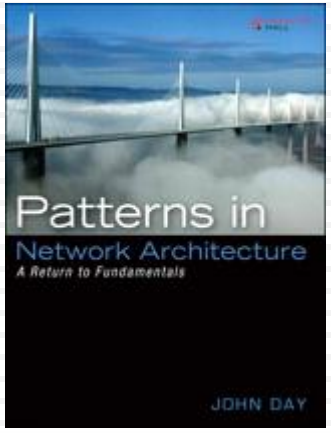
Elementos dos Protocolos

Capítulo 2

Patterns in Network Architecture



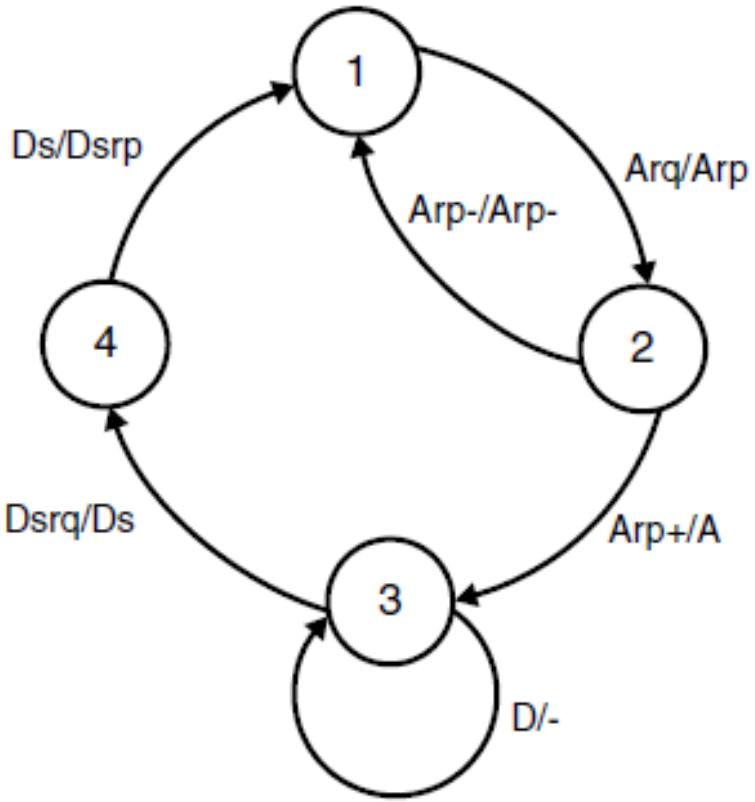
Elementos de um protocolo

- *Toda a comunicação de dados é um efeito colateral.*
- A teoria das máquinas de estado finitas (FSMs) tem sido tradicionalmente usada para descrever e analisar protocolos.
- Ou Máquina de Protocolo (PM):
  - Criada a partir de módulos elementares menores
  - Um pequeno número de conceitos podem ser usados repetidamente para construir uma arquitetura de rede

Definição de uma FSM

- Um alfabeto de entrada:
  - Conjunto $A = \{A_1, \ldots, A_m\}$
- Um conjunto de estados: $S = \{S_1, \ldots, S_n\}$
- Um alfabeto de saída:
  - Conjunto $O = \{O_1, \ldots, O_p\}$
- Duas funções:
  - $F_1(A, S) \rightarrow S$
  - $F_2(A, S) \rightarrow O$

Representação de uma FSM



- Grafo:
  - Nós representam os estados
  - Arcos representam a função F_1
  - Os arcos são rotulados com a entrada e a saída da função F_2
  - Podem ser representadas também através de uma tabela

Aplicabilidade das FSMs

- Usadas como um método de especificação formal
- Na sua forma pura não é adequada para representar a não ser mecanismos simples
- Para algoritmos complexos que envolvam contagens simples, ordenação, etc., o espaço de estados seria aproximadamente o produto das magnitudes de cada parâmetro:
  - Problema da explosão de estados!

Extensões das FSMs

- Para resolver o problema da explosão de estados, a FSM é combinada com linguagens de programação ou linguagens formais.
- FSM é modificada para consistir de:
  - Um alfabeto de entrada e um de saída
  - Conjunto de procedimentos
  - Um vetor de estados que inclui os principais estados (início, espera por algo, meio, fim) e quaisquer variáveis associadas com estado como: números de sequência, contadores, etc.

