

## UM SIMULADOR DE PROTOCOLOS DE COMUNICAÇÃO PARA A LINGUAGEM ESTELLE

Aloysio de Castro Pinto Pedroza  
COPPE/UFRJ  
Programa de Engenharia Elétrica  
Centro de Tecnologia, Bloco H  
Cidade Universitária  
Caixa Postal 68504  
21945 Rio de Janeiro RJ

Paulo Roberto Oliveira Valim  
Ministério da Marinha  
IPqM - Instituto de Pesquisas da Marinha  
Rua Ipirú s/n - Jardim Guanabara  
Ilha do Governador - RJ  
CEP - 21931

### RESUMO

O desenvolvimento de "software" de comunicação para redes de computadores deve seguir uma abordagem formal para que sejam maiores as chances de sucesso de todo o projeto. A linguagem ESTELLE é uma técnica para a descrição formal de sistemas de comunicação distribuídos criada pela ISO e este trabalho tem como objetivo apresentar um simulador desenvolvido para esta linguagem. Este simulador constitui o núcleo de um Sistema de Auxílio ao Projeto de Protocolos de Comunicação para Redes de Computadores.

### 1- INTRODUÇÃO

O desenvolvimento de "software" de comunicação para computadores conectados em rede deve ser feito seguindo uma abordagem formal para que o sistema resultante atinja os requisitos de segurança e qualidade previstos como meta do projeto. O uso de métodos de descrição formal na especificação de protocolos contribui de modo significativo para o sucesso de todo um projeto de "software" de comunicação. A linguagem ESTELLE [1] é uma das técnicas de descrição formal, ou FDT ("Formal Description Techniques"), apropriadas à descrição de "software" de comunicação distribuído; a outra técnica de descrição formal padronizada pela ISO é a linguagem LOTOS [2].

O projeto de protocolos de comunicação seguros deve ser estruturado segundo um certo número de fases que incluem a especificação formal do protocolo, a validação, a implementação e o teste final. É desejável que estas diferentes fases sejam integradas e automatizadas visando a redução do número de erros cometidos durante o projeto. O Simulador de Protocolos apresentado neste trabalho constitui o núcleo de um Sistema de Auxílio ao Projeto de Protocolos de Comunicação para Redes de Computadores que tem como base a linguagem ESTELLE e permite a validação por simulação dos protocolos especificados segundo este formalismo.

Os parágrafos seguintes apresentam uma breve descrição da linguagem ESTELLE, do Sistema de Auxílio ao Projeto de Protocolos ou "CAD para Protocolos" e do simulador de Protocolos para ESTELLE. O desenvolvimento do "CAD para Protocolos" faz parte dos trabalhos que vem sendo realizados no Programa de Engenharia Elétrica e no Núcleo de Robótica da COPPE referentes ao estudo de metodologias e ao desenvolvimento de ferramentas de auxílio ao projeto de protocolos para sistemas de computadores distribuídos.

### 2- A LINGUAGEM ESTELLE

A linguagem ESTELLE [1,5,6] possui características que a tornam apropriada ao uso em especificação de protocolos de comunicação para sistemas de computadores distribuídos. Entre estas características podem ser destacadas a modularidade, o uso de um modelo formal - as Máquinas de Estados Finitas Estendidas - como base para a descrição do comportamento de um conjunto de módulos cooperantes, a presença de mecanismos que permitem a troca de interações entre módulos, além da possibilidade de criação e destruição dinâmica de instâncias de módulos. A descrição da parte procedural do comportamento de um módulo é feita usando construções da linguagem PASCAL.

Uma especificação em ESTELLE é, portanto, constituída por um conjunto de módulos que trocam interações através de seus pontos de interação. Um conjunto de primitivas definidas para ESTELLE permite a criação, a interconexão e a destruição de instâncias de módulo.

Os principais conceitos de ESTELLE - módulo, pontos de interação e primitivas - além de um exemplo simples de especificação são apresentados nos próximos parágrafos.

#### 2.1- MÓDULO

O módulo é definido como um tipo; as instâncias de módulos são criadas e destruídas pelo uso de primitivas apropriadas de ESTELLE.

