República Federativa do Brasil
Ministério da Economia
Instituto Nacional da Propriedade Industrial

(21) PI 1012243-5 A2

(22) Data do Depósito: 17/03/2010

(43) Data da Publicação Nacional: 18/08/2020

(54) Título: SISTEMA E MÉTODO PARA COMPRESSÃO DE VÍDEO MULTI-FLUXO

(51) Int. Cl.: G06K 9/36.

(30) Prioridade Unionista: 07/08/2009 US 12/538,041; 23/03/2009 US 61/210,888.

(71) Depositante(es): ONLIVE, INC..

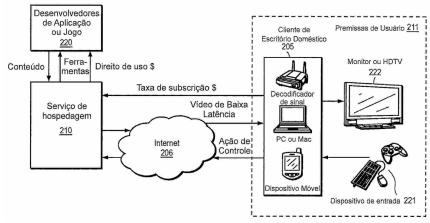
(72) Inventor(es): STEPHEN G. PERLMAN; ROGER VAN DER LAAN; TIMOTHY COTTER; SCOTT FURMAN; ROBERT MCCOOL; IAN BUCKLEY.

(86) Pedido PCT: PCT US2010027725 de 17/03/2010

(87) Publicação PCT: WO 2010/111100 de 30/09/2010

(85) Data da Fase Nacional: 23/09/2011

(57) Resumo: SISTEMA E MÉTODO PARA COMPRESSÃO DE VÍDEO MULTIFLUXO. A presente invenção refere-se a um servidor de video game 5 recebendo entradas de usuário relacionados a um video game online, para render uma sequência de imagens de vídeo; um primeiro codificador de fluxo para comprimir a sequência de imagens de vídeo e gerar um fluxo de vídeo ao vivo durante uma sessão de jogo ao vivo com um usuário de um aparelho de cliente, o primeiro codificador de fluxo, recebendo sinais de retorno de canal do aparelho do cliente e responsivamente adaptando a compressão da sequência de imagens de vídeo baseado nos sinais de retorno de canal, o primeiro codificador de fluxo transmitindo continuamente o fluxo de vídeo ao vivo ao aparelho do cliente durante a sessão de jogo ao vivo com o usuário; um segundo codificador de fluxo para comprimir a sequência de imagens de vídeo numa qualidade de vídeo específica e/ou taxa de compressão não m - relacionada ao sinal de retorno do canal durante a sessão de jogo ao vivo . com o usuário, assim gerando qualidade alta de fluxo de vídeo (HQ), a HQ de fluxo de video tendo qualidade de vídeo relativamente maior e/ou taxa de (...).



Relatório Descritivo da Patente de Invenção para **"SISTEMA E MÉTODO PARA COMPRESSÃO DE VÍDEO MULTIFLUXO".**

PEDIDO RELACIONADO

Este pedido reivindica prioridade para o Pedido provisório US de No. Serial 61/210.888, depositado em 23 de março de 2009, intitulado, "Sistema e Método para Compressão de Vídeo Usando Realimentação de Informação", que é uma continuação em parte do Pedido US de No. Serial US 12/359.150 co-pendente, depositado em 23 de janeiro de 2009, intitulado, "Sistema e Método para Proteção de Certos Tipos de Dados Multimídia Transmitidos Sobre um Canal de Comunicação", e é uma continuação em parte do Pedido US de No. Serial US 11/999.475 co-pendente, depositado em 5 de dezembro de 2007, intitulado, "Hospedagem e Difusão de Eventos Virtuais Usando Vídeo Interativo de Fluxo Contínuo" que é uma continuação em parte (CIP) do Pedido de No. Serial 10/315.460 depositado em 10 de dezembro de 2002 intitulado, "APARELHO E MÉTODO PARA JOGOS EM VÍDEO SEM FIO", que é designado para o requerente do presente pedido CIP.

CAMPO DA TÉCNICA

A presente invenção refere-se em geral ao campo de sistemas de processamento de dados que melhoram a habilidade de um usuário para manipular e acessar mídia de áudio e vídeo.

ANTECEDENTES

Mídia gravada de áudio e imagem em movimento tem sido um aspecto da sociedade desde os tempos de Thomas Edison. No início do século 20 ouve uma ampla distribuição de mídia de áudio gravada (cilindros e discos) e mídia de imagem em movimento (cinemas (nickelodeons) e filmes), mas ambas as tecnologias ainda estavam em sua infância. No final dos anos 1920s imagens em movimento foram combinadas com áudio com base em um mercado de massa, seguidos por imagens em movimento coloridas com áudio. A difusão de rádio evoluiu gradualmente largamente suportada por propaganda em uma forma de difusão de mídia de áudio para mercado de massa. Quando foi estabelecido um padrão de difusão de

televisão (TV) no meio dos anos 1940s, a televisão juntou-se ao rádio como uma forma de difusão de mídia de mercado de massa trazendo imagens em movimento previamente gravadas ou ao vivo para dentro de casa.

No meio do século 20, uma grande porcentagem de residências US tinha tocadores de gravação fonográfica para ticar mídia de áudio gravada, um rádio para receber difusão de áudio ao vivo e um conjunto de televisão para tocar difusão mídia de áudio / vídeo (A / V)ao vivo. Muito frequentemente estes 3 "tocadores de mídia" (tocador de gravação, rádio e TV) foram combinados em um gabinete compartilhando alto-falantes comuns o que se tornou uma "central de mídia" para as residências. Embora as escolhas de mídia fossem limitadas para o consumidor, o "ecossistema" de mídia era bastante estável. A maior parte dos consumidores sabia como usar os "tocadores de mídia" e eram capazes de aproveitar completamente seus recursos. Ao mesmo tempo, os publicadores de mídia (largamente os estúdios de filmes e televisões, e as companhias de música) eram capazes de distribuir suas mídias tanto para cinemas como para as residências sem sofrer de pirataria ou "segundas vendas", ou seja, a revenda de mídia usada. Tipicamente as publicadoras não derivam receita de segundas vendas, e, portanto, isto reduz as receitas que os publicadores poderiam de outra forma derivar do comprador de mídia usada para novas vendas. Embora certamente fossem usadas gravações vendidas durante a metade do século 20, estas vendas não tinham o grande impacto nos publicadores de gravação, diferente de um filme ou programa de vídeo - que é tipicamente assistido uma vez ou apenas umas poucas vezes por um adulto - uma trilha de música pode ser ouvida centenas ou mesmo milhares de vezes. Assim, mídia de música é muito menos "perecível" (ou seja, a mesma tem valor duradouro para um consumidor adulto) do que uma mídia de filme / vídeo. Uma vez tendo sido comprada uma gravação, se o consumidor gostou da música, o consumidor provavelmente irá mantê-la por um longo tempo.