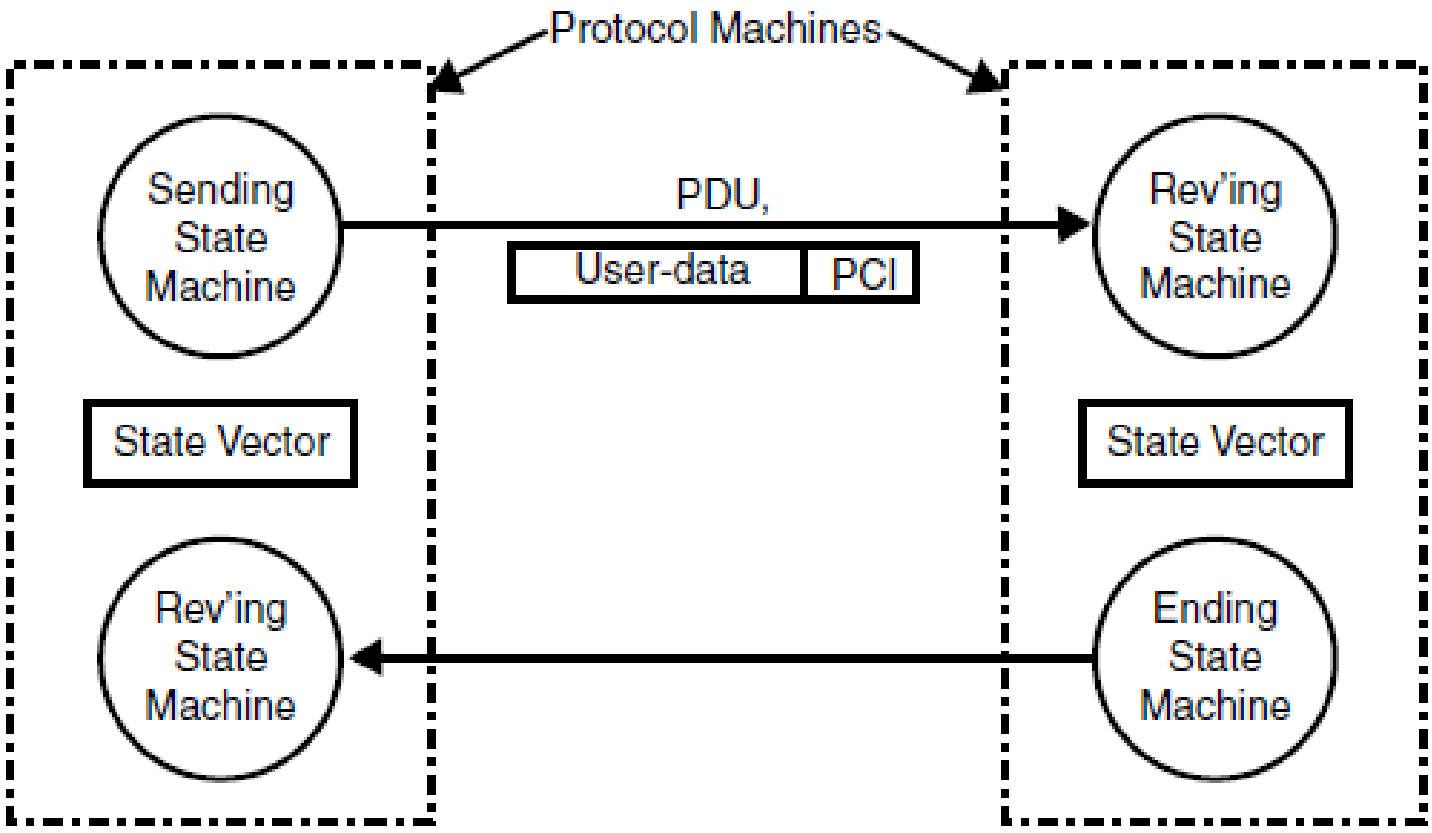
Protocolo

- Para que dois sistemas possam se comunicar, devem possuir um esquema conceitual compartilhado.
- O conjunto de regras e procedimentos que cada sistema participante da comunicação deve seguir para manter a coordenação de seu esquema compartilhado é chamado de protocolo.
- As FSMs que implementam o protocolo serão referenciadas como máquinas de estado de protocolo ou, simplesmente, máquinas de protocolos (PMs).

Protocolos

- São usados em comunicação de computadores em duas classes abrangentes de problemas:
  - Coordenação a distância e
  - Ação a distância.
- A especificação do protocolo se torna a especificação das FSMs que se comunicam.

Máquina de Protocolos típica



PMs

- Uma PM modela uma instância de comunicação, um único fluxo.
- Frequentemente o serviço de suporte assim como o usuário de uma PM são também PMs.
  - A (N+1)-PM usa a (N)-PM
  - A (N-1)-PM é usada pela (N)-PM
- Tipo de Máquina de Protocolo (PMT): todas as PMs de um protocolo particular num dado sistema.
- Em geral, um sistema terá mais do que uma PMT para cada protocolo

Protocolos

- Simétricos:
  - Comportamento das PMs comunicantes é o mesmo
- Assimétricos:
  - PMs comunicantes têm comportamentos distintos
  - Subcasos:
    - Usuário/servidor
    - Cliente/servidor
    - Mestre/escravo

Associações, Conexões, Fluxos e Ligações

- Dado que os sistemas não compartilham estado (i.e., memória), uma PM deve ser capaz de notificar a outra de mudanças importantes no estado.
- Isto é realizado através da troca de informações finitas.
- Esta troca contínua de informações entre as PMs cria um "campo" fraco ou ligação entre as PMs.
- Tipos de ligações:
  - Mínima: não necessita de troca de atualizações
  - Fraca: alguma dependência mas não é afetada caso algumas atualizações sejam perdidas
  - Forte: requer que as atualizações sejam recebidas para evitar comportamentos patológicos.

Associações, Conexões, Fluxos e Ligações

- Uma associação representa um estado de compartilhamento e acoplamento mínimos, frequentemente associado com comunicação sem conexões.
- Um fluxo possui mais estado compartilhado mas não fortemente acoplado (sem realimentação).
- Uma conexão possui um estado compartilhado fortemente acoplado (com realimentação), como nos protocolos de transporte fim-a-fim.
- Uma ligação possui um estado compartilhado mais fortemente acoplado, caracterizado geralmente por memória compartilhada.

