 1984 , pags 79-87

- [JIN88] Mario Jino , Caetano Traina Jr., Fabio Nauras Akhras , Luiz Fernando Capretz , Mario Bento de Carvalho , Miriam A.M. Capretz - "Automação de Metodologias de Desenvolvimento de Software usando o SIPS" - Documento interno do CTI , Centro Tecnológico para Informática , Campinas-sp , 1988
- [K0076] Koontz,H. e O'Donnell,C. "Management - A systems and contingency analysis of managerial functions" 6th edition - McGraw_Hill 1976 ,p.129,p.639
- [K0084] H.Koontz, C.O'Donnell e H.Weihrich , Management , 8.ed. McGraw-Hill Book Co. NY , 1984.
- [LIU89] Liu , Lun-Chun e Horowitz , Ellis - "A Formal Model for Software Project Management" - IEEE Transactions on Software Engineering , Vol.15 n. 10. , Outubro 1989
- [LOP89] Maritza Rodríguez de López - "Um Analisador de Especificações Operacionais" - Dissertação apresentada ao Instituto de Ciencia Matematicas de São Carlos , USP , como parte dos requisitos para obtenção do titulo de Mestre em Ciencias de Computação e Matemática Computacional , Dezembro 1989
- [MAS43] Maslow , A.H. - "A Theory of Humam Motivation" - Psychology Rev. n. 50 , 1943
- [MAS54] Maslow , A.H. - "Motivation and Personality" - Harpeer & Row , NY , 1954
- [MAT81] Mantel , M. "The Effect of Programming Team Structures on Programming Tasks" - Cmmunications of the ACM , Março 1981 pags 106-113
- [NET90] Manoel Gomes de Mendonça Neto - "Um Ambiente Baseado em Conhecimento para Suporte ao Planejamento do Desenvolvimento de Software" - Tese de Mestrado , FEE - UNICAMP 1990
- [PEN85] Maria H.Penedo e E.Don Stucke - "PMDB - A Project Master Database for Software Engineering Environments" - CH2139 IEEE , Abril de 1985 , pp. 150-157
- [PEN88] PENEDO , MARIA HELOISA IEEE TRANSACTIONS ON SOFTWARE ENGINEERING VOL.14 NO.6 JUNHO 1988 PAG 689-695
- [RAD85] D.R.Addleman,M.J.Davis e P.E.Presson - Specification Technology Guidebook , RADG-TR-85-135 , preparado por Boeing Aerospace Company para RADG , Griffis AFB ,NY , Agosto 1985

- [R0086] Controlling Software Projects - P.Rook ,Soft.Eng.Journal , Janeiro 1986 - pag.7-16 .
- [THA88] Thayer, Richard H. SEPM - A TopDown View - IEEE Tutorial: Software Engineering Project Mngement pags.15-53 1988
- [TRA86] Caetano Traina Jr. - "Máquina e modelo de dados dedicados para aplicações de engenharia" - Tese apresentada ao IFQSC-USP para obtenção do título de Doutor , São Carlos , Dezembro 1986
- [TRA88] Caetano Traina Jr. e J.F.W.Staets - "Um Modelo de Representação de Objetos" - Anais do III Simpósio Brasileiro de Bancos de Dados , Recife , Março 1988
- [WEI71] Weinberg , G. - "The Psychology of Computer Programming" - Van Nostrand Reinhold , NY 1971
- [W0088] Wood , L. - "Product Focus : The Promise of Project Management" - Revista Byte , pp. 180 a 192 , Novembro 1988
- [YOU76] Yourdon , E. - "Managing the Structured Technique" - Prentice-Hall , Englewood Cliffs , NJ 1976
- [YOU77] Youker , R. - "Organizational Alternatives for Project Management " - Project Management Journal , Março 1977 - pags 106 - 113 .

ANEXO 1 - LINGUAGEM DE DESCRIÇÃO DO MODELO PROPOSTO

>CLASSES

modelo, formulario, sistema, analise-estruturada, petri ,
gerenciamento,projeto

>OBJETOS

modelo,global
formulario,modelo
bloco,formulario
cabecalho,formulario
rodape,formulario
item-elementar,formulario
item-de-menu,formulario
condicao,formulario
item-de-escolha,formulario
relacionamento-automatico,formulario
diagrama-petri, sistema
lugar, petri
transicao, petri
analise-estruturada,sistema
diagrama-fd,analise-estruturada
entidade-externa,analise-estruturada
funcao,analise-estruturada
fluxo-de-dados,analise-estruturada
arquivo-logico,analise-estruturada
estrutura-d,analise-estruturada
elemento-d,analise-estruturada
orgao,global
termo,global
sistema,global
cliente,global
ambiente-de-gerenciamento,global
tipo-de-ciclo-de-vida,gerenciamento
fases,gerenciamento,
tipos-de-atividades,gerenciamento
metodologias,gerenciamento
categorias-recursos-materiais,gerenciamento
categorias-recursos-humanos,gerenciamento
pessoas,gerenciamento
projeto-eng-soft,gerenciamento
atividade,projeto

>RELACIONAMENTOS

insere op e-inserido
consiste-de op consiste
forma op formado-por
acessa op acessado-pelo
condiciona op e-condicionado-por
escolhe op e-escolhido-pelo
e-do-tipo op e-tipo-de
faz-referencia-a op tem-referencia
aparece-em op mostra
inibe op inibida-por
envia-marca-para op recebe-marca-de
ativa op ativada-por
compoe op composto-por
recebe op recebido-por

gera op gerado-por
constitui op constituído-por
representa op representado-por
responsavel op e-de-responsabilidade-de
pertence-a op possui
explica op explicado-por
informa op informado-por
e-gerenciado-por op gerencia
associa op esta-associada
coordena op e-coordenada-por
aloca op alocado-por
sucede op precede
precisa-de op e-necessario-para
define op e-definido