Do meio do século 20 até os dias atuais, o ecossistema de mídia tem passado por uma série de mudanças radicais, tanto em benefício como em detrimento dos consumidores e publicadores. Com a introdução da

ampla difusão de gravadores de áudio, especialmente fitas cassete com som estéreo de alta qualidade, certamente houve um maior grau de conveniência do consumidor. Mas isto também marcou o início do que é agora uma prática amplamente difundida com mídia de consumidor: pirataria. Certamente, muitos consumidores usaram as fitas cassete para gravar suas próprias gravações por conveniência, mas uma quantidade crescente de consumidores (por exemplo, estudantes em um dormitório com acesso direto a coleções de gravações de outros) faria cópias pirateadas. Também, consumidores gravariam músicas tocadas pelo rádio em vez de comprar uma gravação ou fita do publicador.

O advento do VCR para consumidor levou a ainda maior conveniência do consumidor, uma vez que um VCR podia ser ajustado para gravar um programa de TV para ser assistido em um momento posterior, e isto também levou a criação de um negócio de aluguel de vídeo, onde filmes bem como programas de TV podiam ser acessados em uma base sob demanda. O rápido desenvolvimento do mercado de massa dos dispositivos de mídia domésticos a partir do meio dos anos 1908s levou a um nível sem precedentes de escolha e conveniência para o consumidor, e também levou a uma rápida expansão do mercado de mídia publicitária.

Hoje, os consumidores se deparam com uma pletora de escolhas de mídia bem como uma pletora de dispositivos de mídia, muitos dos quais são ligados a formas particulares de mídia de publicadores particulares. E o consumidor ávido por mídia pode ter uma pilha de dispositivos conectados a TVs e computadores em várias salas da residência, resultando em um "ninho de rato" de cabos para um ou mais conjuntos de TV e / ou computadores pessoais (PCs) bem como a um grupo de controles remotos. (No contexto do presente pedido, o termo "computador pessoal" ou "PC" refere-se a qualquer tipo de computador adequado para uso em residência ou escritório, incluindo um computador de mesa, um Macintosh® ou outros computadores não Windows, dispositivos compatíveis com Windows, variações de Unix, computadores portáteis, etc.). Estes dispositivos podem incluir uma console de jogos de vídeo, VCR, tocador de DVD, amplificador /

processador de som ambiente para áudio, conjuntos de decodificadores de satélite, etc. E, para um consumidor ávido, pode haver múltiplos dispositivos com funções similares devido a questões de compatibilidade. Por exemplo, o consumidor pode possuir tanto um DVD-HD como um tocador de BDB Blu-ray, ou tanto um sistema de jogos de vídeo Microsoft Xbox® como um Sony Playstation®. De fato, devido à incompatibilidade de alguns jogos através de versões de consoles de jogos, o consumidor pode possuir tanto um XBox como uma versão posterior, tal como um Xbox 360®. Frequentemente, os consumidores são confundidos como com qual entrada de vídeo ou controle remoto usar. mesmo após um disco ser colocado no tocador correto (por exemplo, DVD, HD-DVD, Blu-ray, Xbox ou Playstation), a entrada de vídeo e áudio ser selecionada para aquele dispositivo, e o controle remoto correto ser achado, o consumidor ainda se depara com desafios técnicos. Por exemplo, no caso de um DVD de tela panorâmica (wide-screen), o usuário pode ter que primeiro determinar e em seguida configurar a relação de aspecto correta em sua TV ou tela de monitor (por exemplo, 4:3, Completa, Aproximação (Zoom), Aproximação Panorâmica (Wide Zoom), Panorâmica de Cinema, etc.). De maneira similar, o usuário pode ter que primeiro determinar e em seguida configurar o formato correto do sistema de som ambiente de áudio (por exemplo, AC-3, Dolby Digital, DTS, etc.). Muitas vezes, o consumidor não está consciente de que ele pode não estar aproveitando o conteúdo da mídia com toda a capacidade de seu sistema de televisão e áudio (por exemplo, assistindo um filme achatado ou com relação de aspecto errada, ou ouvindo o áudio em estéreo em vez de em som ambiente).

Cada vez mais, dispositivos de mídia com base em Internet têm sido adicionados a pilha de dispositivos. Dispositivos de áudio tais como o sistema de áudio de fluxo contínuo de Música Digital Sonos® diretamente da Internet. Da mesma forma, dispositivos como o gravador e tocador de entretenimento Slingbox™ grava vídeo e distribui o mesmo em fluxo contínuo através de uma rede doméstica ou para fora através da Internet onde o mesmo pode ser assistido através de um PC. E serviços de Televisão por

Protocolo Internet (IPTV) oferecem serviços semelhantes à TV a cabo através de Linha Digital de Assinante (DSL) Ou outras conexões domésticas de Internet. também tem havido recentes esforços para integrar múltiplas funções de mídia em um único dispositivo, tal como Central de Mídia Moxi® e PCs rodando Edição de Central de Mídia (Media Center Edition) do Windows XP. Embora cada um destes dispositivos ofereça um elemento de conveniência para as funções que o mesmo executa, cada um carece de acesso simples e ubíquo a maior parte de mídia. Adicionalmente, estes dispositivos frequentemente custam centenas de dólares para fabricar, frequentemente devido à necessidade de processamento e / ou armazenamento locais que são caros. Adicionalmente, estes dispositivos eletrônicos modernos de consumidor tipicamente consomem uma grande quantidade de energia, mesmo quando inativos, o que significa que os mesmos são caros ao longo do tempo e desperdiçam recursos de energia. Por exemplo, um dispositivo pode continuar a operar se o consumidor esquece de desligá-lo ou comuta para uma entrada de vídeo diferente. E, devido a nenhum dos dispositivos ser uma solução completa, o mesmo precisa ser integrado com uma outra pilha de dispositivos na residência, o que ainda deixa o usuário com um ninho de rato de fios e um mar de controles remotos.