Um módulo é definido pela descrição de um cabeçalho (header), que caracteriza sua visibilidade externa, e pela descrição de um corpo (body), que caracteriza o comportamento de cada instância de módulo:

o cabeçalho contém as declarações dos pontos de interação externos, das variáveis exportadas e

dos parâmetros de inicialização, além da declaração dos atributos referentes à estruturação hierárquica entre módulos, detalhadas a seguir no parágrafo referente aos princípios de estruturação;

o corpo contém as declarações dos objetos internos ao módulo, tais como seus módulos filhos, as variáveis internas, os pontos de interação internos e o conjunto das transições que descrevem o comportamento do módulo considerado.

Os princípios de estruturação de ESTELLE estabelecem uma hierarquia entre os diversos módulos que compõem uma especificação: no topo desta hierarquia encontram-se os módulos Sistema cujos atributos são `systemprocess` e `systemactivity`, que podem ser refinados em módulos com atributos `process` e `activity` segundo o critério apresentado pela tabela 2.1.

A semântica de ESTELLE estabelece que não há execução em paralelo entre uma transição de uma instância de módulo e as transições de seus descendentes. Além disso, uma transição de uma dada instância de módulo tem prioridade em relação às transições das instâncias dos módulos seus descendentes.

## 2.2- PONTOS DE INTERAÇÃO

Os Pontos de Interação definem canais de comunicação entre os diversos módulos que compõem uma especificação em ESTELLE. Quando uma instância de módulo é criada, são associadas filas aos pontos de interações declarados para o módulo considerado.

## 2.3- PRIMITIVAS DE ESTELLE

As primitivas de ESTELLE permitem:

- . a criação (primitiva `init`) e destruição (primitiva `release`) de instâncias de módulos;
- . a conexão (primitiva `connect`) e desconexão (primitiva `disconnect`) de pontos de interação de instâncias de módulos com mesmo nível hierárquico;
- . a associação (primitiva `attach`) e desassociação (primitiva `detach`) de pontos de interação de instâncias de módulos com níveis hierárquicos adjacentes;
- . o envio de interações a um ponto de interação (primitiva `output`) e a recepção de interações (primitiva `when`) depositadas em um ponto de interação, isto é, em uma fila associada a um determinado ponto de interação.

## 2.4- UM EXEMPLO DE ESPECIFICAÇÃO EM ESTELLE

O emprego de ESTELLE em especificações de "software" concorrente pode ser mostrado através de um

exemplo simples em que um produtor envia mensagens (MSG) a um consumidor que controla a produção destas mensagens através de um sinal de liberação (LIB). A arquitetura a ser descrita em ESTELLE é apresentada na fig. 2.1 e a especificação completa é dada em seguida.

```

Specification Produtor_Consumidor systemprocess;
default individual queue; (* uma única fila é
                           associada a cada ponto de interação *)

Channel
  Interface_type (i_produtor, i_consumidor);
  by i_produtor:
    MSG(...); (* Produtor deposita MSG na interface *)
  by i_receptor:
    LIB; (* Consumidor deposita LIB na interface *)
(* definição dos módulos
    que compõem a especificação *)

Module Produtor_type process;
  ip Interface: Interface_type(i_produtor);
end;
Body Produtor_body for Produtor_type;
trans
  When Interface.LIB
  begin
    :
    (* construção da mensagem MSG *)
    output Interface.MSG(...); (* envio da
                                mensagem *)
  end;
end Produtor_body;
Module Consumidor_type process;
  ip Interface: Interface_type(i_consumidor);
end;
Body Consumidor_body for Consumidor_type;
initialize
  begin
    output Interface.LIB; (* liberação inicial
                          do produtor *)
  end;
trans
  When Interface.MSG(...) (* recepção de
                           mensagens *)
  begin
    :
    (* tratamento de mensagens *)
    output Interface.LIB (* liberação do envio de
                          uma nova mensagem *)
  end;
end Consumidor_body;
(* declaração das variáveis da especificação
correspondentes a cada tipo de módulo (header) *)
var Produtor: Produtor_type;
    Consumidor : Consumidor_type;
(* configuração da especificação *)
Initialize
  begin
    init Produtor with Produtor_body;
    init Consumidor with Consumidor_body;
    Connect Produtor.Interface to Consumidor.Interface
  end;
end Produtor_Consumidor.