>ATRIBUTOS

repeticao,1,5,2,sim,nao
posicao-coluna,1,2
posicao-linha,1,2
tipo-campo,1,5,7,objeto,relacionamento,atributo,comentario,sinonimo,atrib
tamanho,1,2
leitura,1,5,2,ler,ler-escrever
operador,1,5,8,igual,diferente,maior-que,menor-que,maior-igual,menor-igu
limite-simulacao,1,2
limite-ativacoes,1,2
intervalo-observacao,1,2
marca-inicial,1,2
marca-corrente,1,2
n-max-marcas,1,2
num-pontos,2,10,3
sub-tipo,1,2
peso,1,2
peso-posicao,1,2
tempo-inic-ativacao,1,2
valor-temporizacao,1,2
lim-inf-temporizacao,1,2
lim-sup-temporizacao,1,2
n-max-ativacoes,1,2
posicao,2,10,3
posicao-nome,2,10,3
execucao,1,5,2>manual,automatizada
quantidade,1,2
periodo,1,1
periodicidade,1,1
ocorrencia-maxima,1,2
ocorrencia-minima,1,2
vol-estimado,1,2
insercoes,1,2
eliminacoes,1,2
alteracoes,1,2
posicao-total,1,2
posicao-decimal,1,2
caracteristica,1,5,3,numerico,alfabetico,alfanumerico
unidade,1,1

lim-minimo,1,1
 lim-maximo,1,1
 local,1,1
 nome,1,2
 data-inicio-mais-cedo,1,2
 data-inicio-mais-tarde,1,2
 data-termino-mais-cedo,1,2
 data-termino-mais-tarde,1,2
 data-inicio-ferias,1,2
 duracao-ferias,1,2
 estado,1,2
 unidade-producao,1,2
 quant-unidades-a-serem-prod,1,2
 quant-unidades-produzidas,1,2
 recursos-humanos-previstos,1,2
 recursos-humanos-gastos,1,2
 recursos-materiais-previstos,1,2
 recursos-materiais-gastos,1,2
 data,1,2
 duracao-prevista,1,2
 folga-livre,1,2
 folga-total,1,2
 folga-independente,1,2
 quant-disponivel-por-periodo,1,2
 tipo-de-objeto,1,2
 custo,1,2
 custo-unitario,1,2
 z-recursos-humanos-periodo,1,2
 z-recursos-materiais-periodo,1,2
 z-ultimo-periodo-apurado,1,2
 z-dias-em-execucao,1,2
 z-quant-unidades-periodo,1,2

>COMENTARIOS

texto-precedente
 operando
 auxiliar-operando
 valor-pai-especial
 nome-tipo
 nome-tipo-obj-destino
 texto-selecao
 valor-da-escolha
 texto-de-menu
 texto-de-help
 descricao,observacoes
 composicao,val-discretos
 glossario
 recursos-humanos-alocados
 recursos-materiais-disp
 produto

>TIPO-SINONIMO

referencia, sinonimo
codigo
sigla

>CLASSES-CHAVES

sistema, modelo
formulario, analise-estruturada
petri, gerenciamento, projeto

>FIM-NOMES

>CARAC-CLASSES-CHAVES

chave sistema
 fecha analise-estruturada, petri
 abre sistema
 identificador global
chave modelo
 fecha formulario
 abre modelo
 identificador global
chave formulario
 fecha formulario
 abre formulario
 identificador modelo
chave analise-estruturada
 fecha analise-estruturada
 abre analise-estruturada
 identificador sistema
chave petri
 fecha petri
 abre petri
 identificador sistema
chave gerenciamento
 fecha projeto
 abre gerenciamento
 identificador global
chave projeto
 fecha projeto
 abre projeto
 identificador gerenciamento

>SINTAXE-REL

objeto formulario
 insere relacionamento-automatico
 consiste-de bloco, item-elementar
objeto relacionamento-automatico
 e-inserido formulario
objeto bloco
 consiste formulario, bloco
 consiste-de bloco, item-elementar
 acessado-pelo item-de-menu
 forma cabecalho, rodape
 e-condicionado-por condicao
objeto cabecalho
 formado-por bloco, item-elementar
objeto rodape
 formado-por bloco, item-elementar

objeto item-elementar
 consiste formulario, bloco
 acessado-pelo item-de-menu
 forma cabecalho, rodape
 e-do-tipo item-elementar
 e-tipo-de item-elementar
 faz-referencia-a item-elementar
 tem-referencia item-elementar
 e-condicionado-por condicao
 escolhe item-de-escolha
objeto condicao
 condiciona bloco, item-elementar
objeto item-de-escolha
 e-escolhido-pelo item-elementar
objeto item-de-menu
 acessa item-elementar, bloco
objeto diagrama-petri
 mostra lugar
 mostra transicao
objeto lugar
 aparece-em diagrama-petri
 ativa transicao
 recebe-marca-de transicao
 inibe transicao
objeto transicao
 aparece-em diagrama-petri
 ativada-por lugar
 envia-marca-para lugar
 inibida-por lugar
objeto diagrama-fd
 mostra funcao, fluxo-de-dados
 mostra entidade-externa, arquivo-logico
 representa funcao
objeto entidade-externa
 aparece-em diagrama-fd
 recebe fluxo-de-dados
 gera fluxo-de-dados
 e-de-responsabilidade-de orgao
 pertence-a sistema
 informado-por cliente
objeto funcao
 compoe funcao
 composto-por funcao
 recebe fluxo-de-dados
 gera fluxo-de-dados
 representado-por diagrama-fd
 aparece-em diagrama-fd
 e-de-responsabilidade-de orgao
 pertence-a sistema