Além disso, quando muitos dispositivos mais novos com base em Internet não funcionam corretamente, os mesmos tipicamente oferecem mídia de uma forma mais genérica do que poderia estar disponível de outra forma. Por exemplo, dispositivos que exibem vídeo através da Internet frequentemente exibem apenas o material do vídeo, não os "extras" interativos que frequentemente acompanham os DVDs, tais como os vídeos de "bastidores", jogos , ou comentários do diretor. Isto é devido ao fato de que frequentemente o material interativo é produzido em um formato particular destinado a um dispositivo particular que trata a interatividade localmente. Por exemplo, cada disco de DVD, HD-DVDs e Blu-ray têm seu próprio formato interativo particular. Qualquer dispositivo de mídia doméstico ou computador local que poderia ser desenvolvido para suportar todos os formatos populares exigiria um nível de sofisticação e flexibilidade que provavelmente

o tornaria proibitivamente caro e complexo para o consumidor operar.

Adicionando ao problema, se um novo formato é introduzido posteriormente o dispositivo local pode não ter a configuração de hardware para suportar o novo formato, que pode significar que o consumidor tenha que comprar um dispositivo de mídia local atualizado. Por exemplo, se vídeo de maior resolução ou vídeo estereoscópico (por exemplo, um fluxo contínuo para cada olho) forem introduzidos em uma data posterior, o dispositivo local pode não ter a capacidade computacional para decodificar o vídeo, ou o mesmo pode não ter o hardware para fornecer o vídeo no novo formato (por exemplo, assumindo que estereoscopia é alcançada através de vídeo a 120fps sincronizado com óculos com obturadores, com 60fps entregues para cada olho, se o hardware do vídeo do consumidor pode suportar apenas vídeo de 60fps esta opção estaria indisponível afora uma compra de hardware atualizado).

A questão da obsolescência e complexidade do dispositivo de mídia é um problema sério quando vem para mídia interativa sofisticada, especialmente jogos de vídeo.

Aplicações modernas de jogos de vídeo são largamente divididas em quatro plataformas de hardware não portáveis principais: Sony PlayStation® 1, 2 e 3 (PS1, PS2, e PS3); Microsoft Xbox® e Xbox 360®; e Nintendo Gamecube® e Wii™; e jogos com base em PC. Cada uma destas plataformas é diferente das outras de modo que os jogos escritos para rodar em uma plataforma geralmente não rodam em outra plataforma. Também podem haver problemas de compatibilidade de uma geração do dispositivo para a próxima. Apesar de a maior parte dos desenvolvedores de software de jogos criarem softwares de jogos que são independentes de uma plataforma particular, a fim de rodar um jogo particular em uma plataforma específica uma camada de software proprietário (frequentemente chamada "mecanismo de desenvolvimento de jogo") é necessária para adaptar o jogo para uso em uma (plataforma específica. cada plataforma é vendida para o consumidor como uma "console" (ou seja, uma caixa separada ligada a uma TV ou monitor / alto-falantes) ou é ela própria um PC. Tipicamente, os jogos

de vídeo são vendidos em mídia ótica tal como um Blu-ray DVD, DVD-ROM ou CD-ROM, que contém o jogo de vídeo incorporado como uma aplicação de software sofisticada em tempo real. Como as velocidades de banda larga residenciais tem aumentado, os jogos de vídeo estão se tornando cada vez mais disponíveis para download.

As exigências específicas para obter compatibilidade de plataforma com software de jogo de vídeo é extremamente exigente devido à natureza de tempo real e alta exigência computacional de jogos de vídeo avançados. Por exemplo, poder-se-ia esperar compatibilidade total de jogo de uma geração para a próxima dos jogos de vídeo (por exemplo, de XBox para XBox 360, ou de Playstation 2 ("PS2") para Playstation 3 ("PS3"), uma vez que existe uma compatibilidade geral de aplicações de produtividade (por exemplo, Microsoft Word) de um PC para outro com uma unidade de processamento ou núcleo mais rápido. Entretanto, este não é o caso com jogos de vídeo. Devido a os fabricantes de jogos estarem tipicamente buscando a performance mais alta possível para um dado ponto de preço quando uma geração de jogos de vídeo é liberada, mudanças de arquitetura dramáticas ao sistema são feitas frequentemente de modo que muitos jogos escritos para o sistema de geração anterior não funcionam no sistema de geração posterior. Por exemplo, XBox foi baseado na família de processadores x86, enquanto que o XBox 360 foi baseado na família PowerPC.

Podem ser utilizadas técnicas que emulam uma arquitetura anterior, mas dado que jogos de vídeo são aplicações de tempo real, frequentemente é inviável obter exatamente o mesmo comportamento em uma emulação. Isto em detrimento ao consumidor, o fabricante de console de jogos de vídeo e o publicador de software de jogos de vídeo. Para o consumidor, isto significa a necessidade de manter tanto a velha como a nova geração de consoles de jogos conectada a TV para ser capaz de jogar todos os jogos. Para o fabricante de console isto significa custo associado com emulação e adoção mais lenta das novas consoles. E para o publicador isto significa que múltiplas versões de novos jogos podem ter que ser liberadas a fim de alcançar todos os consumidores em potencial - não apenas

liberando uma versão para cada marca de jogo de vídeo (por exemplo, XBox, Playstation), mas frequentemente uma versão para cada versão de uma dada marca (por exemplo, PS2 e PS3). Por exemplo, uma versão separada do "Madden NFL 08" da Electronic Arts foi desenvolvida para XBox, XBox 360, PS2, PS3, Gamecube, Wii, e PC, dentre outras plataformas.