```

### 3- O SISTEMA DE AUXILIO AO PROJETO

O Sistema de Auxilio ao Projeto de Protocolos, ou "CAD para Protocolos" [3], permitirá que o desenvolvimento de um protocolo de comunicação seja efetuado de maneira integrada, procurando fornecer um nível elevado de conforto ao projetista, em termos de recursos apropriados que estarão disponíveis.

A filosofia deste sistema, desenvolvida inicialmente em [4], tem como base a linguagem ESTELLE, uma vez que este é um formalismo padronizado pela ISO para a especificação de protocolos de comunicação entre computadores, que vem sendo bem aceito pela comunidade internacional [5].

O Sistema de Auxilio oferecerá uma interface amigável ao usuário permitindo:

- . a construção interativa de especificações de protocolos em ESTELLE;
- . a depuração interativa de protocolos;
- . a validação por simulação;
- . a implementação automatizada;
- . o teste final de implementação.

A figura 3.1 apresenta a arquitetura geral deste Sistema de Auxilio onde os seguintes módulos se destacam:

- . editor dedicado;
- . compilador ESTELLE;
- . gerador de sequências de teste;
- . implementador;
- . simulador;
- . validador/analizador de correção e desempenho;
- . interface homem-máquina.

O Editor de ESTELLE deverá facilitar ao máximo a tarefa de confecção de especificações de protocolos em ESTELLE. Com este objetivo, serão associados, sempre que possível, formas gráficas aos diversos conceitos representados pela linguagem como a modularidade, a hierarquia entre os diversos módulos, a interconexão entre módulos através de pontos de interação e as interações trocadas entre módulos. A edição da máquina de estados finita estendida que represente o comportamento de cada módulo será facilitada pela pré-impressão em tela de palavras chaves ou também pelo uso de representações gráficas associadas aos conceitos por elas representados. Será buscado sobretudo um equilíbrio entre a representação gráfica e a representação através de texto em ESTELLE, sendo sempre mantida a possibilidade de edição de especificações totalmente em texto ESTELLE. A saída do Editor será um código fonte em ESTELLE.

O Compilador ESTELLE se encarregará de efetuar a tradução do código fonte em ESTELLE numa Forma Intermediária [7] onde todos os conceitos de ESTELLE estão descritos de maneira estruturada, adequada como entrada comum dos módulos de geração de sequência de teste, de implementação, de simulação e

de análise. Na Forma Intermediária as estruturas de dados correspondentes aos diversos objetos ESTELLE encontram-se representadas como conjuntos e a parte procedural deduzida da especificação em ESTELLE encontra-se em MODULA-2. Assim sendo, deve-se fazer uso de um compilador MODULA-2 para obtenção do código executável correspondente à parte procedural.

O Gerador de Sequências de Teste, a partir do conhecimento do modelo de um protocolo dado por sua especificação em ESTELLE, gerará sequências de teste que serão usadas tanto na fase de simulação quanto na fase de teste final de implementação. Os trabalhos desenvolvidos por Sarikaya [8,9] são uma referência neste tópico.

O Implementador terá como função gerar código executável em uma dada máquina alvo a partir das informações fornecidas pela Forma Intermediária. A saída gerada pelo Implementador deve considerar a linguagem aceita e o sistema operacional disponível na máquina alvo. A arquitetura deste sub-sistema deve ser estruturada em módulos que dependem e em módulos que não dependem da máquina alvo. Uma opção pela geração de código fonte em uma linguagem de programação de sistemas, do tipo MODULA-2, permitirá uma maior independência em relação a uma dada máquina alvo.

O Simulador para ESTELLE executará o modelo de ESTELLE permitindo ao projetista um acesso a todos os objetos ativos em tempo de simulação. Este simulador é descrito com detalhes no parágrafo 4.

