Especificação algébrica de processos da camada de aplicação do RM OSI/ISO (*)

Bernardo Gonçalves Riso (**)
CEC/CTC/Universidade Federal de Santa Catarina (UFSC)
Campus da Trindade
88.040 Florianópolis (SC)

Wanderley Lopes de Souza
GRC/DSC/CCT/Universidade Federal da Paraíba (UFPb)
Av. Aprígio Veloso, 882
58.100 Campina Grande (Pb)

SUMÁRIO

O objetivo principal deste trabalho é o de demonstrar a viabilidade de utilização de álgebras, voltadas para a descrição da dinâmica da comunicação entre processos concorrentes, para a especificação formal dos serviços e protocolos da Camada de Aplicação do RM OSI/ISO. Para tal CCS, desenvolvido por Robin Milner, é empregado como uma Técnica de Descrição Formal para a especificação de um subconjunto do serviço e do protocolo FTAM da ISO. Após uma introdução à camada de aplicação, são definidos um subconjunto do serviço FTAM e o protocolo correspondente. Em seguida, após uma introdução à álgebra CCS, são realizadas as especificações, em CCS, dos subconjuntos definidos. Na conclusão, apresenta-se uma avaliação da metodologia empregada.

(*) realizado com o auxílio do CNPq e da CAPES. Faz parte de um programa de trabalho entre o GRC/UFPb e o Centro Científico Rio da IBM Brasil.

(**) doutorando junto ao CPgEE da UFPb.

1. Introdução

Na arquitetura do Basic Reference Model for Open Systems Interconnection (RM OSI) [ISO 7498], a camada de aplicação é composta de duas subcamadas: a inferior, constituída dos Common Application Service Elements (CASE); e a superior, composta dos Specific Application Service Elements (SASE). Entre os elementos do SASE está o File Transfer Access and Management (FTAM) [ISO 8571]. Uma apresentação detalhada da estrutura da camada de aplicação pode ser encontrada em [MoSa 86].

Algumas propriedades típicas da maioria dos protocolos da camada de aplicação são apreciadas em [BoDe 86]. A primeira delas diz respeito à correspondência entre as primitivas de serviço e as unidades de dados de protocolo (UDPs, que são compostas de informações de controle e, possivelmente, dados do usuário). Há uma correspondência direta entre a recepção (ou o envio) de uma UDP e a execução de uma primitiva. Essa correspondência é uma-a-uma, e inclui a correspondência entre os parâmetros das UDPs e os parâmetros das primitivas de serviço.

A segunda propriedade refere-se à independência que, em geral, ocorre entre as regras que determinam a ordem na qual as UDPs são executadas e as regras para a escolha dos valores apropriados dos parâmetros dessas UDPs. Uma exceção é o parâmetro "resultado", de certas UDPs (por exemplo, daquela correspondente à primitiva F_INITIALIZE response do FTAM), que indica se a solicitação de um serviço (no exemplo, o estabelecimento de uma associação FTAM) foi atendida, ou não. Para essas exceções, entretanto, a característica de independência pode ser recuperada substituindo-se, na definição do protocolo, a UDP que contém um parâmetro "resultado", por duas UDPs distintas, sem tal parâmetro. Assim, uma UDP indicaria o atendimento (F_INITIALIZE response positivo) e a outra, a recusa (F_INITIALIZE reponse negativo) do serviço solicitado. Nas especificações do presente trabalho, este expediente é utilizado.

A reunião das propriedades mencionadas acima resulta na pro-

priedade de independência entre a ordem de execução das primitivas de serviço e os parâmetros dessas primitivas. Considerando essa propriedade dos protocolos de aplicação, vê-se como pode ser adequado o tratamento dos aspectos de controle, separadamente dos aspectos de dados, na tarefa de especificação e validação desses protocolos. Desse modo, Calculus of Communicating Systems (CCS) [Miln 80] [Miln 89], uma Técnica de Descrição Formal (TDF) voltada para a definição dos aspectos dinâmicos da comunicação entre processos, pode ser utilizada para a especificação do sequenciamento das primitivas de serviço e das UDPs, sem preocupação imediata quanto à formalização dos tipos de dados. Estes, possivelmente, seriam especificados de modo formal, em separado, através de extensões adequadas à álgebra CCS.