Dispositivos portáteis , tais como telefones celulares e tocadores de mídia portáteis também apresentam desafios para desenvolvedores de jogos. Cada vez mais estes dispositivos são conectados a redes de dados sem fio e são capazes de baixar jogos de vídeo. Mas, existe uma ampla variedade de telefones celulares e dispositivos de mídia no mercado, com uma ampla variedade de diferentes resoluções de exibição e recursos computacionais. Também, devido a estes dispositivos terem tipicamente restrições de consumo de energia, custo e peso, os mesmos tipicamente não possuem hardware de aceleração gráfica avançado tal como uma Unidade de Processamento Gráfico ("GPU"), tal como os dispositivos feitos pela NVIDIA de Santa Clara, CA. Consequentemente, os desenvolvedores de software de jogos tipicamente desenvolvem um dado titulo de jogo simultaneamente para muitos tipos diferentes de dispositivos portáteis. Um usuário pode achar que um dado título de jogo não está disponível para o seu telefone celular ou tocador de mídia portátil particular.

No caso de consoles de jogos residenciais, fabricantes de plataforma de hardware tipicamente cobram um direito de exploração aos desenvolvedores de software de jogos para permitir a publicação de um jogo em sua plataforma. Operadoras de telefone celular sem fio também cobram tipicamente um direito de exploração ao publicador do jogo para baixar um jogo no telefone celular. No caso de jogos de PC, não existe direito de exploração pago para jogos publicados, mas os desenvolvedores de jogos tipicamente se deparam com altos custos devido à maior sobrecarga de serviço ao cliente para suportar uma ampla variedade de problemas de configurações e instalações de PC que podem surgir. Também, os PCs tipicamente apresentam menos barreiras para a pirataria do software de jogo

uma vez que os mesmos são prontamente reprogramáveis por um usuário tecnicamente qualificado e os jogos podem ser pirateados mais facilmente e distribuídos mais facilmente (por exemplo, através da Internet). Portanto, para um desenvolvedor de software de jogos, existem custos e desvantagens em publicar em consoles de jogos, telefones celulares e PCs.

Para publicadores de software de jogos de console e PC, os custos não param aí. Para distribuir jogos através de canais varejo, os publicadores cobram um preço de atacado abaixo do preço de venda para que o varejista tenha uma margem de lucro. O publicador também tipicamente tem que pagar o custo de fabricar e distribuir a mídia física contendo o jogo. Frequentemente também é cobrada ao publicador uma "taxa de proteção de preço" pelo varejista para cobrir possíveis contingências tais como onde o jogo não vende, ou se o preço do jogo é reduzido, ou se o varejista tem que r5estituir parte ou todo o preço de atacado e / ou receber uma devolução de um jogo do comprador. Adicionalmente, os varejistas tipicamente também cobram taxas aos publicadores para ajudar o mercado de jogos em panfletos de propaganda. Além disso, os varejistas estão tipicamente cada vez mais comprando de volta os jogos dos usuários que acabaram de jogá-los, e então os vendem como jogos usados, tipicamente não compartilhando nada da venda do jogo usado com o publicador do jogo. Adicionando a carga de custos colocada sobre os publicadores de jogos está o fato de que os jogos frequentemente são pirateados e distribuídos através da Internet para usuários baixarem e fazerem cópias gratuitas.

Conforme as velocidades de banda larga de Internet tem aumentado e a conectividade de banda larga tem se tornado mais difundida nos EUA e por todo o mundo, particularmente para as residências e Internet "cafés" onde PCs conectados a Internet são alugados", os jogos estão cada vez mais sendo distribuídos através de downloads para PCs ou consoles. Conexões de banda larga também são cada vez mais usadas para jogar jogos em linha multijogadores e multijogadores em larga escala (ambos os quais são referenciados na presente invenção pelo acrônimo "MMOG"). Estas mudanças mitigam alguns dos custos e problemas associados com a

distribuição por varejo. Baixar jogos em linha endereça algumas das desvantagens para os publicadores de jogos pelo fato de que os custos de distribuição tipicamente são menores e existem pequenos ou nenhum custo pela mídia não vendida. Mas os jogos baixados ainda são sujeitos a pirataria, e devido ao seu tamanho (frequentemente um tamanho de muitos gigabytes) os mesmos podem demorar um tempo muito longo para baixar. Adicionalmente, múltiplos jogos podem encher completamente pequenos discos, tais como aqueles vendidos com computadores portáteis ou com consoles jogos de vídeo. Entretanto, no âmbito em que os jogos ou MMOGs requerem uma conexão em linha para o jogo ser jogável, o problema de pirataria é mitigado uma vez que usualmente é requerido que o usuário tenha uma conta de usuário válida. Diferente de mídia linear (por exemplo, vídeo e música) que podem ser copiados por uma câmera filmando o vídeo de uma tela de exibição ou um microfone gravando áudio de alto falantes, cada experiência de jogo de vídeo é única, e não pode ser copiada usando uma simples gravação de vídeo / áudio. Portanto, mesmo em regiões onde as leis de direitos autorais não são fortemente aplicadas e a pirataria é excessiva, os MMOGs podem ser protegidos da pirataria e portanto um negócio pode ser suportado. Por exemplo, o MMOG "World of Warcraft" da Vivendi SA foi lançado com sucesso sem sofrer de pirataria por todo o mundo E muitos jogos em linha ou MMOG, tais como o MMOG "Second Life" da Linden Lab geram receita para os operadores dos jogos através de modelos econômicos construídos dentro dos jogos onde recursos podem ser comprados, vendidos e mesmo criados usando ferramentas em linha. Portanto, mecanismos adicionais às compras ou subscrições de software de jogos convencionais podem ser usados para pagar o uso de jogos em linha.