A arquitetura geral do Sistema de Auxilio prevê a ligação com outros sistemas dedicados à análise de modelos de sistemas concorrentes quanto ao aspecto de correção e de avaliação de desempenho. E o caso, por exemplo, da ligação com o sistema ANAPETRI [10] de análise de Redes de Petri desenvolvido no Programa de Engenharia Elétrica da COPPE e que permite a análise de propriedades de sistemas modelados em Redes de Petri. Outra possibilidade é a criação de um sub-sistema específico para o modelo de ESTELLE voltado para a análise de correção e do desempenho de protocolos de comunicação.

A Interface Homem-Máquina tem como objetivo estabelecer uma interação amigável com o usuário permitindo o fácil acesso e a integração entre os diferentes recursos do Sistema de Auxilio. Deve ser dada uma especial atenção à interação usuário/Sistema de Auxilio durante a fase de simulação de protocolos.

### 4- O SIMULADOR DE PROTOCOLOS

O simulador de protocolos permite o teste de um protocolo pela execução do conjunto de módulos que compõem a sua especificação em ESTELLE. A simulação pode ser utilizada para analisar o comportamento lógico e o desempenho [13]. Em sua primeira versão, o simulador considera somente os aspectos lógicos do protocolo especificado.

Durante a simulação, o modelo do protocolo testado é estimulado por sequências de primitivas de serviço para que sejam detectadas as condições de mau funcionamento do protocolo. Estas sequências devem ser escolhidas de modo a maximizar a possibilidade de detecção de erros de concepção e de especificação do protocolo [8,9].

As principais características, a arquitetura, a interface com o ambiente exterior e os detalhes de implementação referentes à implementação do simulador são apresentados a seguir.

#### 4.1- CARACTERÍSTICAS DO SIMULADOR

Uma especificação ESTELLE consiste da descrição de um conjunto de módulos que cooperam através de troca de interações depositadas em pontos de interação, que são associados a filas FIFO ('FIST IN FIST OUT'). O simulador de protocolos exercita o conjunto de módulos componentes de uma especificação ESTELLE: durante uma simulação são criadas instâncias de módulos cooperantes que podem ser destruídas ao longo da simulação. A criação de uma instância de módulo corresponde à criação de um conjunto de instâncias de objetos associados a este módulo, tais como as filas FIFO correspondentes aos pontos de interação, as variáveis e os módulos declarados em seu interior.

O simulador deve oferecer ao projetista do protocolo facilidades que lhe permitam uma interação com o ambiente de simulação. Entre as facilidades oferecidas, podem ser destacadas:

- o acesso em tempo de simulação a todas as instâncias de objetos definidos na especificação ESTELLE do protocolo considerado;
- a inserção de pontos de parada;
- a obtenção de traços da simulação;
- a introdução de sequências de teste quando da ocorrência de um ponto de parada.

#### 4.2- ARQUITETURA DO SIMULADOR

A figura 4.1 apresenta a arquitetura geral do simulador, destacando o fluxo de informações entre os diversos módulos que o compõem, além da estrutura da Forma Intermediária que fornece ao simulador as informações referentes a uma dada especificação em ESTELLE.

A Forma Intermediária é composta de :

- uma parte de dados constituindo o conjunto das estruturas que correspondem à topologia do modelo descrito em uma especificação em ESTELLE;
- uma parte de programa constituindo o conjunto de funções e de procedimentos associados ao modelo descrito. Estes procedimentos e funções pertencem a duas categorias :
  - . aqueles que são escritos pelo projetista e que são recuperados pelo compilador diretamente do fonte em ESTELLE;

aqueles que são gerados pelo compilador a partir da análise das cláusulas escritas em ESTELLE, como por exemplo, o conjunto de funções e de procedimentos associados, respectivamente, ao teste de condição de sensibilização e a ação correspondente.