objeto fluxo-de-dados
gerado-por funcao
recebido-por funcao,arquivo-logico,entidade-externa
constituído-por estrutura-d,elemento-d
gerado-por arquivo-logico
gerado-por entidade-externa
aparece-em diagrama-fd
e-de-responsabilidade-de orgao
pertence-a sistema
objeto arquivo-logico
aparece-em diagrama-fd
gera fluxo-de-dados
recebe fluxo-de-dados
constituído-por estrutura-d,elemento-d
e-de-responsabilidade-de orgao
pertence-a sistema
objeto estrutura-d
constitui arquivo-logico, fluxo-de-dados, estrutura-d
constituído-por elemento-d, estrutura-d
e-de-responsabilidade-de orgao
informado-por cliente
objeto elemento-d
constitui arquivo-logico
constitui estrutura-d,fluxo-de-dados
e-de-responsabilidade-de orgao
informado-por cliente
objeto orgao
responsavel fluxo-de-dados,funcao
responsavel arquivo-logico,termo
responsavel elemento-d,entidade-externa
responsavel estrutura-d
objeto sistema
possui fluxo-de-dados,arquivo-logico
possui entidade-externa,funcao
explicado-por termo
objeto cliente
informa termo,elemento-d
informa entidade-externa,estrutura-d
objeto termo
e-de-responsabilidade-de orgao
informado-por cliente
explica sistema
objeto AMBIENTE-DE-GERENCIAMENTO
define tipo-de-ciclo-de-vida,categorias-recursos-humanos,categorias-rec
possui pessoas
objeto TIPOS-DE-ATIVIDADES
e-definido fases
associa metodologias
possui atividade
sucede tipos-de-atividades
precede tipos-de-atividades
objeto PROJETO-ENG-SOFT
possui atividade,tipo-de-ciclo-de-vida
e-gerenciado-por pessoas

objeto FASES
 e-definido tipo-de-ciclo-de-vida
 constituído-por fases
 constitui fases
 define tipos-de-atividades
 associa metodologias
 objeto TIPO-DE-CICLO-DE-VIDA
 pertence-a projeto-eng-soft
 e-definido ambiente-de-gerenciamento
 define fases
 objeto CATEGORIAS-RECURSOS-HUMANOS
 alocado-por atividade
 e-definido ambiente-de-gerenciamento
 possui pessoas
 objeto ATIVIDADE
 pertence-a projeto-eng-soft
 e-coordenada-por pessoas
 aloca categorias-recursos-humanos
 sucede atividade
 precede atividade
 precisa-de categorias-recursos-materiais
 pertence-a tipos-de-atividades
 objeto CATEGORIAS-RECURSOS-MATERIAIS
 e-necessario-para atividade
 e-definido ambiente-de-gerenciamento
 objeto METODOLOGIAS
 esta-associada fases,tipos-de-atividades
 objeto PESSOAS
 gerencia projeto-eng-soft,pessoas
 coordena atividade
 pertence-a categorias-recursos-humanos,ambiente-de-gerenciamento
 e-gerenciado-por pessoas
 >SINTAXE-ATR
 objeto bloco
 repeticao
 objeto item-de-menu
 posicao-coluna
 objeto item-elementar
 tipo-campo, posicao-coluna, posicao-linha ,tamanho, leitura, repeticao
 objeto condicao
 operador
 objeto diagrama-petri
 limite-simulacao
 limite-ativacoes
 intervalo-observacao
 objeto lugar
 marca-inicial
 marca-corrente
 n-max-marcas

objeto transicao
 sub-tipo
 valor-temporizacao
 tempo-inic-ativacao
 lim-inf-temporizacao
 lim-sup-temporizacao
 n-max-ativacoes
 objeto funcao
 execucao, periodicidade, local
 objeto fluxo-de-dados
 quantidade, periodo
 objeto arquivo-logico
 vol-estimado, insercoes, eliminacoes
 alteracoes
 objeto elemento-d
 posicao-total, posicao-decimal, caracteristica
 unidade, lim-minimo, lim-maximo
 objeto CATEGORIAS-RECURSOS-MATERIAIS
 custo-unitario
 objeto ATIVIDADE
 data-inicio-mais- cedo
 data-inicio-mais- tarde
 data-termino-mais- cedo
 data-termino-mais- tarde
 estado
 unidade-producao
 quant-unidades-a-serem-prod
 quant-unidades-produzidas
 data
 duracao-prevista
 folga-livre
 folga-total
 folga-independente
 z-dias-em-execucao
 z-quant-unidades-periodo
 objeto CATEGORIAS-RECURSOS-HUMANOS
 custo-unitario
 objeto PROJETO-ENG-SOFT
 custo
 data-inicio-mais- cedo
 data-inicio-mais- tarde
 data-termino-mais- cedo
 data-termino-mais- tarde
 z-ultimo-periodo-apurado
 objeto PESSOAS
 quant-disponivel-por-periodo
 data-inicio-ferias
 duracao-ferias

>SINTAXE-COM

objeto bloco
 texto-precedente
objeto item-de-menu
 texto-de-menu, texto-de-help
objeto item-elementar
 nome-tipo, texto-precedente, nome-tipo-obj-destino
objeto condicao
 operando, valor-pai-especial, auxiliar-operando
objeto item-de-escolha
 texto-selecao, valor-da-escolha
objeto diagrama-petri
 descricao
objeto lugar
 descricao
objeto transicao
 descricao
objeto entidade-externa
 descricao
objeto funcao
 descricao, observacoes
objeto fluxo-de-dados
 descricao, observacoes
objeto arquivo-logico
 descricao, observacoes
objeto estrutura-d
 descricao, observacoes
objeto elemento-d
 descricao, composicao, val-discretos
 observacoes
objeto termo
 glossario
objeto PROJETO-eng-soft
 descricao
objeto tipo-de-ciclo-de-vida
 descricao
objeto ambiente-de-gerenciamento
 descricao
objeto CATEGORIAS-RECURSOS-MATERIAIS
 descricao, recursos-materiais-disp
objeto metodologias
 descricao
objeto FASES
 descricao, produto
objeto atividade
 produto
objeto PESSOAS
 recursos-humanos-alocados