Interfaces

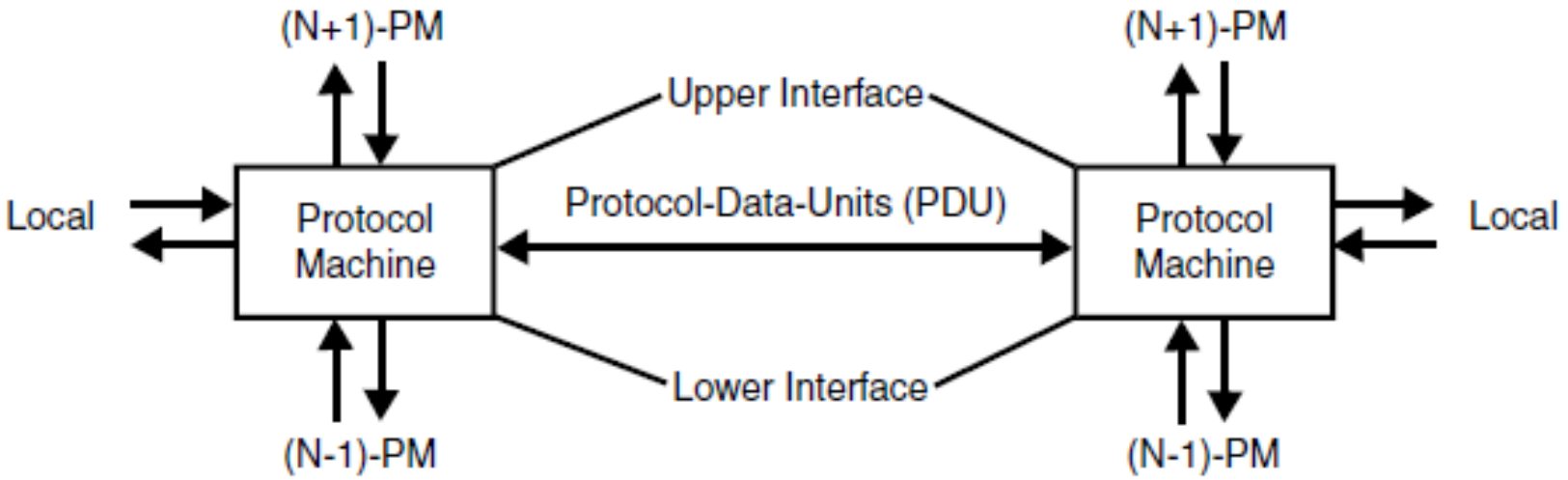
- Um protocolo não existe por si só.
- O usuário é uma FSM acima e que deve se comunicar com a PM para coordenar os seus comportamentos.
- A interface seria a fronteira do usuário com a PM.

- Em telecomunicações “interface” é um protocolo entre tipos de sistemas onde um deles é de propriedade da rede.

Interfaces

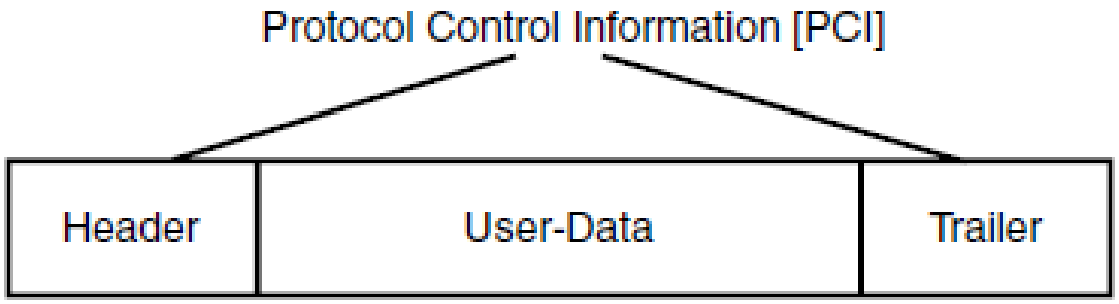
- Troca de informações em um mesmo sistema.
- Maior acoplamento do que em uma conexão!
- Esta troca entre FSMs é frequentemente referenciado como sendo uma Interface de Programação de Aplicação (API).

Relação de uma (N)-PM com outras PMs



Unidades de Dados

- Para que haja a comunicação é necessária uma troca de informações.
- Quadros (*frames*), células (*cell*), pacote, segmento, mensagem, etc.
- Termos diferentes para o mesmo conceito. Chamaremos simplesmente de PDU (*Protocol Data Unit*)



Unidades de dados x Instruções de Processadores

- **Semelhança:** As PDUs executam operações no estado do processador (PM) baseado nos parâmetros da instrução (PCI) e no estado do processador (PM).
- **Diferença:** ao contrário das instruções que carregam um ponteiro (endereço) para os dados sobre os quais opera, as PDUs devem carregar os seus próprios dados.

Tamanho das PDUs

- As PDUs podem ou não conter dados do usuário.
- Não há limites arquiteturais ao tamanho das PDUs.
- Considerações de Engenharia que impõem limites:
  - Em protocolos que operam em ambientes "ruidosos", uma PDU pequena aumenta a probabilidade de que seja recebida sem erros (menor *overhead* de retransmissões).
  - Em uma rede de sensores de tempo-real, os sistemas podem ter espaço muito limitado de *buffer*, de modo que pode ser necessário usar PDUs menores.

Cabeçalhos

- A maior parte das PCIs são contidas no cabeçalho.
- Muitos campos dos protocolos de transferência de dados têm comprimento fixo para facilitar o processamento.
- Normalmente os campos de comprimento fixo antecedem os campos de comprimento variável.
- É fortemente recomendado o uso de um campo que forneça o comprimento total da PDU.

Conteúdo dos Cabeçalhos

- Todo cabeçalho deve ter:
  - Identificador do protocolo
  - Versão do protocolo
  - Campo que indique a função da PDU
  - Campo que codifique a ação associada à PDU
    - Codificação horizontal (bits de controle, como no TCP) ou vertical (*opcode*)
    - Geralmente, *opcodes* são recomendados por questões de eficiência.

Caudas

- As PDUs de alguns protocolos têm uma cauda.
- O mais comum é usá-la para transportar o CRC (*Cyclic Redundancy Code*).
- Geralmente usado em protocolos que operam próximo ao meio físico, pois estando a PDU na memória não há muitas vantagens em ter uma cauda.
- Orientações para o uso de cauda:
  - A informação contida na cauda não é conhecida no instante em que o cabeçalho é criado; é uma função do cabeçalho e dos dados do usuário;
  - O tempo de processamento da PDU é muito menor do que o tempo necessário para a transmissão ou recepção da PDU.