O simulador de protocolos é constituído pelos seguintes módulos destacados na figura 4.1:

- **Rotinas do Usuário:** módulo formado pelos conjuntos de procedimentos extraídos da especificação, obtidos da forma intermediária;
- **Rotinas do Sistema:** módulo formado pelos conjuntos de procedimentos gerados pelo compilador, também obtido da forma intermediária;
- **Topologia:** módulo contido pelo conjunto de objetos estáticos que formam a parte de dados do simulador sendo obtidos diretamente da Forma Intermediária. Estes objetos descrevem a topologia da especificação do protocolo em ESTELLE e não sofrem nenhuma alteração no decorrer da simulação.
- **Contexto da Simulação:** módulo constituído por um conjunto de objetos dinâmicos que são constantemente atualizados durante a simulação. Estas estruturas particulares ao simulador refletem o estado de uma simulação e estão classificadas em três categorias :
  - . conjunto de transições sensibilizadas ;
  - . conjunto de transições disparáveis ;
  - . estruturas que definem as ligações entre módulos.
- **Primitivas ESTELLE:** contém as primitivas que são descritas no ambiente de simulação; durante a execução de uma primitiva as estruturas de dados estáticas que descrevem a topologia da especificação são consultadas e as estruturas de dados dinâmicas da simulação são atualizadas.
- **Motor do Simulador:** é a parte responsável pela execução do modelo de base do ambiente de simulação: o conjunto de máquinas de estados finitas estendidas e coloridas interconectadas por filas. O simulador cria as filas interconectando as diversas máquinas de estados finitas, inclusive aquelas que são conectadas ao ambiente exterior e que constituem uma interface entre o núcleo e um usuário externo. O algoritmo geral do motor do simulador pode ser organizado segundo os seguintes passos :
  - i) obtenção de uma interação do usuário: esta fase é executada pela interface H/M do simulador. Aqui o usuário pode enviar uma sequência de interações para serem interpretadas pelo simulador. O envio destas interações pode ser feito de duas maneiras :
    - . uma a uma, via teclado;
    - . o usuário fornece o nome de um arquivo contendo interações previamente definidas;
  - ii) tratamento da interação: a condição de sensibilização das interações são testadas e é constituído um conjunto contendo as transições

- disparáveis;
- iii) disparo de uma transição: é feita uma seleção de uma entre as transições pertencentes ao conjunto de transições disparáveis; a ação correspondente é executada;
- iv) avanço para o próximo estado;
- v) teste do ponto de parada: se o estado atual foi setado como um ponto de parada retornar ao item i, senão, retornar ao item ii.

#### 4.3- INTERFACE COM O AMBIENTE EXTERNO

O ambiente de simulação esquematizado na figura 4.2 fornece ao usuário meios de interagir com o modelo simulado para verificar propriedades da camada de protocolo sob teste.

O modelo simulado é deduzido da especificação do protocolo em ESTELLE, onde cada corpo de módulo ESTELLE corresponde a uma máquina de estados finita estendida. Estas máquinas são ditas coloridas, uma vez que cada cor corresponde a existência de uma instância de módulo do mesmo tipo em tempo de execução da simulação. As diversas instâncias de módulos trocam interações com o ambiente externo por meio das filas. Na figura 4.2, os pontos de interação I<sub>Pi</sub> do modelo simulado representam os pontos de interação que jamais são conectados ao longo da especificação ESTELLE e que são diretamente acessíveis a partir do mundo exterior; supõem-se que os pontos de interação I<sub>Pi</sub> estão implicitamente conectados aos pontos de interação I<sub>PSi</sub> do módulo de interface.

O módulo de interface é responsável pela interface entre o usuário e o simulador. Através dele o usuário pode enviar dois tipos de interações aos pontos I<sub>Pi</sub>'s do modelo simulado:

- interações temporizadas ;
- interações temporizadas condicionadas pela ocorrência de um dado evento, que corresponde à recepção de uma interação particular proveniente do modelo simulado.

A notação utilizada para a descrição destas interações é baseada naquela que permite descrever uma transição em ESTELLE.

As interações temporizadas são descritas por:

Sintaxe :

```

delay (Tinf,Tsup)
priority (Prty)
begin
    output IPi.INTa
end ;

```

Semântica :

- a interação INTa deve ser enviada, com a prioridade Prty, ao ponto de interação I<sub>Pi</sub> dentro do intervalo de tempo (Tinf, Tsup) após a ocorrência do ponto de parada.

Da mesma forma, as interações temporizadas decorrentes da chegada de um evento são

representadas por :

Sintaxe :

```

when IPj.INTb
priority Prty
delay (Tinf,Tsup)
begin
    output IPi.INTa
end ;

```

Semântica :

- a interação INTa deve ser enviada, com a prioridade Prty, ao ponto de interação I<sub>Pi</sub>, dentro dos limites de tempo (Tinf, Tsup) após a chegada da interação INTb sobre o ponto IPj que está implicitamente conectado ao módulo de interface .