2. Serviço e protocolo FTAM

O modelo abstrato do serviço FTAM pode ser definido através das interações de dois usuários com o provedor do serviço. Um dos usuários é o iniciador da comunicação e o outro, o respondedor. Este último manipula um depósito virtual de arquivos.

Para oferecer o serviço FTAM, duas entidades de protocolo (a iniciadora e a respondedora) comunicam-se utilizando os serviços oferecidos pelo CASE e pela camada de apresentação.

2.1. Definição de um subconjunto para o FTAM

Neste trabalho, para obter simplicidade, somente são especificados os serviços e os protocolos para o estabelecimento e a terminação ordenada dos regimes aninhados de associação FTAM e de seleção de arquivo (figura 1). Os serviços considerados são do tipo confirmado e fazem parte da unidade funcional Núcleo. Para oferecê-los, as entidades de protocolo precisam executar apenas um subconjunto do protocolo básico definido na part: 4 de [ISO 8571].

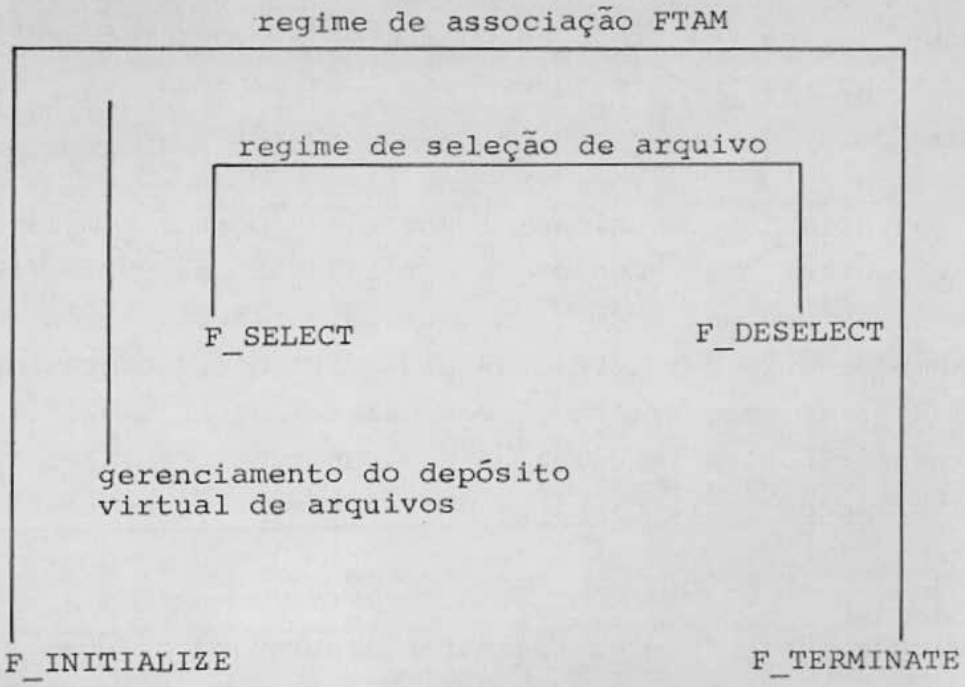

Fig. 1 - Regimes de um subconjunto FTAM e primitivas relacionadas.

3. CCS: uma TDF para a especificação de serviços e protocolos

CCS é uma álgebra voltada para a descrição do comportamento de processos, do modo como eles são percebidos por um observador externo. Assim, um processo é visto como uma verdadeira caixa preta, o que dá, à sua descrição, um caráter completamente abstrato. Nessa álgebra, os processos são representados por agentes que podem se comunicar, no modo síncrono, com ou sem passagem de valores. A seguir são apresentados alguns elementos de CCS. Um resumo dessa álgebra pode ser encontrado em [LoRi 88].