Embora a pirataria possa ser frequentemente mitigada devido à natureza dos jogos em linha ou MMOGs, o operador de jogos em linha ainda se depara com desafios remanescentes. Muitos jogos requerem recursos de processamento locais (ou seja, in-home) substanciais para os jogos em linha ou MMOGs funcionarem adequadamente. Se um usuário tem um computador local de baixa performance (por exemplo, um sem uma GPU, tal como

um computador portátil básico), o mesmo pode não ser capaz de jogar o jogo. Adicionalmente, as consoles de jogos envelhecem, as mesmas ficam adicionalmente atrás do estado da técnica e não são capazes de manipular jogos mais avançados. mesmo assumindo que o PC local do usuário é capaz de controlar as exigências computacionais de um jogo, frequentemente existem complexidades de instalação. Pode haver incompatibilidades de acionador (driver) (por exemplo, se um novo jogo é baixado, o mesmo pode instalar uma nova versão de um acionador de gráfico que torna um jogo instalado previamente, compatível com uma versão mais velha do acionador de gráfico, inoperante). Uma console pode ficar sem espaço em disco conforme mais jogos são baixados. Jogos complexos tipicamente recebem correções baixadas ao longo do tempo a partir do desenvolvedor do jogo quando erros são descobertos e corrigidos, ou se são feitas modificações ao jogo (por exemplo, se o desenvolvedor do jogo descobre que um nível do jogo é muito difícil ou muito fácil de jogar). Estas correções requerem novos downloads. Mas algumas vezes nem todos os usuários completam os downloads de todas as correções. Outras vezes, as correções baixadas introduzem outros problemas de compatibilidade ou de consumo de espaço em disco.

Também, quando se joga o jogo, podem ser requeridos grandes downloads de dados para fornecer gráficos ou informações comportamentais para o PC ou console local. Por exemplo, se o usuário entra em uma sala em um MMOG e encontra um cenário ou personagem construído de dados gráficos ou com comportamentos que não estão disponíveis na máquina local do usuário, então os dados daquele cenário ou personagem têm que ser baixados. isto pode resultar em um atraso substancial enquanto se joga o jogo se a conexão de Internet não for rápida o suficiente. E, se o cenário ou personagem encontrado exigir espaço de armazenamento ou capacidade computacional além daquela do PC ou console local, isto pode criar uma situação em que o usuário não pode prosseguir no jogo, ou tem que continuar com qualidade gráfica reduzida. Portanto, jogos em linha ou MMOG frequentemente limitam suas exigências de armazenamento e /ou comple-

xidade computacional. Adicionalmente, os mesmos frequentemente limitam a quantidade de transferências de dados durante o jogo. Jogos em linha ou MMOG também podem estreitar o mercado de usuários que podem jogar os jogos.

Além disso, usuários com conhecimento técnico estão cada vez mais fazendo engenharia reversa nas cópias locais dos jogos e modificando os jogos de forma que eles possam trapacear. As trapaças talvez tão simples como fazer uma repetição de pressionamento de botão mais rápida do que humanamente possível (por exemplo, como para atirar com um revólver rapidamente). Em jogos que suportam transações de recursos no jogo a trapaça pode alcançar um nível de sofisticação que resulta em transações fraudulentas envolvendo recursos de valor econômico real. Quando um modelo econômico em linha ou MMOG é baseado nestas transações de recursos, isto pode resultar em consequências prejudiciais substanciais aos operadores de jogos.

Os custos para desenvolver um novo jogo têm crescido conforme os PCs e consoles são capazes de produzir jogos cada vez mais sofisticados (por exemplo, com gráficos mais realistas, tal como um rastreamento em tempo real, e comportamentos mais realistas, tal como simulação física em tempo real). No início da indústria de jogos de vídeo, o desenvolvimento de jogos de vídeo era muito similar ao processo de desenvolvimento de software de aplicação, ou seja, a maior parte do custo do desenvolvimento era no desenvolvimento do software, ao contrário do desenvolvimento de gráfico, áudio e elementos comportamentais ou "recursos", tais como aqueles desenvolvidos para um filme com muitos efeitos especiais. Hoje, muitos esforços de desenvolvimento de jogos de vídeo sofisticados se assemelham mais proximamente ao desenvolvimento de filmes ricos em efeitos especiais do que ao desenvolvimento de software. Por exemplo, muitos jogos de vídeo proporcionam simulações de mundos em 3-D, e geram personagens, acessórios e ambientes cada vez mais fotorrealísticos (ou seja, gráficos de computador que parecem tão realistas quanto conjunto de imagens de ação ao vivo registrada fotograficamente). Um dos aspectos mais desafiadores do

desenvolvimento de jogos é criar uma face humana gerada por computador que seja indistinguível de uma face humana em ação ao vivo. Tecnologias de captura facial tais como Contour™ Reality Capture desenvolvida pela Mova de San Francisco, CA captura e rastreia a geometria precisa de uma face de um ator em alta resolução enquanto a mesma está em movimento. Esta tecnologia permite que uma face em 3D seja representada em um PC ou console de jogo que seja virtualmente indistinguível de uma face capturada em ação ao vivo. Capturar e representar uma face humana "foto-realista" precisa é útil em diversos aspectos. Primeiro, celebridades ou atletas altamente conhecidos são frequentemente usados em jogos de vídeo (frequentemente contratados a alto custo), e imperfeições podem ficar aparentes para o usuário, tornando a experiência de visualização confusa ou não prazerosa. Frequentemente é exigido um alto grau de detalhe para obter um alto grau de foto-realismo - exigindo a renderização de uma grande quantidade de polígonos e texturas de alta resolução, potencialmente com os polígonos e / ou texturas mudando em uma base quadro a quadro quando a face se move.

Quando um cenário com grande quantidade de polígonos com texturas detalhadas muda rapidamente, o PC ou console de jogo que suporta o jogo pode não ter suficiente RAM para armazenar dados de polígonos e textura suficientes para a quantidade requerida de quadros de animação gerados no segmento do jogo. Adicionalmente, o único controlador ótico ou único controlador de disco tipicamente disponível em um PC ou console de jogo é usualmente muito mais lento do que a RAM, e tipicamente não pode manter-se com a taxa de dados máxima que a GPU pode aceitar na renderização de polígonos e texturas. Os jogos atuais tipicamente carregam a maior parte dos polígonos e texturas dentro da RAM, o que significa que um dado cenário é grandemente limitado em complexidade e duração pela capacidade da RAM. No caso de animação facial, por exemplo, isto pode limitar um PC ou uma console de jogo a ou uma face de resolução baixa que não é foto-realista ou a uma face foto-realista que somente pode ser animada por uma quantidade limitada de quadros, antes de o jogo pausar, e

carregar polígonos e texturas (e outros dados) para mais quadros.