A Natureza da Fronteira de Serviço

- SDU *(Service Data Unit)*: unidade de dados entregue à PM pela (N+1)-PM através da fronteira de serviço.
- Deixamos o termo *interface* para o caso específico de uma dada implementação (ex: interface Unix ou Windows).
- A SDU contém apenas dados do usuário.
- A primitiva de serviço invocada para passar a SDU para a PM passará também outros parâmetros para manipulação da SDU (*ex.: port-id)*

Segmentação da SDU

- A PM pode ter que segmentar a SDU em diversas PDUs, ou agregar diversas SDUs numa única PDU, dado que o tamanho da PDU é um tamanho conveniente para os requisitos do protocolo (N).

- Esta nomenclatura é baseada no Modelo de Referência OSI, não por apoiá-lo, mas porque é uma nomenclatura existente que define termos comuns. Não havendo necessidade de reinventar a roda!

Stream x Registro

- Dado que a SDU pode ter sido fragmentada ou concatenada, o que a (N)-PM deve entregar à (N+1)-PM remota? Aquilo que foi enviado ou aquilo que foi recebido?
- Modos *stream* e registro:
  - Discussão derivada de práticas de sistemas operacionais.
  - Os *mainframes* antigos operavam com registros de comprimento fixo.
  - Sistemas mais modernos (Multics, UNIX) se comunicavam com uma cadeia de bytes.
  - Modo *stream* foi considerada uma abordagem mais flexível e elegante, garantindo uma maior independência entre as camadas.
  - No modo *stream* o protocolo (N+1) deve ter um mecanismo de delimitação pois não pode se basear na camada abaixo para lhe dizer onde se encontra o início e o fim de uma PDU.

Generalização do Modo Registro

- Terceira abordagem.
- As SDUs não têm comprimento fixo.
- A *identidade* das SDUs são mantidas entre o usuário transmissor e o receptor.
- Como nunca foi batizado, vamos chamá-lo de *modo idempotente* (em relação à manutenção da identidade da SDU).
- A camada (N) entrega SDUs na forma que as recebeu. Ou seja, se ele teve que fragmentar a SDU, é sua responsabilidade, remontá-la antes de entregar à (N+1)-PM.
- Boa prática de Engenharia de Software.

Segmentação e Remontagem

- Para o sistema não importa muito em que camada é feito: o trabalho é praticamente o mesmo.
- Um *stream* é simplesmente uma SDU muito longa!

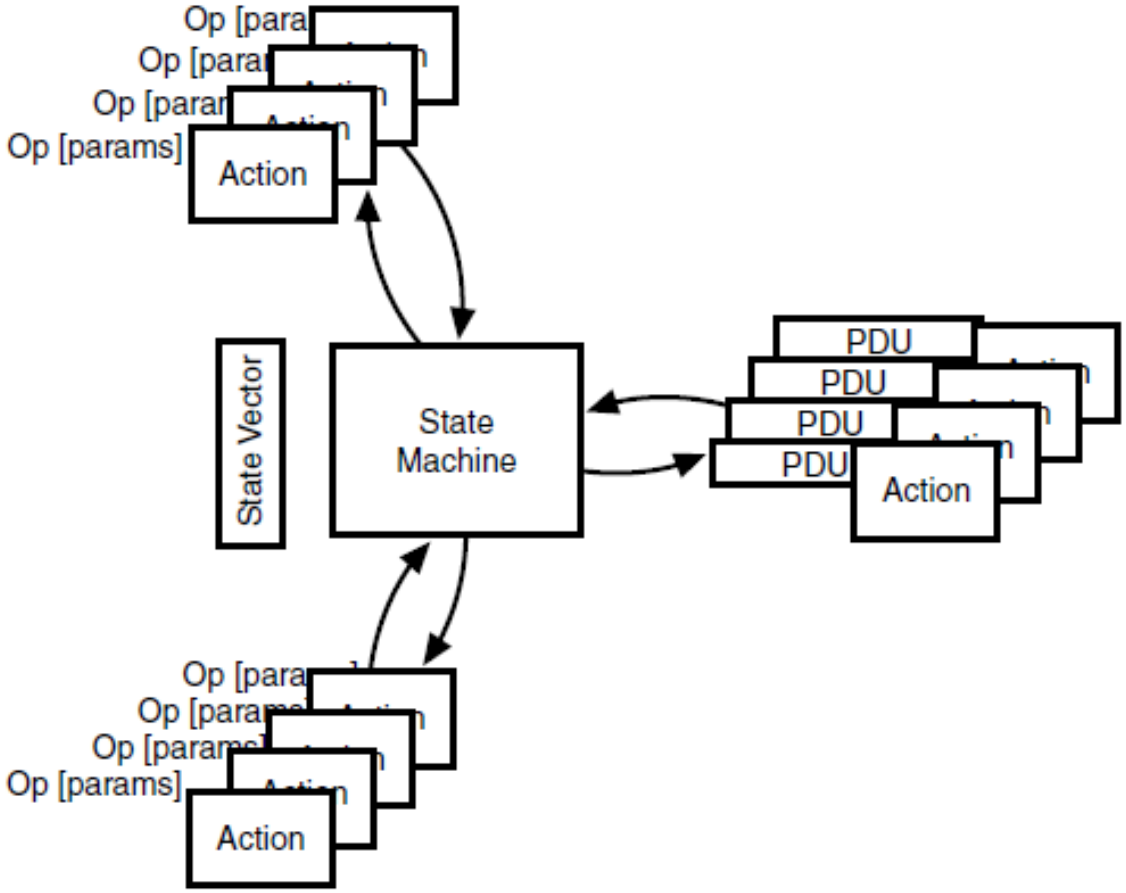
Supporting Both Stream and Idempotent

Delivery of Incomplete SDU Allowed	More Data	Description
0	0	Self-contained PDU, equivalent to Don't Fragment
0	1	Idempotent
1	0	Stream (with huge buffers!)
1	1	Stream