3.1. Sintaxe de CCS

Inicialmente, é preciso dizer que, neste trabalho, introduziram-se ligeiras modificações na notação original de CCS. As modificações e as justificativas correspondentes, são: (1) ao invés da barra superior ($\bar{}$), usam-se os símbolos ! e ? para distinguir as portas oferecedoras das portas receptoras de eventos, facilitando a edição das especificações; e (2) usa-se uma espécie de

"guarding", para evitar o aninhamento de construções "if-then-else". Ambas as modificações foram inspiradas em LOTOS [ISO 8807].

Os agentes CCS comunicam-se através de portas complementares, por exemplo: a? e a!, sendo que o rótulo de uma porta receptora pode estar vinculado a um conjunto de variáveis $\tilde{x} = \{x_1,...,x_n\}$, por exemplo: a?$\tilde{x}$, e o rótulo de uma porta oferecedora pode estar vinculado a um conjunto de expressões de valor $\tilde{E} = \{E_1,...,E_n\}$, por exemplo: b!$\tilde{E}$. A comunicação com passagem de valores exige compatibilidade entre os tipos das variáveis e os valores passados.

As operações dinâmicas permitem descrever os comportamentos dos agentes: ausência de ação (NIL ou 0); escolha não-determinística (+); ações de comunicação com o ambiente (a?$\tilde{x}$, b!$\tilde{E}$); e intercomunicação dos componentes de um sistema (i). As operações estáticas são usadas para fixar uma estrutura de ligação entre agentes: composição concorrente (|); ocultamento de portas (\a); e re-rotulação de portas (T = ab/cd). Outros meios de expressão incluem a recursividade e a utilização de identificadores parametrizáveis para as expressões de comportamento (C[$\tilde{E}$]).

3.2. Semântica de CCS

A semântica de CCS é definida através de um conjunto de regras de inferência a partir de ações atômicas. $C \xrightarrow{pv} C'$ significa que o comportamento de um agente, expresso por C, passa a ser definido por C' após a ocorrência de um evento na porta p desse agente, com passagem de valor v.

3.3. Equivalência de observação ($\simeq$)

Dois agentes são equivalentes quanto à observação se forem indistinguíveis para um observador que realize experimentos com esses agentes. Formalmente, a definição de $\simeq$ é realizada, em [Miln 80], através de uma sequência decrescente de graus de equivalência e, em [Miln 89], através do conceito de bi-simulação.

3.4. Teorema da expansão (TE)

O TE permite o desdobramento de qualquer expressão de comportamento em um conjunto de ações e escolhas não-determinísticas. Aliando-se a operação de ocultamento de portas ao TE e às propriedades da $\approx$, obtêm-se recursos poderosos para a análise de especificações.

3.5. Validação das especificações CCS

Essa validação é realizada através da prova de $\approx$ entre as especificações de um mesmo sistema, em diferentes níveis de abstração. Seguindo-se um método intuitivo para a realização de uma tal prova, inicialmente compõem-se os agentes de uma especificação refinada e ocultam-se as portas de comunicação entre eles. Após isso, aplica-se repetidamente o TE (uma vez para cada evento), seguindo-se as diversas possibilidades de sequenciamento de ações (traços). Usando-se, então, as propriedades da $\approx$, pode-se suprimir as representações de eventos internos para destacar o comportamento observável da composição. A expressão obtida após a manipulação descrita acima deve representar o mesmo comportamento que a especificação abstrata do sistema, para que a especificação composta seja considerada um refinamento correto da especificação abstrata.

Mesmo para sistemas simples, essa prova pode se tornar longa porque é constituída, principalmente, de aplicações repetidas do TE (vide [LoRi 88]). Para sistemas complexos, como aqueles que pertencem ao modelo RM OSI, a tarefa de validação deve se valer de recursos automatizados.