Assistir a uma barra de progresso se mover lentamente pela tela quando um PC ou console exibe uma mensagem similar a "Carregando..." é aceito como um problema inerente pelos usuários atuais de jogos de vídeo complexos. O atraso enquanto o próximo cenário carrega do disco ("disco" neste documento, a menos que qualificado de outra forma, refere-se à mídia ótica ou magnética não volátil, bem como a mídia não disco tal como uma memória "Flash" de semicondutor) pode demorar diversos segundos ou mesmo diversos minutos. Isto é uma perda de tempo e pode ser muito frustrante para quem está jogando. Como discutido previamente, grande parte ou todo o atraso pode ser devido ao tempo para carregar polígono, texturas e outros dados de um disco, mas também pode ser o caso de que parte do tempo é gasto enquanto o processador e / ou GPU no PC ou console prepara dados para o cenário. Por exemplo, um jogo de vídeo de futebol pode permitir que os jogadores escolham entre uma grande quantidade de jogadores, times, estádios e condições climáticas. Assim, dependendo de qual combinação particular é escolhida, diferentes polígonos, texturas e outros dados (coletivamente "objetos") podem ser requeridos para o cenário (por exemplo, times diferentes tem cores e padrões diferentes em seus uniformes). Pode ser possível enumerar muitas ou todas as várias permutações e pré-computar muitos ou todos os objetos antecipadamente e armazenar os objetos no disco usado para armazenar o jogo. Mas se a quantidade de permutações é grande, a quantidade de armazenamento requerida para todos os objetos pode ser muito grande para caber no disco (ou muito pouco prático ser carregada). Portanto, os sistemas de PC e console existentes são tipicamente são restritos tanto na complexidade como duração do jogo de dados cenários e sofrem com longos tempos de carregamento para cenários complexos.

Outra limitação significativa com sistemas de jogos de vídeo e sistemas de software de aplicação da técnica anterior é que os mesmos estão cada vez mais usando grandes bancos de dados, por exemplo, de objetos 3D tais como polígonos e texturas, que precisam ser carregados

para o PC ou console de jogo para processamento. Como discutido acima, estes bancos de dados podem demorar um longo tempo para carregar quando armazenados localmente em um disco. Tempo de carregamento, entretanto, é usualmente muito mais severo se o banco de dados é armazenado em uma localização remota e é acessado através da Internet. Em uma situação como esta isto pode levar minutos, horas ou mesmo dias para baixar um grande banco de dados. Adicionalmente, estes bancos de dados são frequentemente criados com grande custo (por exemplo, um modelo 3D de uma veleiro com mastros altos para uso em um jogo, filme, ou documentário histórico) e são destinados a venda para o usuário final local. Entretanto, o banco de dados tem o risco de ser pirateado uma vez que o mesmo tenha sido baixado para o usuário local. Em muitos casos, um usuário quer baixar um banco de dados simplesmente para o propósito de avaliá-lo para ver se o mesmo se adéqua as necessidades do usuário (por exemplo, se uma roupa 3D para um personagem de jogo tem uma aparência ou aspecto satisfatório quando o usuário realiza um movimento particular). Um tempo de carregamento longo pode ser desencorajante para o usuário avaliar o banco de dados 3D antes de decidir fazer uma compra.

Problemas similares ocorrem em MMOGs, particularmente com jogos que permitem que os usuários utilizem personagens cada vez mais personalizados. Para um PC ou console de jogo exibir um personagem o mesmo precisa ter acesso ao banco de dados de geometria 3D (polígonos, texturas, etc.) bem como aos comportamentos (por exemplo, se o personagem tem um escudo, se o escudo é forte o bastante para desviar uma lança ou não t) para aquele. Tipicamente, quando um MMOG é jogado pela primeira vez por um usuário, uma grande quantidade de bancos de dados para personagens já estão disponíveis com a cópia inicial do jogo, que está disponível localmente no disco ótico do jogo ou baixado para um disco. Mas, conforme o jogo progride, se o usuário encontra um personagem ou objeto cujo banco de dados não está disponível localmente (por exemplo, se outro usuário tiver criado um personagem personalizado), antes que aquele personagem ou objeto possam ser exibidos, seu banco de dados tem que

ser baixado. Isto pode resultar em um grande atraso do jogo.

Dada à sofisticação e complexidade dos jogos de vídeo, outro desafio para os desenvolvedores e publicadores de jogos de vídeo com as consoles de jogos da técnica anterior, é que o mesmo frequentemente leva 2 a 3 anos para desenvolver um jogo de vídeo ao custo de dezenas de milhões de dólares. Dado que novas plataformas de console de jogos de vídeo são introduzidas a uma taxa de aproximadamente uma vez a cada cinco anos, os desenvolvedores de jogos precisam iniciar o trabalho de desenvolvimento naqueles jogos anos antes da liberação da nova console de jogos a fim de ter os jogos de vídeo disponíveis concorrentemente quando a nova plataforma for liberada. Diversas consoles de fabricantes concorrentes são algumas vezes liberadas quase ao mesmo tempo (por exemplo, dentro de um ano ou dois uma da outra), mas o que resta para ser visto é a popularidade de cada console, por exemplo, qual console produzirá as maiores vendas de software de jogos de vídeo. Por exemplo, em um recente ciclo de consoles, o Microsoft XBox 360, o Sony Playstation 3, e o Nintendo Wii foram programados para serem introduzidos aproximadamente na mesma janela de tempo. Mas anos antes da introdução os desenvolvedores de jogos tiveram que essencialmente que "fazer suas apostas" em quais plataformas de console teriam mais sucesso do que outras e destinar seus recursos de desenvolvimento adequadamente. Companhias de produção de filmes também têm que dividir seus recursos de produção limitados baseado no que eles estimam ser o provável sucesso de um filme muito antecipadamente a liberação do filme. Dado o nível crescente de investimentos exigidos para jogos de vídeo, a produção de jogos está cada vez mais ficando semelhante à produção de filmes, e as companhias de produção de jogos rotineiramente dedicam seus recursos de produção baseados em suas estimativas de sucesso futuro de um jogo de vídeo particular. Mas, diferente das companhias de filmes, esta aposta não é simplesmente baseada no sucesso da própria produção; em vez disso, é pressuposta no sucesso que a console de jogo em que o jogo é destinado a rodar. A liberação do jogo em múltiplas consoles de uma vez pode mitigar o risco, mas este esforço

adicional aumenta o custo, e frequentemente atrasa a liberação real do jogo.