Construindo um Protocolo

- Uma PM deve interpretar quatro entradas:
  - Interações com a interface superior
  - PDUs da(s) PMs parceiras
  - Interações com o sistema local
  - Interações com a interface inferior
- Todas estas podem ser consideradas equivalentes a procedimentos ou chamadas de sistema da seguinte forma:
  - <nome do procedimento>(<param 1>,<param i>*)
- Analogamente:
  - <tipo da PDU>(<elemento da PCI>,<elemento da PCI>*, dados-do-usuário)

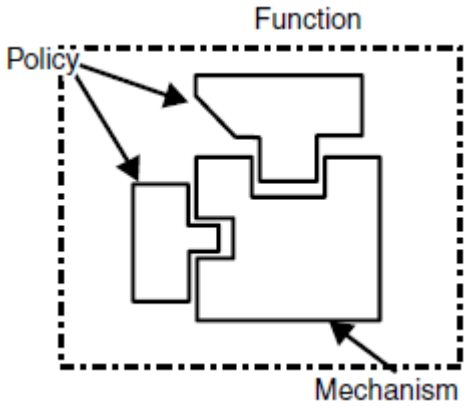
Modelo mais detalhado de uma PM



O Futuro da Internet (2012.2)

Mecanismo e Política

- Modelo para a separação entre o mecanismo e a política.
- Um protocolo é composto por um conjunto de funções que realizam os requisitos básicos daquele protocolo.
- A escolha das funções é feita baseada na região de operação para a qual o protocolo está sendo criado e o nível de serviço desejado como resultado da sua operação.

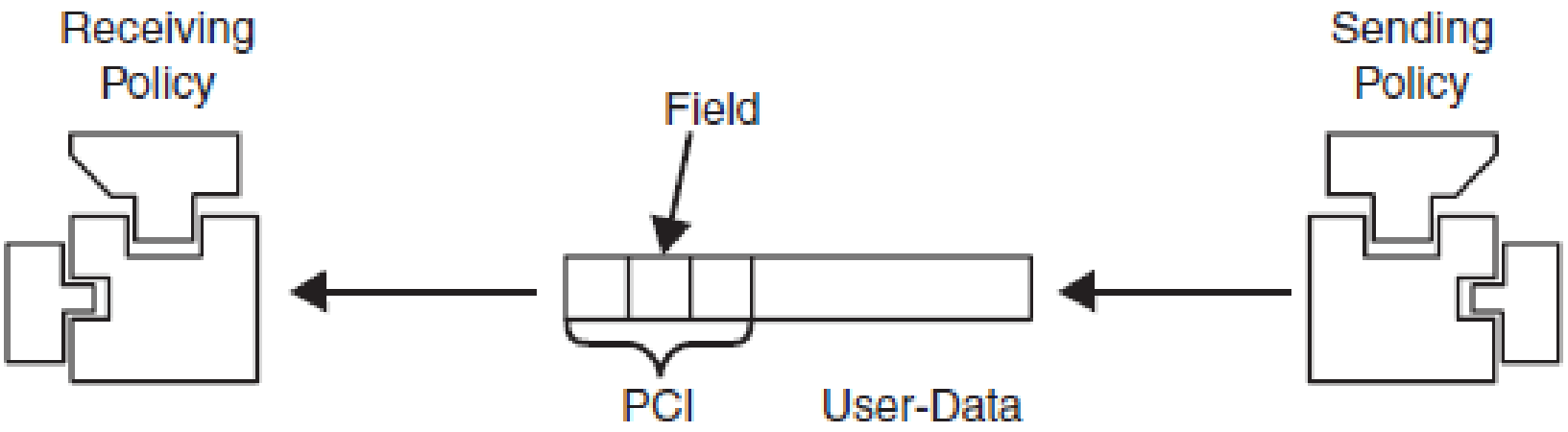


Mecanismo e Política

- Mecanismos são estáticos e não mudam após a especificação do protocolo.
  - Ex.: ordem de interpretação dos campos de um (N)-PCI
- Tipos de políticas em geral aparecem em pares:
  - Política de transmissão e política de recepção
  - Detecção de erros:
    - a política de transmissão calcula o polinômio e o mecanismo o insere na PDU
    - A política de recepção calcula o polinômio de uma PDU entrante, e o mecanismo compara o resultado com o campo no PCI.
- Exceções: políticas de inicialização, associadas a temporizadores, etc.

Coordenação entre os Mecanismos

- Através da troca de campos de informação específicos na (N)-PCI.
- Uma única PDU pode transportar campos para diversos mecanismos na (N)-PCI.



Um mecanismo e diversas Políticas

- Mecanismo de controle de fluxo de janela deslizante.
- A janela deslizante faz parte da especificação do protocolo.
  - Após a especificação este mecanismo não é modificado.
- Diversas políticas para o controle de fluxo:
  - Estender o crédito ao receber uma PDU
  - Envio periódico de novo crédito
- Políticas diferentes podem ser usadas para conexões diferentes ao mesmo tempo.

Mecanismos e Políticas

- Reconhecimento é um mecanismo, mas *quando* enviar um *ack* é uma política.
- Os protocolos devem incluir um mecanismo para especificar ou negociar a política para todos os mecanismos durante a sincronização ou estabelecimento.
- Geralmente mudar a política durante a fase de transferência de dados requer sincronização que é essencialmente equivalente ao estabelecimento de um novo fluxo ou conexão.