O módulo de interface possui dois conjuntos SI e SD que armazenam as interações temporizadas até que elas sejam enviadas ao modelo simulado :

- o conjunto SI armazena as interações até que as restrições temporais sejam satisfeitas;
- o conjunto SD armazenam as interações decorrentes da chegada de um evento até que este evento seja produzido. Quando o evento ocorre, a interação associada é transferida para o conjunto SI.

#### 4.4 IMPLEMENTAÇÃO DO SIMULADOR

O simulador encontra-se totalmente especificado em uma linguagem apropriada à especificação de algoritimos e dados [14]. Os dados, em particular, são especificados como conjuntos que admitem um certo número de operações sobre seus objetos, segundo o conceito de estrutura de dados abstratos.

Os diversos módulos componentes do simulador estão sendo implementados em MODULA-2 por ser esta uma linguagem particulamente adequada ao desenvolvimento de sistemas de médio porte [15]. Os conjuntos são implementados como listas encadeadas; a cada conjunto está associado um grupo de primitivas (procedimentos e funções) que executam operações sobre os mesmos e que estão encapsuladas no interior de um módulo definido em MODULA-2.

#### 5- CONCLUSÃO

Este trabalho apresentou a descrição de um simulador de protocolos para ESTELLE que constitui o núcleo de um Sistema de Auxílio ao Projeto de Protocolos de Comunicação. Este simulador é o primeiro de uma série de trabalhos que estão sendo desenvolvidos pelo Programa de Engenharia Elétrica e pelo Núcleo de Robótica da COPPE/UFRJ objetivando a criação de um conjunto de ferramentas que auxiliem o projeto de protocolos de comunicação.

Os próximos trabalhos corresponderão ao desenvolvimento de um tradutor/compilador ESTELLE-Forma Intermediária, assistido por um editor orientado

por sintaxe. Seguem-se o desenvolvimento dos demais módulos de auxílio ao projeto que constituirão um "CAD para Protocolos de Comunicação". Ainda dentro do âmbito de ESTELLE serão feitos estudos relativos aos novos avanços propostos para ESTELLE que visem o aumento do poder de expressão e das facilidades de validação para protocolos especificados neste formalismo [11,12].

**6- REFERENCIAS BIBLIOGRAFICAS:**

- [1] ISO/TC 97/SC 21/WG1/FDT/B, "ESTELLE - A Formal Description Technique Based on an Extended State Transition Model", ISO, 1986;
- [2] ISO/TC 97/SC 21/WG1/FDT/C, "LOTOS - A Formal Description Technique based on the Temporal Ordering of Observational Behaviour", ISO, 1986;
- [3] A. Pedroza, "Desenvolvimento de um Ambiente de C.A.D. para o Auxilio a Concepção, ao Teste e a Implementação de Protocolos em Sistemas de Computadores Distribuidos", Projeto CNPq ref: 4031187/85-CC, novembro 1985;
- [4] A. Pedroza, "Contribution a la Definition d'un Environnement pour la Conception de Protocoles", These de Docteur, Univ. Paul Sabatier, Toulouse, 1985;
- [5] R. J. Linn, "The Features and Facilities of ESTELLE", Int. Workshop on Prot. Specif. Verif. and Testing, 5, Moissac, 1985;
- [6] J. P. Courtiat, "A Petri net based semantics for ESTELLE", rapport de recherche LAAS 86113, May 1986 - version revisee et completee dans SEDOS report 109, Octobre 1987;
- [7] J. P. Courtiat, J. M. Ayache, "Design Specification of ESTELLE Intermediate Form", rapport SEDOS 410/LAAS/B1/6, May 1986;
- [8] B. Sarikaya, "Test Design for Computer Network Protocols", Ph.D. thesis, McGill University, Montreal, March 1984;
- [9] B. Sarikaya, "Recent Developments in Protocol Testing", 5o. Simposio Brasileiro de Redes de Computadores, Sao Paulo, Abril 1987;
- [10] F. J. de Souza, "ANAPETRI: analisador de Redes de Petri", Tese de Ms.C., COPPE - UFRJ, Marco 1987;
- [11] J. P. Courtiat, "How could Estelle become a better FDT?", 7h IFIP Symposium on Protocol Specification, Testing and Verification, Zurich, May 1987;
- [12] J. P. Courtiat, "Contribution a la Description Formelle de Protocoles", These de Docteur, Univ. Paul Sabatier, Toulouse, 1987;
- [13] G. V. Bochmann, "Recent Developments in Protocol Specification, Validation and Testing", Publication 557, Departement D'Informatique et de Recherche Operationelle, Universite de Montreal, janeiro 1986;
- [14] J. P. Ayache, J. P. Courtiat, "Intermediate Language Specification", rapport SEDOS 410/LAAS/B1/5, May 1985
- [15] G. Pomberger, "Software Engineering and Modula-2", Prentice Hall, 1984.