Software aplicativo e ambientes de usuário em PCs estão se tornando mais intensivos, dinâmicos e interativos computacionalmente, não apenas para torná-los mais atrativos visualmente para os usuários, mas também para torná-los mais úteis e intuitivos. Por exemplo, tanto o novo sistema operacional Windows Vista™ e sucessivas versões do sistema operacional Macintosh® incorporam efeitos de animação visual. Ferramentas gráficas avançadas tais como MayaTM da Autodesk, Inc., proporcionam recursos de renderização e animação em 3D muito sofisticados que pressiona os limites das CPUS e GPUs do estado da técnica. Entretanto, as exigências computacionais destas novas ferramentas cria uma quantidade de questões técnicas para usuários e desenvolvedores de software destes produtos.

Uma vez que a exibição visual de um sistema operacional (SO) tem que trabalhar em uma ampla variedade de classes de computadores -- incluindo computadores de gerações anteriores não mais vendidos, mas que ainda são atualizáveis com o novo SO – as exigências gráficas do SO são limitadas em um alto grau por um mínimo denominador comum dos computadores para os quais o SO é direcionado, o que tipicamente inclui computadores que não incluem uma GPU. Isto limita severamente a configuração gráfica do SO. Além disso, computadores portáteis alimentados por bateria (por exemplo, computadores de colo(laptops)) limitam a configuração de exibição visual uma vez que alta atividade computacional em uma CPU ou GPU tipicamente resulta em maior consumo de energia a menor vida da bateria. Computadores portáteis tipicamente incluem softwares que reduzem atividade do processador automaticamente para reduzir o consumo de energia quando o processador não é utilizado. Em alguns modelos de computadores o usuário pode reduzir a atividade do processador manualmente. Por exemplo, o computador de colo VGN-SZ280P da Sony contém um comutador chamado “Stamina” em um lado (para baixa performance, mais vida de bateria) e “Speed” do outro lado (para alta performance, menos vida de bateria). Um OS rodando em um computador portátil precisa ser capaz de

funcionar adequadamente mesmo no caso de o computador estar rodando com uma fração de sua capacidade de performance de pico. Portanto, a performance gráfica de OS frequentemente permanece muito abaixo da capacidade computacional disponível no estado da técnica.

Aplicações de alta qualidade computacionalmente intensivas como Maya são frequentemente vendidas com a expectativa de que as mesmas serão usadas em PCs de alta performance. Isto tipicamente estabelece um mínimo denominador comum de exigência de performance muito mais alto, mais caro e menos portátil. Como consequência, estas aplicações tem um público alvo muito mais limitado do que um SO de propósito geral (ou aplicação de produtividade de propósito geral, tal como Microsoft Office) e tipicamente vende volume muito menor do que software de SO ou software aplicativo de propósito geral. A audiência potencial é limitada adicionalmente porque muitas vezes é difícil para um futuro usuário testar antecipadamente estas aplicações computacionalmente intensivas. Por exemplo, supondo que um estudante quer aprender como usar o Maya ou um comprador potencial que já tem conhecimento sobre estas aplicações quer testar o Maya antes de fazer o investimento na compra (o que também pode envolver a compra de um computador de alta qualidade capaz de rodar o Maya). Embora tanto o estudante como o comprador potencial possam baixar, ou obter uma cópia de mídia física de uma versão de demonstração do Maya, se eles não tiverem um computador capaz de rodar o Maya em todo o seu potencial (por exemplo, manipulando um cenário 3D complexo), então eles serão incapazes de fazer uma avaliação completa do produto. Isto limita substancialmente a audiência para estas aplicações de alta qualidade. Isto também contribui para um alto preço de venda uma vez que o custo de desenvolvimento é amortizado através de uma quantidade muito menor de compras do que aquelas de uma aplicação de propósito geral.

Aplicações de alto preço também criam mais incentivo para indivíduos e empresas usarem cópias piratas do software aplicativo. Como resultado o software aplicativo de alta qualidade sofre de pirataria excessiva, a despeito dos significativos esforços feitos pelos publicadores de tais

softwares para mitigar esta pirataria através de várias técnicas. Ainda, mesmo quando usando aplicações de alta qualidade pirateadas, os usuários não podem remediar a necessidade de investimento em PCs caros do estado da técnica para rodar as cópias piratas. Assim, embora eles possam obter o uso de um software aplicativo por uma fração de seu preço de venda real, ainda é requerido que os usuários de software pirateado comprem ou obtenham um PC caro a fim de utilizar completamente a aplicação.

O mesmo é verdade para usuários jogos de vídeo de alta performance pirateados. Embora piratas possam ter os jogos por uma fração de seu preço real ainda é requerido que eles comprem hardware de computação caro (por exemplo, um PC equipado com uma GPU, ou uma console de jogo de vídeo de alta qualidade tal como XBox 360) necessário para jogar o jogo adequadamente. Dado que jogos de vídeo são tipicamente um passatempo para os consumidores o custo adicional para um sistema de jogo de vídeo de alta qualidade pode ser proibitivo. Esta situação é pior em países (por exemplo, China) onde a renda média anual dos trabalhadores atualmente é muito baixa relativa àquela dos Estados Unidos da América. Como resultado, uma porcentagem muito menor da população possui um sistema de jogo de vídeo system de alta qualidade ou um PC de alta qualidade. Nestes países, "Internet cafés", nos quais os usuários pagam uma taxa para usar um computador conectado à Internet, são muito comuns. Frequentemente, estes Internet cafés têm modelos mais velhos ou PCs básicos sem características de alta performance, tal como uma GPU, o que poderia de outra forma permitir que os jogadores jogassem jogos de vídeo de computação intensiva. Este é um fator chave no sucesso dos jogos que rodam em PCs básicos, tal como o "World of Warcraft" da Vivendi que é um grande sucesso na China, e comumente jogado em Internet cafés naquele país. Ao contrário, é muito menos provável que um jogo computacionalmente intensivo, tal como o "Second Life" possa ser jogado em um PC instalado em um Internet café chinês. Estes jogos são virtualmente inacessíveis para usuários que apenas têm acesso a PCs de baixa performance em Internet cafés.