4. Especificação CCS de um subconjunto FTAM

Nesta seção são apresentadas as especificações, em CCS, dos serviços do núcleo FTAM (excluindo a terminação abrupta do regime de associação FTAM) e da parte correspondente do protocolo básico. Os nomes das primitivas de serviço e das UDPs são abreviados de acordo com [ISO 8571]. As primitivas começam com F_.

4.1. Especificação dos serviços

De acordo com a figura 2, o provedor dos serviços do núcleo FTAM, P_Serv, é considerado como uma caixa preta e o seu comportamento observável (o serviço que oferece) é descrito através dos eventos (primitivas de serviço) que podem ocorrer nas portas de comunicação com os seus usuários.

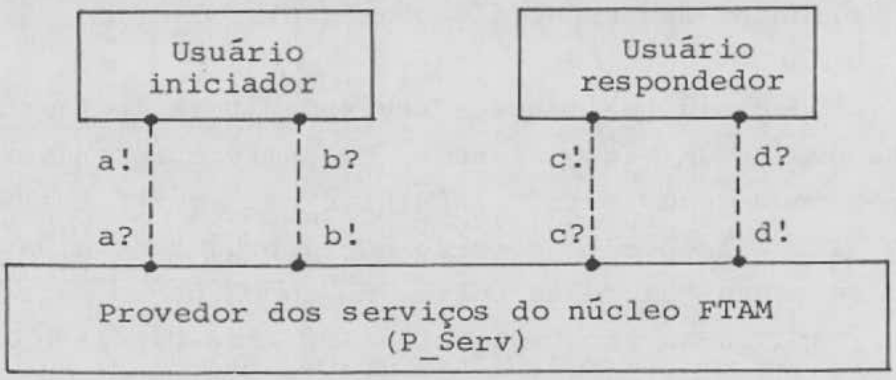

Fig. 2 - Modelo para o serviço do núcleo FTAM.

Inicialmente, o provedor dos serviços do núcleo FTAM comporta-se como

$$P\_Serv \stackrel{def}{=} a?\tilde{x}.([\tilde{x} = F\_INIRQ] \rightarrow P(\tilde{x}))$$

de modo que pode receber, na porta a?, uma primitiva de serviço e seus parâmetros (atribuindo-os à tupla de variáveis $\tilde{x}$), provenientes do usuário inciador. Após esse evento, o provedor verifica se se trata de uma F_INITIALIZE request (única primitiva permitida nesse contexto), caso em que passa a comportar-se como

$$P(\tilde{x}) \stackrel{def}{=} ([Pl(\tilde{x}) = true] \rightarrow Q(\tilde{x}) + [Pl(\tilde{x}) = false] \rightarrow b!F\_INICFneg.P\_Serv)$$

para avaliar, localmente, a solicitação de estabelecimento do regime FTAM. Essa avaliação é realizada através do predicado Pl (definido em [ISO 8571]). No caso de atendimento, o provedor passa a comportar-se como $Q(\tilde{x})$; caso contrário, pode emitir, na

porta b!, uma primitiva F_INITIALIZE confirm negativo para o usuário iniciador, após o que volta a comportar-se como P_Serv.

$$Q(\tilde{x}) \stackrel{def}{=} ([P5(\tilde{x}) = true] \rightarrow d!F\_INIIN.In + [P5(\tilde{x}) = false] \rightarrow b!F\_INICFneg.P\_Serv)$$

Como $Q(\tilde{x})$, o provedor decide sobre a aceitação remota avaliando o predicado P5 (idem). No caso favorável, oferece, na porta d!, uma primitiva F_INITIALIZE indication ao respondedor e, após a ocorrência desse evento, passa a comportar-se como In; no caso contrário, oferece, na porta b!, uma primitiva F_INITIALIZE confirm negativo ao iniciador e volta a comportar-se como P_Serv.