>SINTAXE-SIN

objeto lugar
referencia
objeto transicao
referencia
objeto diagrama-fd
referencia
objeto entidade-externa
referencia
objeto funcao
referencia
objeto arquivo-logico
referencia
objeto estrutura-d
sinonimo
objeto elemento-d
sinonimo

>SINTAXE-ATR-REL

relacionamento inibe
posicao, num-pontos
relacionamento envia-marca-para
posicao, peso, peso-posicao, num-pontos
relacionamento ativa
posicao, peso, peso-posicao, num-pontos
relacionamento aparece-em
posicao, posicao-nome, num-pontos
relacionamento constituido-por
ocorrencia-maxima, ocorrencia-minima
relacionamento E-NECESSARIO-PARA
recursos-materiais-previstos, recursos-materiais-gastos, z-recursos-mater
relacionamento PRECISA-DE
recursos-materiais-previstos, recursos-materiais-gastos, z-recursos-mater
relacionamento ALOCA
recursos-humanos-previstos, recursos-humanos-gastos, z-recursos-humanos-p
relacionamento ALOCADO-POR
recursos-humanos-previstos, recursos-humanos-gastos, z-recursos-humanos-p
relacionamento associa
tipo-de-objeto
relacionamento esta-associada
tipo-de-objeto

>FIM-SINTAXE

>METODOLOGIA, MODELO-FORMULARIO, 1

>OBJETOS-METODOLOGIA

bloco
cabecalho
condicao
formulario
item-elementar
item-de-escolha
item-de-menu
relacionamento-automatico
rodape

>PAL-ESP-METODOLOGIA

acessado-pelo
auxiliar-operando
consiste-de
e-do-tipo
escolhe
e-condicionado-por
faz-referencia-a
forma
insere
leitura
nome-tipo
operador
operando
posicao-coluna
posicao-linha
repeticao
tamanho
texto-de-help
texto-de-menu
texto-selecao
texto-precedente
tipo-campo
nome-tipo-obj-destino
valor-da-escolha
valor-pal-especial

>METODOLOGIA, DFD, 2

>OBJETOS-METODOLOGIA

arquivo-logico
cliente
diagrama-fd
elemento-d
entidade-externa
estrutura-d
fluxo-de-dados
funcao
orgao
sistema
termo

>PAL-ESP-METODOLOGIA

alteracoes
aparece-em
caracteristica
compos
composicao
composto-por
analise-estruturada
constitui
constituído-por
descricao
e-de-responsabilidade-de
eliminacoes
execucao
explica
explicado-por
gera
gerado-por
glossario
informa
informado-por
insercoes
lim-maximo
lim-minimo
local
mostra
observacoes
ocorrencia-maxima
ocorrencia-minima
periodicidade
periodo
pertence-a
posicao
posicao-decimal
posicao-nome
posicao-total
possui
quantidade
recebe
recebido-por
referencia
representa
representado-por
responsavel
sinonimo
unidade
val-discretos
vol-estimado
num-pontos

>METODOLOGIA, PETRI, 3

>OBJETOS-METODOLOGIA

diagrama-petri

lugar

transicao

sistema

>PAL-ESP-METODOLOGIA

aparece-em

ativa

ativada-por

petri

descricao

envia-marca-para

inibe

inibida-por

intervalo-observacao

lim-inf-temporizacao

lim-sup-temporizacao

limite-ativacoes

limite-simulacao

marca-inicial

marca-corrente

mostra

n-max-marcas

n-max-ativacoes

peso

peso-posicao

posicao

recebe-marca-de

referencia

sub-tipo

tempo-inic-ativacao

valor-temporizacao

num-pontos

>NOME-COMANDOS

editor-de-descricoes,2,10,ed_descr.nom

listador-de-descricoes,5,8,ed_list.nom

gerencia-projeto,6,2,ge_proj.nom

ANEXO 2 - LISTAGEM DOS FORMULÁRIOS

AMBIENTE DE GERENCIAMENTO

*-----

f1-b1-1 (BLOCO)

*-----

--- Relacionamentos do Objeto ---

ACESSADO-PELO

f1-imb-1

CONSISTE

ambiente-de-gerenciamento

CONSISTE-DE

f1-ie-3

f1-ie-4

f1-ie-5

f1-ie-6

--- Atributos do Objeto ---

REPETICAO

SIM

--- Comentarios do Objeto ---

TEXTO-PRECEDENTE

CATEGORIAS RECURSOS MATERIAIS

*-----

f1-b1-2 (BLOCO)

*-----

--- Relacionamentos do Objeto ---

ACESSADO-PELO

f1-imb-2

CONSISTE

ambiente-de-gerenciamento

CONSISTE-DE

f1-ie-7

f1-ie-8

--- Atributos do Objeto ---

REPETICAO

SIM

--- Comentarios do Objeto ---

TEXTO-PRECEDENTE

CATEGORIAS RECURSOS HUMANOS

*-----
f1-b1-3 (BLOCO)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

f1-imb-3

CONSISTE

ambiente-de-gerenciamento

CONSISTE-DE

f1-ie-9

f1-ie-10

--- Atributos do Objeto ---

REPETICAO

SIM

--- Comentarios do Objeto ---

TEXTO-PRECEDENTE

TIPOS DE CICLOS DE VIDA

*-----
f1-b1-4 (BLOCO)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

f1-imb-4

CONSISTE

ambiente-de-gerenciamento

CONSISTE-DE

f1-ie-11

f1-ie-12

f1-ie-13

f1-ie-14

f1-ie-15

f1-ie-16

--- Atributos do Objeto ---

REPETICAO

SIM

--- Comentarios do Objeto ---

TEXTO-PRECEDENTE

PESSOAS

*-----
f1-ie-1 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-1
CONSISTE
ambiente-de-gerenciamento