Escolha da Política

- A escolha da política depende das características do tráfego da associação (N-1) e da *qualidade de serviço* (QoS) requerida pelo usuário.
- A tarefa da (N)-PM é traduzir estas características de QoS requeridas pela (N+1)-PM numa escolha particular de mecanismos e políticas baseadas no serviço da (N-1)-PM.
- Protocolos próximos ao meio possuem políticas dominadas pelas características do meio, enquanto que protocolos distantes do meio possuem uma maior variedade de políticas aplicáveis.

Mecanismos de “novos protocolos”

- Nenhum novo mecanismo foi proposto em aproximadamente 25 anos.
- Mudanças consistiram basicamente em rearranjo de cabeçalhos e velhos mecanismos com políticas diferentes.
- Os mecanismos de um protocolo são fixados em tempo de especificação, enquanto que as políticas selecionadas são adiadas até a sincronização ou estabelecimento.

Mecanismos e Políticas

- Os mecanismos são fixos, mas uma política apropriada pode torná-los nulos.
- Exemplo:
  - Sempre se considerou necessário outro protocolo de transporte para voz.
  - Para voz, as PDUs devem ser ordenadas, mas podem se toleradas pequenas falhas de sequência.
  - No entanto, não é necessário um novo protocolo. Basta modificar a política de reconhecimento... e mentir! Não há nenhum requisito em dizer a verdade! Se a falha for pequena, envie um *ack* de qualquer modo, apesar de nem todos os dados terem sido recebidos.

Semânticas do Ack

- Um Ack não significa, como se pensa normalmente, “Eu recebi”.
- Significa: “Eu não vou pedir uma retransmissão” ou “Estou satisfeito com o que recebi”!
- Neste livro o que é chamado em outros protocolos como “mecanismo de reconhecimento” aqui é referenciado como “controle de retransmissão”.

QoS versus NoS

- QoS = Qualidade de Serviço
  - Conjunto de características desejadas para a comunicação tais como: largura de banda, atraso, taxa de erros, *jitter*, etc.
  - Diversos parâmetros foram especificados, mas "quando um parâmetro de QoS é alterado, que políticas do protocolo mudam e por que?". Normalmente a resposta é "nenhum"!
  - Causas:
    - Qualquer mudança de política que afete o parâmetro seria uma questão de gerenciamento de recursos (considerado um domínio da implementação)
    - Atraso: um protocolo pode tentar minimizar piorar o atraso mas não pode fazer nada para melhorá-lo. Parâmetros deste tipo são chamados de NoS (*Nature of Service*).

QoS e NoS

- NoS:
  - Parâmetros determinados em grande parte pela “natureza”.
  - Podemos ser capazes de evitar piorá-los, mas há pouco ou nada que possamos fazer para melhorá-lo!
- QoS representa um conjunto de características que uma (N+1)-PM deseja de uma (N)-PM para uma instância particular de comunicação.
- NoS representa o conjunto de características que uma (N-1)-PM provê realmente e provavelmente será capaz de prover no futuro.
- A (N)-PM usa a diferença entre a QoS e a NoS para selecionar o protocolo, mecanismos e políticas para casar o desejo com a realidade.

Alguns Mecanismos de Transferência de Dados

Roteiro

- Delimitação
- Sincronização do Estado Inicial
- Seleção de Política
- Endereçamento
- Identificação de Fluxo ou Conexão
- Repasse (*Relaying*)
- Multiplexação
- Ordenação
- Fragmentação/Remontagem
- Combinação/Separação
- Corrupção dos Dados
- Detecção de Perdas e Duplicatas
- Controle de Fluxo
- Controle de Retransmissão ou Reconhecimento
- Compressão
- Autenticação
- Controle de Acesso
- Integridade
- Confidencialidade
- Não repúdio
- Atividade

Delimitação

- Mecanismo usado para indicar o início e o fim de uma PDU.
- Métodos básicos: delimitação externa e interna.
- Delimitação externa:
  - Padrão especial de bits (sequência de *flag*)
    - Transparência garantida com mecanismo de “escape”.
  - Uso da camada inferior para delimitar a PDU:
    - Campo de comprimento na (N-1)-PCI
    - Codificação de bits na camada física (Ex.: Codificação de Manchester para delimitação de quadros MAC no Ethernet)

Delimitação

- Delimitação Interna:
  - A PDU contém um campo de comprimento como um elemento da PCI.

Sincronização do Estado Inicial

- Inicialização do estado compartilhado das PMs antes do início da transferência de dados.
- Mecanismos:
  1. Criação de associações locais entre a (N+1)-PM e a (N-1)-PM; sem troca de PDUs. Usada por protocolos que requerem estado compartilhado mínimo (ex.: UDP).
  2. O anterior mais a troca de PDUs de pedido e resposta, apresentação em duas vias (*two-way handshake*) usado por protocolos que não possuem mecanismos de realimentação. Ex.: HDLC, X.25, aplicações)
  3. Anterior mais um *ack* do iniciador quando chega a resposta, apresentação em três vias (*three-way handshake*) usado por protocolos com realimentação (Ex.: TCP)
  4. Mecanismo simples baseado em temporizador limitando o tempo de vida máximo de uma PDU: tempo máximo que o transmissor tentará reenviar a PDU, e tempo máximo que o receptor esperará antes de enviar o reconhecimento.

Seleção de Política

- Este mecanismo permite:
  - a seleção de política durante a alocação e
  - a mudança de políticas durante a transferência de dados em certas circunstâncias.
- Exemplos de protocolos:
  - HDLC: é possível escolher diversas opções.

Endereçamento

- Protocolos que operam em ambientes com múltiplo acesso devem conter alguma forma de identificar a origem e o destino das PDUs.
- Isto é feito incluindo campos de *endereços* no PCI.
- Os endereços devem ser grandes o suficiente para nomear todos os elementos que podem se comunicar sem depender da camada superior.