$$In \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_INIRPpos] \rightarrow b!F\_INICpos.Inic + [\tilde{y} = F\_INIRPneg] \rightarrow b!F\_INICneg.P\_Serv)$$

de modo que a recepção de uma F_INITIALIZE response positivo permite estabelecer um regime de associação FTAM (o provedor, então, passa a comportar-se como Inic). Contrariamente, a recepção de uma F_INITIALIZE response negativo impede, momentaneamente, o estabelecimento desse regime (o provedor, então, volta a comportar-se como P_Serv, onde ficará disponível para outra tentativa do usuário iniciador).

$$Inic \stackrel{def}{=} a?\tilde{x}.([\tilde{x} = F\_TERRQ] \rightarrow d!F\_TERIN.T + [\tilde{x} = F\_SELRQ] \rightarrow d!F\_SELIN.S)$$

assim, a recepção de uma F_TERMINATE request inicia a fase de terminação do regime de associação FTAM, e leva o provedor a comportar-se como T; na outra escolha, a recepção de uma F_SELECT request permite tratar um pedido de estabelecimento do regime de seleção de arquivo, levando o provedor a comportar-se como S.

$$T \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_TERRP] \rightarrow b!F\_TERCF.P\_Serv)$$

termina ordenadamente o regime de associação FTAM.

$$S \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_SELRPpos] \rightarrow b!F\_SELCFpos.Sel + [\tilde{y} = F\_SELRPneg] \rightarrow b!F\_SELCFneg.Inic)$$

$$Sel \stackrel{def}{=} a?\tilde{x}.([\tilde{x} = F\_DESRQ] \rightarrow d!F\_DESIN.D)$$

$$D \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_DESRP] \rightarrow b!F\_DESCF.Inic)$$

permite estabelecer e encerrar o regime de seleção de arquivo.

4.2. Especificação do protocolo

O protocolo, visto como uma implementação lógica do serviço correspondente, pode ser obtido através de um conjunto de regras de transformação como em [BoGo 86]. Em outra abordagem, o protocolo é obtido como o refinamento de um agente CCS que representa o serviço [Riso 88]. Neste trabalho, adota-se a segunda abordagem, e o protocolo básico FTAM, correspondente aos serviços descritos na subseção 4.2, é obtido pelo rompimento da caixa preta P_Serv. Isso permite ver três novas caixas pretas, correspondentes às entidades de protocolo FTAM iniciadora e respondedora, e ao provedor dos serviços subjacentes (figura 3). Os comportamentos das entidades de protocolo são, a seguir, descritos em CCS.

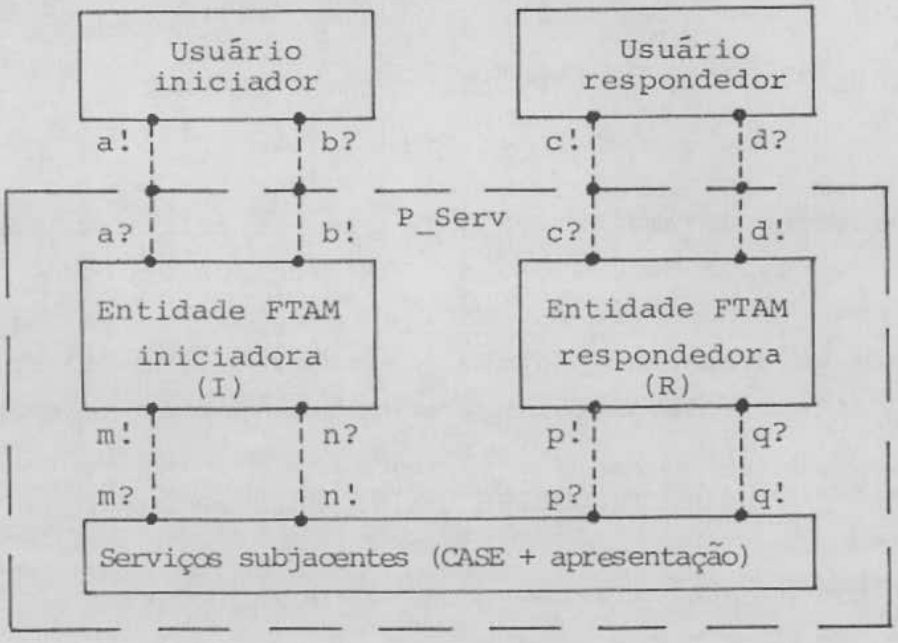

Fig. 3 - Modelo para o protocolo FTAM básico.