--- Atributos do Objeto ---

LEITURA
ler
POSICAO-COLUNA
1
POSICAO-LINHA
3
REPETICAO
nao
TAMANHO
60
TIPO-CAMPO
objeto-corrente

--- Comentarios do Objeto ---

NOME-TIPO
ambiente-de-gerenciamento
TEXTO-PRECEDENTE
NOME DO AMBIENTE ----- :

*-----
f1-ie-10 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-10
CONSISTE
f1-bl-3
FAZ-REFERENCIA-A
f1-ie-9

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---
NOME-TIPO
DESCRICAO
TEXTO-PRECEDENTE
DESCRICAO DO CICLO DE VIDA - :

*-----
f1-ie-11 (ITEM-ELEMENTAR)
*-----

--- Relacionamentos do Objeto ---
ACESSADO-PELO
f1-im-11
CONSISTE
f1-bi-4
TEM-REFERENCIA
f1-ie-12
f1-ie-13
f1-ie-14
f1-ie-15
f1-ie-16

--- Atributos do Objeto ---
LEITURA
ler-escrever
POSICAO-COLUNA
1
POSICAO-LINHA
-2
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
POSSUI
NOME-TIPO-OBJ-DESTINO
PESSOAS
TEXTO-PRECEDENTE
PESSOA ----- :

*-----
f1-ie-12 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-12
CONSISTE
f1-bl-4
FAZ-REFERENCIA-A
f1-ie-11

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---

NOME-TIPO
QUANT-DISPONIVEL-POR-PERIODO
TEXTO-PRECEDENTE
QUANT.DISPONIVEL POR PERIODO :

*-----
f1-ie-13 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-13
CONSISTE
f1-bl-4
FAZ-REFERENCIA-A
f1-ie-11

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---
NOME-TIPO
RECURSOS-HUMANOS-ALOCADOS
TEXTO-PRECEDENTE
RECURSOS HUMANOS ALOCADOS -- :

*-----
f1-ie-14 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
f1-im-14
CONSISTE
f1-bl-4
FAZ-REFERENCIA-A
f1-ie-11

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---
NOME-TIPO
DATA-INICIO-FERIAS
TEXTO-PRECEDENTE
DATA DE INICIO DE FERIVIAS - :

*-----
f1-ie-15 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-15
CONSISTE
f1-bi-4
FAZ-REFERENCIA-A
f1-ie-11

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---

NOME-TIPO
DURACAO-FERIAS
TEXTO-PRECEDENTE
DURACAO DAS FERIAS ----- :

*-----
f1-ie-16 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-16
CONSISTE
f1-bi-4
FAZ-REFERENCIA-A
f1-ie-11

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
E-GERENCIADO-POR
NOME-TIPO-OBJ-DESTINO
PESSOAS
TEXTO-PRECEDENTE
NOME DO GERENTE ----- :

*-----
f1-ie-2 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
f1-im-2
CONSISTE
ambiente-de-gerenciamento

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-2
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---
NOME-TIPO
DESCRICAO
TEXTO-PRECEDENTE
DESCRICAO DO AMBIENTE ----- :

*-----
f1-ie-3 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
f1-im-3
CONSISTE
f1-bl-1
TEM-REFERENCIA
f1-ie-4
f1-ie-5
f1-ie-6

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-2
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
DEFINE
NOME-TIPO-OBJ-DESTINO
CATEGORIAS-RECURSOS-MATERIAIS
TEXTO-PRECEDENTE
CAT.REC.MATERIAIS ----- :

*-----
f1-ie-4 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
f1-im-4
CONSISTE
f1-bl-1
FAZ-REFERENCIA-A
f1-ie-3

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---
NOME-TIPO
DESCRICAO
TEXTO-PRECEDENTE
DESCRICAO DO REC.MATERIAL -- :

*-----
f1-ie-5 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-5
CONSISTE
f1-bi-1
FAZ-REFERENCIA-A
f1-ie-3

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---

NOME-TIPO
CUSTO-UNITARIO
TEXTO-PRECEDENTE
CUSTO UNITARIO ----- :

*-----
f1-ie-6 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-6
CONSISTE
f1-bi-1
FAZ-REFERENCIA-A
f1-ie-3

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---
NOME-TIPO
RECURSOS-MATERIAIS-DISP
TEXTO-PRECEDENTE
DISPONIBILIDADE REC.MATERIAIS:

*-----
f1-ie-7 (ITEM-ELEMENTAR)
*-----

--- Relacionamentos do Objeto ---
ACESSADO-PELO
f1-im-7
CONSISTE
f1-bl-2
TEM-REFERENCIA
f1-ie-8

--- Atributos do Objeto ---
LEITURA
ler-escrever
POSICAO-COLUNA
1
POSICAO-LINHA
-2
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
DEFINE
NOME-TIPO-OBJ-DESTINO
CATEGORIAS-RECURSOS-HUMANOS
TEXTO-PRECEDENTE
CATEGORIA RECURSO HUMANO --- :

*-----
f1-ie-8 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-8
CONSISTE
f1-bl-2
FAZ-REFERENCIA-A
f1-ie-7

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---

NOME-TIPO
CUSTO-UNITARIO
TEXTO-PRECEDENTE
CUSTO UNITARIO ----- :

*-----
f1-ie-9 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f1-im-9
CONSISTE
f1-bl-3
TEM-REFERENCIA
f1-ie-10

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-2
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
DEFINE
NOME-TIPO-OBJ-DESTINO
TIPO-DE-CICLO-DE-VIDA
TEXTO-PRECEDENTE
TIPO DE CICLO DE VIDA ----- :

*-----
f1-im-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---
ACESSA
f1-ie-1

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
AMBIENTE DE GERENCIAMENTO

*-----
f1-im-10 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---
ACESSA
f1-ie-10

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
DESCRICAO CICLO DE VIDA

*-----
f1-im-11 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---
ACESSA
f1-ie-11

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
PESSOA

*-----
f1-im-12 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-ie-12

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
DISPONIBILIDADE POR PERIODO

*-----
f1-im-13 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-ie-13

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
ALOCACAO

*-----
f1-im-14 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-ie-14

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
INICIO DE FERIAS

*-----
f1-im-15 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-ie-15

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
DURACAO DAS FERIAS

*-----

f1-im-16 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-ie-16

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
NOME DO GERENTE

*-----

f1-im-2 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-ie-2

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
DESCRICAO DO AMBIENTE

*-----
f1-1m-3 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-1e-3

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
CATEGORIA REC.MATERIAL

*-----
f1-1m-4 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-1e-4

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
DESCRICAO REC.MATERIAL

*-----
f1-1m-5 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-1e-5

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
CUSTO UNITARIO REC.MAT.