Identificação de Fluxo ou Conexão

- Protocolos que suportam múltiplas instâncias de comunicação (i.e., associações, fluxos ou conexões entre as mesmas duas estações) também requerem um identificador de conexão ou de fluxo.
- Tradicionalmente isto é feito usando "*port-ids*".
- Se estes identificadores forem únicos a toda a camada e não apenas ao protocolo, então podem ser usados para multiplexar fluxos de múltiplos protocolos.

Repasse (*Relaying*)

- Muitas redes não são malhas completamente conectadas.
- Portanto, alguns protocolos repassam uma PDU de uma PM para a próxima.
- É preciso incluir na PCI um elemento que contenha o endereço do destino. Em muitos casos, ele contém também o endereço da origem.
- Quando uma PDU chega, o mecanismo de repasse inspeciona o endereço e determina se ela está endereçada a uma de suas (N+1)-PMs.
  - Em caso afirmativo, ela é entregue à (N+1)-PM correspondente.
  - Em caso negativo, ele determina qual (N-1)-PM pode levar a PDU o mais próximo possível ao destino. Isto é conhecido como *encaminhamento* *(forwarding)*.

Multiplexação

- É o mapeamento de fluxos de (N)-PMs em fluxos de (N-1)-PMs.

Ordenação

- Muitos (mas não todos) protocolos assumem uma ordenação simples:
  - As PDUs chegam na mesma ordem em que foram enviadas.
- No entanto, alguns serviços de comunicação não garantem esta propriedade.
- Este mecanismo é implementado incluindo um número de sequência com um elemento da PCI que é incrementado em:
  - Unidades de bytes (octetos) de acordo com o comprimento dos dados do usuário na PDU
  - Unidades de PDUs
  - Que permitem que as PDUs sejam reordenadas no receptor.

Fragmentação/Remontagem

- Restrições práticas frequentemente requerem que SDUs e dados do usuário sejam *fragmentados* em PDUs menores para transmissão e depois sejam *remontadas* no outro lado.
- Elementos na PCI:
  - Bit que indica se este é o último fragmento
  - Número de sequência
  - Número de fragmentos
  - Campo de comprimento da PDU (também usado para delimitação e detecção de corrupção dos dados)

Combinação/Separação

- Busca de eficiência *combinando* SDUs em uma única PDU.
- Técnicas usadas:
  - SDUs de comprimento fixo
  - Cadeia de campos de comprimento

Corrupção dos Dados

- O conteúdo de uma PDU pode ser corrompido durante a transmissão.
- Mecanismos para lidar com o problema:
  - Uso de uma soma de verificação ou CRC para detectar a corrupção:
    - O código é calculado sobre a PDU recebida. Se falhar a PDU é descartada e algum outro mecanismo garante a retransmissão
  - Uso de código corretor de erro.
    - FEC (*Forward Errror Correcting*) pode detectar e corrigir alguns erros, evitando o descarte da PDU.

Corrupção dos Dados: Escolha dos Códigos

- Dependem da natureza do ambiente do erro:
  - Protocolos próximos ao meio elétrico são mais sujeitos a erros em rajadas e, portanto, requerem códigos que possam detectar rajadas de erros.
  - O meio óptico possui características distintas de erros e, portanto, requer um tipo diferente de código.
  - Protocolos distantes do meio têm maior probabilidade de encontrar erros de um único bit (falha de memória) e, portanto, usam códigos de erros apropriados.
- Devem ser considerados:
  - Análise de erros tanto do protocolo como do ambiente operacional.
  - Efeito do tamanho da PDU na força do polinômio.

Detecção de Perdas e Duplicatas

- Congestionamento ou PDUs com erro levam ao descarte.
- Como elas devem ser retransmitidas, pode levar à geração de duplicatas.
- O número de sequência (no PCI) usado para ordenação é também usado para a detecção de perda ou duplicata.
- Diante de perdas, pode ser acionado o controle de retransmissões.
- Duplicatas são descartadas.

Controle de Fluxo

- Usado para evitar que o transmissor envie dados mais rapidamente do que o destino pode recebê-los.
- Formas básicas de controle de fluxo:
  - Esquema de créditos:
    - O destino diz ao receptor quantas mensagens pode enviar antes de receber mais créditos (frequentemente associado ao mecanismo de reconhecimento).
  - Esquema de taxas:
    - O destino indica ao transmissor a que taxa ele pode enviar os dados.

Controle de Retransmissão ou Reconhecimento

- Usado pelo destino para dizer ao transmissor que PDUs foram recebidas com sucesso.
- Esquema prevalente inclui como elemento da PCI um número de sequência que indica que todas as PDUs com números de sequência inferiores ao mesmo foram recebidas.
- Se o transmissor não receber um *ack* para um dado número de sequência após um determinado período de tempo, ele automaticamente retransmite todas as PDUs até a última PDU enviada.

Controle de Retransmissão ou Reconhecimento

- Quando um *ack* é recebido, o transmissor pode deletar PDUs com números de sequência inferiores, de sua lista de potenciais retransmissões.
- Para ambientes com grandes valores do produto largura de banda-atraso, é usado um mecanismo mais complexo de *ack seletivo* ou *reconhecimento negativo (nack)*, para informar ao transmissor de erros específicos, limitando assim o número de PDUs retransmitidas.
- Retransmissões podem provocar atrasos inaceitáveis.

Mecanismo de Janela Deslizante

- Diversos mecanismos usam os números de sequência das PDUs:
  - Controles de perda e duplicação, controle de fluxo, controle de retransmissões.
- Janela Deslizante:
  - Transmissor e Receptor mantêm uma janela deslizante baseada nos números de sequência das PDUs que eles transmitem ou recebem.
  - O limite mais à esquerda representa a última PDU reconhecida ou da qual se recebeu um reconhecimento.
  - A largura da janela corresponde à quantidade de crédito fornecido pelo controle de fluxo.