4.2.1. Especificação da entidade iniciadora

Essa entidade comporta-se como

$$I \stackrel{def}{=} a?\tilde{x}.([\tilde{x} = F\_INIRQ] \rightarrow I1(\tilde{x}))$$

$$I1(\tilde{x}) \stackrel{def}{=} ([P1(\tilde{x}) = true] \rightarrow m!INIRQ.I\_IP + [P1(\tilde{x}) = false] \rightarrow b!F\_INICFneg.I)$$

$$I\_IP \stackrel{def}{=} n?\tilde{r}.([\tilde{r} = INIRPpos] \rightarrow b!F\_INICFpos.I\_In + [\tilde{r} = INIRPneg] \rightarrow b!F\_INICFneg.I)$$

para tratar um pedido de estabelecimento do regime de associação FTAM, e como

$$I\_In \stackrel{def}{=} a?\tilde{x}.([\tilde{x} = F\_TERRQ] \rightarrow m!TERRQ.I\_TP + [\tilde{x} = F\_SELRQ] \rightarrow m!SELRQ.I\_SP)$$

$$I\_TP \stackrel{def}{=} n?\tilde{r}.([\tilde{r} = TERRP] \rightarrow b!F\_TERCF.I)$$

$$I\_SP \stackrel{def}{=} n?\tilde{r}.([\tilde{r} = SELRPpos] \rightarrow b!F\_SELCFpos.I\_S + [\tilde{r} = SELRPneg] \rightarrow b!F\_SELCFneg.I\_In)$$

para tratar um pedido de terminação ordenada do regime de associação FTAM ou um pedido de estabelecimento do regime de seleção de arquivo.

Comportando-se como

$$I\_S \stackrel{def}{=} a?\tilde{x}.([\tilde{x} = F\_DESRQ] \rightarrow m!DESRQ.I\_D)$$

$$I\_D \stackrel{def}{=} n?\tilde{r}.([\tilde{r} = DESRP] \rightarrow b!F\_DESCF.I\_In)$$

permite encerrar o regime de seleção de arquivo.

4.2.2. Especificação da entidade respondedora

Essa entidade comporta-se como

$$R \stackrel{def}{=} q?\tilde{s}.\ ([\tilde{s} = INIRQ] \rightarrow R1(\tilde{s}))$$

$$R1(\tilde{s}) \stackrel{def}{=} ([P5(\tilde{s}) = true] \rightarrow d!F\_INIIN.R\_FIP + [P5(\tilde{s}) = false] \rightarrow p!INIRPneg.R)$$

$$R\_FIP \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_INIRPpos] \rightarrow p!INIRPpos.R\_In + [\tilde{y} = F\_INIRPneg] \rightarrow p!INIRPneg.R)$$

para tratar uma solicitação de estabelecimento do regime de associação FTAM, e como

$$R\_In \stackrel{def}{=} q?\tilde{s}.([\tilde{s} = SELRQ] \rightarrow d!F\_SELIN.R\_FSP + [\tilde{s} = TERRQ] \rightarrow d!F\_TERIN.R\_FTP)$$

$$R\_FSP \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_SELRPpos] \rightarrow p!SELRPpos.R\_S + [\tilde{y} = F\_SELRPneg] \rightarrow p!SELRPneg.R\_In)$$

$$R\_FTP \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_TERRP] \rightarrow p!TERRP.R)$$

para tratar um pedido de terminação ordenada do regime de associação FTAM ou um pedido de estabelecimento do regime de seleção de arquivo.