*-----
f1-im-6 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-ie-6

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
DISPONIBILIDADE REC.MAT.

*-----
f1-im-7 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-ie-7

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
CATEGORIA REC.HUMANO

*-----
f1-im-8 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-ie-8

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
CUSTO UNITARIO REC.HUM.

*-----
f1-im-9 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-ie-9

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
TIPO DE CICLO DE VIDA

*-----
f1-imb-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-b1-1

--- Atributos do Objeto ---

POSICAO-COLUNA
1

--- Comentarios do Objeto ---

TEXTO-DE-MENU
***** RECURSOS MATERIAIS *****

*-----
f1-imb-2 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
f1-b1-2

--- Atributos do Objeto ---

POSICAO-COLUNA
1

--- Comentarios do Objeto ---

TEXTO-DE-MENU
***** RECURSOS HUMANOS *****

*-----
f1-imb-3 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-b1-3

--- Atributos do Objeto ---

POSICAO-COLUNA
1

--- Comentarios do Objeto ---

TEXTO-DE-MENU
***** CICLOS DE VIDA *****

*-----
f1-imb-4 (ITEM-DE-MENU)

*-----

--- Relacionamentos do Objeto ---

ACESSA
f1-b1-4

--- Atributos do Objeto ---

POSICAO-COLUNA
1

--- Comentarios do Objeto ---

TEXTO-DE-MENU
***** PESSOAS *****

CATEGORIAS-RECURSOS-HUMANOS

*-----
F2-BL-1 (BLOCO)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
f2-IMB-1
CONSISTE
categorias-recursos-humanos
CONSISTE-DE
F2-IE-3
F2-IE-4
F2-IE-5
F2-IE-6
F2-IE-7
F2-IE-8

--- Atributos do Objeto ---

REPETICAO
SIM

--- Comentarios do Objeto ---

TEXTO-PRECEDENTE
PESSOAS NESTA CATEGORIA ***

*-----
F2-IE-1 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
F2-IM-1
CONSISTE
categorias-recursos-humanos

--- Atributos do Objeto ---

LEITURA
LER
POSICAO-COLUNA
1
POSICAO-LINHA
3
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
OBJETO-CORRENTE

--- Comentarios do Objeto ---
NOME-TIPO
CATEGORIAS-RECURSOS-HUMANOS
TEXTO-PRECEDENTE
CATEGORIA DE RECURSO HUMANO :

*-----
F2-IE-2 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F2-IM-2
CONSISTE
categorias-recursos-humanos

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-2
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---
NOME-TIPO
CUSTO-UNITARIO
TEXTO-PRECEDENTE
CUSTO UNITARIO ----- :

*-----
F2-IE-3 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F2-IM-3
CONSISTE
F2-BL-1
TEM-REFERENCIA
F2-IE-4
F2-IE-5
F2-IE-6
F2-IE-7
F2-IE-8

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-2
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
POSSUI
NOME-TIPO-OBJ-DESTINO
PESSOAS
TEXTO-PRECEDENTE
PESSOA ----- :

*-----
F2-IE-4 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F2-IM-4
CONSISTE
F2-BL-1
FAZ-REFERENCIA-A
F2-IE-3

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---
NOME-TIPO
QUANT-DISPONIVEL-POR-PERIODO
TEXTO-PRECEDENTE
QUANT.DISPONIVEL POR PERIODO :

*-----
F2-IE-5 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
F2-IM-5
CONSISTE
F2-BL-1
FAZ-REFERENCIA-A
F2-IE-3

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---

NOME-TIPO
RECURSOS-HUMANOS-ALOCADOS
TEXTO-PRECEDENTE
RECURSOS HUMANOS ALOCADOS -- :

*-----
F2-IE-6 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
F2-IM-6
CONSISTE
F2-BL-1
FAZ-REFERENCIA-A
F2-IE-3

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---
NOME-TIPO
DATA-INICIO-FERIAS
TEXTO-PRECEDENTE
DATA DE INICIO DE FERIAS --- :

*-----
F2-IE-7 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F2-IM-7
CONSISTE
F2-BL-1
FAZ-REFERENCIA-A
F2-IE-3

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO

--- Comentarios do Objeto ---
NOME-TIPO
DURACAO-FERIAS
TEXTO-PRECEDENTE
DURACAO DAS FERIAS ----- :

*-----
F2-IE-8 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F2-IM-8
CONSISTE
F2-BL-1
FAZ-REFERENCIA-A
F2-IE-3

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---

NOME-TIPO
E-GERENCIADO-POR
NOME-TIPO-OBJ-DESTINO
PESSOAS
TEXTO-PRECEDENTE
NOME DO GERENTE ----- :

*-----
F2-IM-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-1

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
CATEGORIA REC. HUMANO

*-----
F2-IM-2 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-2

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
CUSTO UNITARIO REC.HUM.