Mecanismo de Janela Deslizante

- A largura da janela do transmissor representa o número de PDUs ou de octetos que podem ser enviados.
- A largura da janela de recepção corresponde ao número de PDUs ou octetos que o receptor espera receber antes que o crédito expire...
  - Qualquer PDU fora da janela é descartada.
  - Qualquer PDU com um número de sequência inferior à da borda esquerda é uma duplicata e será descartada.
- A borda direita é o maior número de sequência que o transmissor pode transmitir (antes de receber mais crédito) ou que o receptor pode receber.

Mecanismo de Janela Deslizante

- O mecanismo de retransmissão modifica apenas a borda esquerda da janela.
- O controle de fluxo modifica apenas a borda direita da janela.
- Qualquer ligação entre os dois é feito através de políticas.

Compressão

- Usada para melhorar a eficiência da transmissão aplicando compressão de dados aos dados do usuário.
- A política deste mecanismo seleciona o algoritmo de compressão a ser utilizado.

Autenticação

- Usada para que o destino autentique a identidade da origem.
- A política associada determina o algoritmo particular de autenticação a ser usado.
- Técnicas de criptografia são geralmente empregadas para prover maior confiança na troca.

Controle de Acesso

- Usado para prevenir o uso não autorizado de recursos.
- Em geral, o controle de acesso é realizado apenas após a autenticação.

Integridade

- Provê proteção contra a inserção ou descarte de PDUs de forma não autorizada.
- Maior integridade do que as fornecidas pelos mecanismos de:
  - Detecção de corrupção dos dados
  - Detecção de perdas ou duplicatas
- Políticas: algoritmos de criptografia e tamanho da chave associada.

Confidencialidade

- Tenta garantir que o conteúdo dos dados do usuário transportado em PDUs ou toda as PDUs de uma comunicação não sejam divulgados para processos ou pessoas não autorizadas.
- Políticas: algoritmos de criptografia e tamanho da chave associada.

Não repúdio

- Tenta garantir que nenhum processo que tenha participado de uma interação possa negar ter participado na mesma.
- Normalmente são usados métodos criptográficos para implementar este mecanismo.

Atividade (*keepalive*)

- Usado em conexões que passam longos períodos sem tráfego.
- Permite que os correspondentes determinem que seus parceiros permanecem em um estado consistente.
- Política: frequência ou condições para invocar o mecanismo.

Fases da Operação

Fase da Operação

1. Registro
2. Estabelecimento ou sincronização
3. Transferência de dados

- Devem ser executados procedimentos associados com estas três fases tanto pelos transmissores como pelos receptores, mesmo que nenhuma PDU seja trocada.
- Devem ser realizadas na ordem indicada.

A fase de Registro

- Esta fase cria, mantém, distribui e deleta a informação dentro de uma camada que é necessária para criar instâncias da comunicação.
- Esta fase torna um objeto e suas funções conhecidas da rede.
- Informações de endereçamento são inicializadas nos diretórios apropriados (e tabelas de roteamento).
- Parâmetros são inicializados que caracterizam a comunicação na qual este protocolo pode participar.
- São estabelecidas regras de controle de acesso.
- São fixados limites da política, etc.

A fase de Registro

- Esta fase sempre esteve presente mas era frequentemente ignorada por que fazia parte de uma configuração inicial confusa (e frequentemente manual).
- Quando as PMs são criadas na fase de estabelecimento, herdarão o conjunto de atributos associados com seus protocolos que foram gravadas durante a fase de registro.

A fase de Registro

- **Operação de Registro:** inclui as informações necessárias para criar uma instância disponível dentro da rede.
- A informação está disponível apenas para sistemas dentro do escopo deste protocolo e da sua camada.
- **Operação de cancelamento** (*deregistration*): deleta o registro do protocolo da rede. Em geral deve esperar até que todas as instâncias tenham saído da fase de alocação; ou seja, que não haja fluxos ativos.

Ativação/desativação

- Operação tradicional de “desligar” um recurso sem deletar do sistema o conhecimento de que ele existe.

Exemplos de Registro

- Até o momento, muitas arquiteturas dependeram de procedimentos ad hoc para o registro.
- Efetuados através:
  - Gerência de rede (estabelecendo circuitos virtuais permanentes).
  - Padrão que define soquetes “bem conhecidos”.
  - DHCP
  - Atribuição de endereços MAC
  - Gerenciamento de chaves
- Com o surgimento e uso de protocolos de diretório e protocolos de atribuição de endereços, a fase de registro está se tornando menos ad hoc e muito mais uma fase regular automatizada.

A fase de Estabelecimento ou Sincronização

- Esta fase cria, mantém e deleta o estado compartilhado necessário para dar suporte às funções da fase de transferência de dados.
- A fase de sincronização garante que as PMs tenham inicialmente um estado consistente (não necessariamente o mesmo) de informação.
- Exemplos:
  - Ligações entre a (N+1)-PM e a (N)-PM (sem conexão)
  - Troca inicial de informações de estado para sincronizar o estado entre duas PMs (conexão)

A fase de Estabelecimento ou Sincronização

- É nesta fase que a solicitação de QoS específica aceitável para o usuário para a transferência de dados é feita (ou modificada) se não tiver sido fixada na fase de registro.
- Classes abrangentes de protocolos:
  - Na faixa (*in-band*): mesmo protocolo para as fases de sincronização e transferência de dados
  - Fora da faixa (*out-of-band*): protocolos distintos para as fases de sincronização e transferência de dados

A Fase de Transferência de Dados

- Entra-se nesta fase quando a transferência de dados real é realizada de acordo com a QoS solicitada entre os endereços especificados durante uma das duas fases anteriores.
- Para os protocolos de aplicação, esta fase pode ser subdividida em subfases especializadas.