Comportando-se como

$$R\_S \stackrel{def}{=} q?\tilde{s}.([\tilde{s} = DESRQ] \rightarrow d!F\_DESIN.R\_D)$$

$$R\_D \stackrel{def}{=} c?\tilde{y}.([\tilde{y} = F\_DESRP] \rightarrow p!DESRP.R\_In)$$

permite encerrar o regime de seleção de arquivo.

5. Conclusões

CCS vem despertando um interesse crescente por parte da comunidade internacional de informática, e tem sido objeto tanto de pesquisas teóricas como de aplicações. No campo prático foi, por exemplo (conforme relatado em [Miln 89], pág. 249), usado para a verificação de um protocolo Carrier Sense Multiple Access

with Collision Detection (CSMA/CD), correspondente à camada física do RM OSI, cabendo, portanto, estender a investigação sobre a sua aplicabilidade a todas as camadas do modelo.

Neste trabalho, discute-se a aplicabilidade de CCS no caso da especificação dos serivços e dos protocolos da camada de aplicação. Para essa camada, os aspectos de dados podem assumir alta complexidade, entretanto, como foi mostrado em [BoDe 86], seus protocolos permitem a abordagem dos aspectos de controle independentemente dos dados. Desse modo, CCS, uma TDF com poucos operadores, e voltada para a descrição da dinâmica da comunicação entre sistemas, é suficiente para a especificação dos aspectos de controle da camada de aplicação do RM OSI.

A validação das especificações complexas, realizadas em CCS, exige ferramental automático. A presente especificação de um subconjunto FTAM, embora relativamente simples, já apresenta complexidade suficiente para impedir que a sua validação manual seja confiável. A máxima complexidade, que ainda permite uma validação manual confiável, está ao nível das especificações do protocolo do bit alternante ([Miln 89], pág. 147).

A metodologia empregada na elaboração das especificações apresentadas permite a intercalação de passos de análise aos passos de desenvolvimento, levando à produção de especificações corretas por construção. Essa metodologia pode ser descrita formalmente, e é nesse sentido que os autores estão, atualmente, realizando as pesquisas que se relacionam mais de perto com o presente trabalho.

Referências

[BoDe 86] Bochmann, G. v.; Deslauriers, M.; Bessette, S., "Application Layer Testing and ASN1 Suport Tools", publication 572, Université de Montréal, Canadá, maio 1986, 14 pgs.

[BoGo 86] Bochmann, G.v.; Gotzhein, R., "Deriving Protocol Specifications from Service Specifications", publication 562, Université de Montréal, Canadá, 1986, 16 pgs.

[ISO 7498] "Information Processing Systems - Open Systems Interconnection - Basic Reference Model", IS, 1983.

[ISO 8571] "Information Processing Systems - Open Systems Interconnection - File Transfer, Access and Management", DIS, 1986.

[ISO 8807] "Information Processing Systems - Open Systems Interconnection - LOTOS - Formal Description Technique Based on the Temporal Ordering of Observational Behaviour", 1987, 126 pgs.

[LoRi 88] Lopes de Souza, W.; Riso, B.G.; Monteiro Filho, A.F., "Especificação e verificação em CCS de Sistemas de Comunicação", anais do 6º SBRC, Belo Horizonte (MG), 1988.

[Miln 80] Milner, R., "A Calculus of Communicating Systems", ed. G. Goos e J. Hartmanis, Lecture Notes in Computer Science, Vol. 92, Springer-Verlag, 1980, 181 pgs.

[Miln 89] Milner, R., "Communication and Concurrency", Prentice Hall International (UK) Ltd., 1989, 260 pgs.

[MoSa 86] Moura, J.A.B.; Sauvé, J.P.; Giozza, W.F.; Marinho de Araújo, J.F., "Redes Locais de Computadores - Protocolos de Alto Nível e Avaliação de Desempenho", McGraw-Hill, Ltda., 1986, 446 pgs.

[Riso 88] Riso, B.G., "Uma Metodologia para a Especificação e a Verificação de Protocolos, Através de Refinamentos Sucessivos", proposta de pesquisa, UFPb, Campina Grande (Pb), 1988, 21 pgs.