*-----
F2-IM-3 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-3

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
PESSOA

*-----
F2-IM-4 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-4

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
DISPONIBILIDADE POR PERIODO

*-----
F2-IM-5 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-5

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
ALOCACAO

*-----
F2-IM-6 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-6

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
INICIO DAS FERIAS

*-----
F2-IM-7 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-7

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
DURACAO DAS FERIAS

*-----
F2-IM-8 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-IE-8

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
NOME DO GERENTE

*-----
f2-IMB-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F2-BL-1

--- Atributos do Objeto ---

POSICAO-COLUNA
1

--- Comentarios do Objeto ---

TEXTO-DE-MENU
***** PESSOAS NESTA CATEGORIA

FASES

*-----
F4-BL-1 (BLOCO)
*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
f4-imb-1
CONSISTE
fases
CONSISTE-DE
F4-IE-5
F4-IE-6
F4-IE-7
F4-IE-8
F4-IE-9

--- Atributos do Objeto ---
REPETICAO
SIM

--- Comentarios do Objeto ---
TEXTO-PRECEDENTE
TIPOS DE ATIVIDADES DESTA FASE

*-----
F4-IE-1 (ITEM-ELEMENTAR)
*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
f4-im-1
CONSISTE
fases

--- Atributos do Objeto ---
LEITURA
LER
POSICAO-COLUNA
1
POSICAO-LINHA
3
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
OBJETO-CORRENTE

--- Comentarios do Objeto ---
NOME-TIPO
FASES
TEXTO-PRECEDENTE
FASE ----- :

*-----
F4-IE-2 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

f4-im-2

CONSISTE

fases

--- Atributos do Objeto ---

LEITURA

LER-ESCREVER

POSICAO-COLUNA

1

POSICAO-LINHA

-2

REPETICAO

NAO

TAMANHO

5

TIPO-CAMPO

COMENTARIO

--- Comentarios do Objeto ---

NOME-TIPO

DESCRICAO

TEXTO-PRECEDENTE

DESCRICAO DA FASE ----- :

*-----
F4-IE-3 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

f4-im-3

CONSISTE

fases

--- Atributos do Objeto ---

LEITURA

LER-ESCREVER

POSICAO-COLUNA

1

POSICAO-LINHA

-2

REPETICAO

NAO

TAMANHO

5

TIPO-CAMPO

COMENTARIO

--- Comentarios do Objeto ---

NOME-TIPO

PRODUTO

TEXTO-PRECEDENTE

PRODUTO DA FASE ----- :

*-----
F4-IE-4 (ITEM-ELEMENTAR)

*-----

--- Relacionamentos do Objeto ---

ACESSADO-PELO

f4-IM-4

CONSISTE

fases

--- Atributos do Objeto ---

LEITURA

LER-ESCREVER

POSICAO-COLUNA

1

POSICAO-LINHA

-2

REPETICAO

NAO

TAMANHO

60

TIPO-CAMPO

RELACIONAMENTO

--- Comentarios do Objeto ---

NOME-TIPO

CONSTITUI

NOME-TIPO-OBJ-DESTINO

FASES

TEXTO-PRECEDENTE

FASE DE QUE FAZ PARTE ----- :

*-----
F4-IE-5 (ITEM-ELEMENTAR)

*-----

--- Relacionamentos do Objeto ---

ACESSADO-PELO

F4-IM-5

CONSISTE

F4-BL-1

TEM-REFERENCIA

F4-IE-6

F4-IE-7

F4-IE-8

F4-IE-9

--- Atributos do Objeto ---

LEITURA

LER-ESCREVER

POSICAO-COLUNA

1

POSICAO-LINHA

-2

REPETICAO

NAO

TAMANHO

60

TIPO-CAMPO

RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
DEFINE
NOME-TIPO-OBJ-DESTINO
TIPOS-DE-ATIVIDADES
TEXTO-PRECEDENTE
TIPO DE ATIVIDADE DESTA FASE

*-----
F4-IE-6 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F4-IM-6
CONSISTE
F4-BL-1
FAZ-REFERENCIA-A
F4-IE-5

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
SUCEDE
NOME-TIPO-OBJ-DESTINO
TIPOS-DE-ATIVIDADES
TEXTO-PRECEDENTE
TIPO DA ATIVIDADE PREDECESSORA

*-----
F4-IE-7 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
 F4-IM-7
CONSISTE
 F4-BL-1
FAZ-REFERENCIA-A
 F4-IE-5

--- Atributos do Objeto ---
LEITURA
 LER-ESCREVER
POSICAO-COLUNA
 1
POSICAO-LINHA
 -1
REPETICAO
 NAO
TAMANHO
 60
TIPO-CAMPO
 RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
PRECEDE
NOME-TIPO-OBJ-DESTINO
TIPOS-DE-ATIVIDADES
TEXTO-PRECEDENTE
TIPO DA ATIVIDADE SUCESSORA :

*-----
F4-IE-8 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
 F4-IM-8
CONSISTE
 F4-BL-1
FAZ-REFERENCIA-A
 F4-IE-5

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---

NOME-TIPO
ASSOCIA
NOME-TIPO-OBJ-DESTINO
METODOLOGIAS
TEXTO-PRECEDENTE
METODOLOGIA QUE ASSOCIA ---- :

*-----

F4-IE-9 (ITEM-ELEMENTAR)

*-----

--- Relacionamentos do Objeto ---

ACESSADO-PELO
F4-IM-9
CONSISTE
F4-BL-1
FAZ-REFERENCIA-A
F4-IE-5

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
ATRIBUTO-RELAC

--- Comentarios do Objeto ---

NOME-TIPO
TIPO-DE-OBJETO
TEXTO-PRECEDENTE
OBJETO DA METOD. ASSOCIADA - :

*-----
f4-im-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-1

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXT0-DE-MENU
FASE

*-----
f4-im-2 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-2

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXT0-DE-MENU
DESCRICAO DA FASE

*-----
f4-im-3 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-3

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXT0-DE-MENU
PRODUTO DA FASE

*-----
f4-IM-4 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-4

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXT0-DE-MENU
FASE DE QUE FAZ PARTE

*-----
F4-IM-5 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-5

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXT0-DE-MENU
TIPOS ATIVIDADES DA FASE