Refinamento	process	activity
Módulo		
systemprocess	SIM	SIM
systemactivity	NÃO	SIM
process	SIM	SIM
activity	NÃO	SIM

TABELA 2.1: Critério para Refinamento de Módulos.

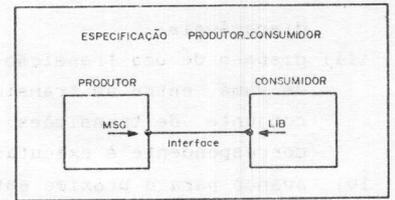


FIGURA 2.1: Arquitetura em ESTELLE para o Exemplo do Produtor-Consumidor.

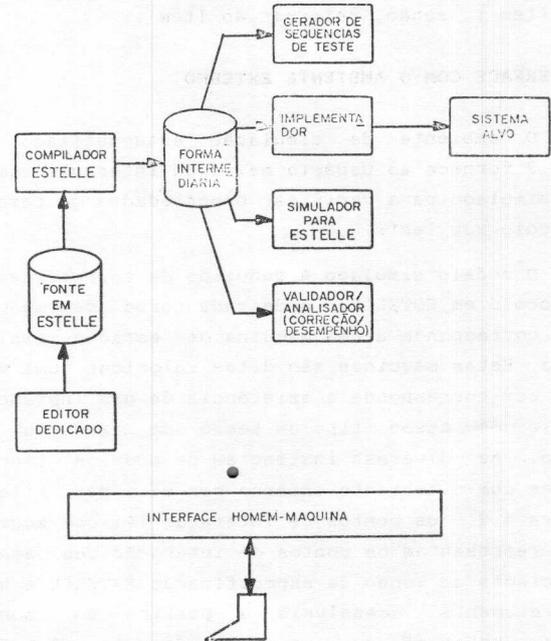


FIGURA 3.1: Arquitetura Geral do "CAD para Protocolos".

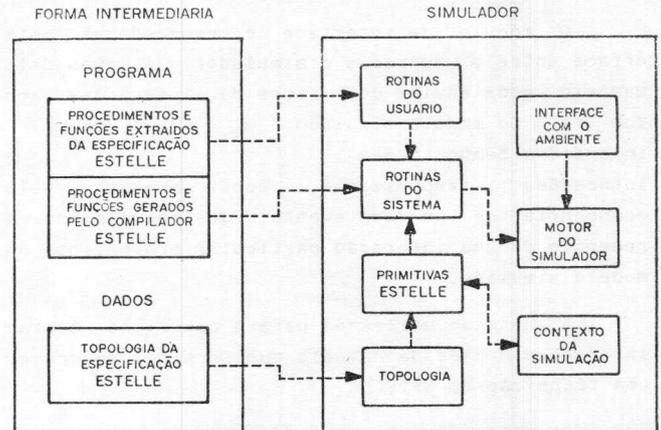


FIGURA 4.1: Arquitetura Geral do Simulador

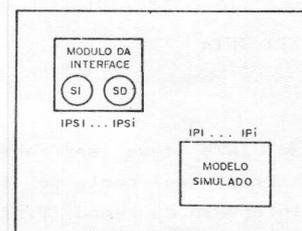


FIGURA 4.2: Interface com o Ambiente Externo