*-----
F4-IM-6 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-6

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXT0-DE-MENU
TIPO ATIVIDADE PREDECESSORA

*-----
F4-IM-7 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-7

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
TIPO ATIVIDADE SUCESSORA

*-----
F4-IM-8 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-8

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
METODOLOGIA ASSOCIADA

*-----
F4-IM-9 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-IE-9

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
TIPO OBJETO ASSOCIADO

*-----
f4-imb-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F4-BL-1

--- Atributos do Objeto ---

POSICAO-COLUNA
1

--- Comentarios do Objeto ---

TEXTO-DE-MENU
TIPO ATIVIDADES DA FASE ****

TIPO-DE-CICLO-DE-VIDA

*-----
F3-BL-1 (BLOCO)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

F3-IMB-1

CONSISTE

tipo-de-ciclo-de-vida

CONSISTE-DE

F3-IE-3

F3-IE-4

F3-IE-5

F3-IE-6

F3-IE-7

--- Atributos do Objeto ---

REPETICAO

SIM

--- Comentarios do Objeto ---

TEXTO-PRECEDENTE

FASES DO CICLO DE VIDA

*-----
F3-IE-1 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

F3-IM-1

CONSISTE

tipo-de-ciclo-de-vida

--- Atributos do Objeto ---

LEITURA

LER

POSICAO-COLUNA

1

POSICAO-LINHA

3

REPETICAO

NAO

TAMANHO

60

TIPO-CAMPO

OBJETO-CORRENTE

--- Comentarios do Objeto ---

NOME-TIPO

TIPO-DE-CICLO-DE-VIDA

TEXTO-PRECEDENTE

TIPO DE CICLO DE VIDA ----- :

*-----
F3-IE-2 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

F3-IM-2

CONSISTE

tipo-de-ciclo-de-vida

--- Atributos do Objeto ---

LEITURA

LER-ESCREVER

POSICAO-COLUNA

1

POSICAO-LINHA

-2

REPETICAO

NAO

TAMANHO

5

TIPO-CAMPO

COMENTARIO

--- Comentarios do Objeto ---

NOME-TIPO

DESCRICAO

TEXTO-PRECEDENTE

DESCRICAO DO GILO DE VIUIDA :

*-----
F3-IE-3 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO

F3-IM-3

CONSISTE

F3-BL-1

TEM-REFERENCIA

F3-IE-4

F3-IE-5

F3-IE-6

F3-IE-7

--- Atributos do Objeto ---

LEITURA

LER-ESCREVER

POSICAO-COLUNA

1

POSICAO-LINHA

-2

REPETICAO

NAO

TAMANHO

60

TIPO-CAMPO

RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
DEFINE
NOME-TIPO-OBJ-DESTINO
FASES
TEXTO-PRECEDENTE
FASE ----- :

*-----
F3-IE-4 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F3-IM-4
CONSISTE
F3-BL-1
FAZ-REFERENCIA-A
F3-IE-3

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---
NOME-TIPO
DESCRICAO
TEXTO-PRECEDENTE
DESCRICAO DA FASE ----- :

*-----
F3-IE-5 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F3-IM-5
CONSISTE
F3-BL-1
FAZ-REFERENCIA-A
F3-IE-3

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
5
TIPO-CAMPO
COMENTARIO

--- Comentarios do Objeto ---
NOME-TIPO
PRODUTO
TEXTO-PRECEDENTE
PRODUTO DESTA FASE ----- :

*-----
F3-IE-6 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---
ACESSADO-PELO
F3-IM-6
CONSISTE
F3-BL-1
FAZ-REFERENCIA-A
F3-IE-3

--- Atributos do Objeto ---
LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---
NOME-TIPO
CONSTITUI
NOME-TIPO-OBJ-DESTINO
FASES
TEXTO-PRECEDENTE
FASE DE QUE FAZ PARTE ----- :

*-----
F3-IE-7 (ITEM-ELEMENTAR)

*-----
--- Relacionamentos do Objeto ---

ACESSADO-PELO
F3-IM-7
CONSISTE
F3-BL-1
FAZ-REFERENCIA-A
F3-IE-3

--- Atributos do Objeto ---

LEITURA
LER-ESCREVER
POSICAO-COLUNA
1
POSICAO-LINHA
-1
REPETICAO
NAO
TAMANHO
60
TIPO-CAMPO
RELACIONAMENTO

--- Comentarios do Objeto ---

NOME-TIPO
ASSOCIA
NOME-TIPO-OBJ-DESTINO
METODOLOGIAS
TEXTO-PRECEDENTE
METODOLOGIA QUE ASSOCIA ---- :

*-----
F3-IM-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F3-IE-1

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
TIPO DE CICLO DE VIDA

*-----
F3-IM-2 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F3-IE-2

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
DESCRICAO DO CICLO DE VIDA

*-----
F3-IM-3 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F3-IE-3

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
FASE

*-----
F3-IM-4 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F3-IE-4

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---

TEXTO-DE-MENU
DESCRICAO DE FASE

*-----
F3-IM-5 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---

ACESSA
F3-IE-5

--- Atributos do Objeto ---

POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
PRODUTO DA FASE

*-----
F3-IM-6 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---
ACESSA
F3-IE-6

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
FASE DE QUE FAZ PARTE

*-----
F3-IM-7 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---
ACESSA
F3-IE-7

--- Atributos do Objeto ---
POSICAO-COLUNA
3

--- Comentarios do Objeto ---
TEXTO-DE-MENU
METODOLOGIA ASSOCIADA

*-----
F3-IMB-1 (ITEM-DE-MENU)

*-----
--- Relacionamentos do Objeto ---
ACESSA
F3-BL-1

--- Atributos do Objeto ---
POSICAO-COLUNA
1

--- Comentarios do Objeto ---
TEXTO-DE-MENU
***** FASES DO CICLO DE VIDA