Também existem barreiras para usuários que estão considerando a compra de um jogo de vídeo gostariam de primeiro tentar uma versão de demonstração do jogo baixando uma demonstração através da Internet para sua residência. Uma demonstração de jogo de vídeo é frequentemente uma versão de pleno direito do jogo com algumas características não habilitadas, ou com limites determinados na quantidade a ser jogada. Isto pode envolver um longo processo (as vezes horas) para baixar gigabytes de dados antes que o jogo possa ser instalado e executado ou em um PC ou em uma console. No caso de um PC, Isto também envolve solucionar quais acionadores especiais são necessários (por exemplo, acionadores DirectX ou OpenGL) para o jogo, baixar a versão correta, instalá-las, e então determinar se o PC é capaz de jogar o jogo. Esta última etapa pode envolver determinar se o PC tem capacidade de processamento suficiente (CPU e GPU), RAM suficiente, e um OS compatível (por exemplo, alguns jogos rodam no Windows XP, mas não no Vista). Portanto, após um longo processo de tentativa para rodar uma demonstração de jogo de vídeo demo, o usuário pode finalmente descobrir que a demonstração do jogo de vídeo não é possível de ser jogada, dada a configuração do PC do usuário. Pior, uma vez que o usuário tenha baixado novos acionadores a fim de testar a demonstração, estes acionadores podem ser incompatíveis com outros jogos ou aplicações que o usuário usa regularmente no PC, portanto a instalação de uma demonstração pode tornar inoperáveis jogos ou aplicações operáveis previamente. Estas barreiras não são apenas frustrantes para o usuário, mas as mesmas criam barreiras para os publicadores de software de jogo de vídeo e desenvolvedores de jogos de vídeo comercializarem seus jogos.

Outro problema que resulta em ineficiência econômica tem a ver com o fato de que o dado PC ou console de jogo é usualmente projetado para acomodar um certo nível de exigência de performance para aplicações e / ou jogos. Por exemplo, alguns PCs têm mais ou menos RAM, CPUs mais rápidas ou mais lentas, e GPUs mais rápidas ou mais lentas, se é que têm uma GPU. Alguns jogos ou aplicações tiram vantagem de toda a potência

computacional de um dado PC ou console, enquanto outros jogos ou aplicações não. Se uma aplicação ou jogo de escolha do usuário fica aquém da capacidade de pico de performance do PC ou console local, então o usuário pode ter desperdiçado dinheiro no PC ou console para características não utilizadas. No caso de uma console, o fabricante da console pode ter pago mais do que o necessário para subsidiar o custo da console.

Outro problema que existe na compra e venda e satisfação de jogos de vídeo envolve permitir que um usuário assista outros jogando os jogos antes de o usuário arriscar a compra daquele jogo. Existem algumas abordagens da técnica anterior para a gravação de jogos de vídeo para reprisar em um momento posterior. Por exemplo, a Patente US de No. 5.558.339 ensina a gravar informação de estado de jogo, incluindo ações do controlador do jogo, durante a execução do jogo no computador cliente do jogo de vídeo (pertencente ao mesmo usuário ou diferente). Esta informação de estado pode ser usada em um momento posterior para reprisar alguma ou toda a ação do jogo em um computador cliente do jogo de vídeo (por exemplo, PC ou console). Uma questão significativa com esta abordagem é que para um usuário visualizar o jogo gravado, o usuário tem que possuir um computador cliente do jogo de vídeo capaz de jogar o jogo e tem que ter a aplicação do jogo de vídeo rodando naquele computador, de modo que a execução do jogo é idêntica quando o estado do jogo gravado é reprisado. Além disso, a aplicação de jogo de vídeo tem que ser escrita de uma tal forma que não exista possibilidade de diferença de execução entre o jogo gravado e o jogo reproduzido.

Por exemplo, os gráficos de jogo são geralmente computados em uma base quadro a quadro. Para muitos jogos,a lógica do jogo algumas vezes pode levar mais ou menos do que um tempo de quadro para computar os gráficos exibidos para o próximo quadro, dependendo de se a cena é particularmente complexa, ou se existem outros atrasos que desaceleram a execução (por exemplo, em um PC, pode estar rodando outro processo que tire ciclos de CPU das aplicações de jogos). Em um jogo como este, um quadro "limite" que é computado e pouco menos do que um tempo de qua-

dro (digamos uns poucos ciclos de CPU a menos) pode ocorrer eventualmente. Quando aquele cenário é computado novamente usando a informação exata do mesmo estado do jogo, isto pode facilmente tomar alguns ciclos de CPU a mais do que um tempo de quadro (por exemplo, se um barramento interno de CPU está ligeiramente fora de fase com o barramento de DRAM e o mesmo introduz uns poucos tempos de ciclo de CPU de atraso, mesmo se não existir grande atraso de outro processo tomando milissegundos de tempo de CPU do processamento do jogo). Portanto, quando o jogo é reproduzido o quadro é calculado em dois tempos de quadro em vez de um único tempo de quadro. Alguns comportamentos são baseados em como geralmente o jogo calcula um novo quadro (por exemplo, quando o jogo busca a entrada dos controladores de jogo. Enquanto o jogo é jogado, esta discrepância na referência de tempo para comportamentos diferentes não impacta a jogabilidade, mas isto pode resultar no jogo reproduzido produzir um resultado diferente. Por exemplo, se uma balística de bola de basquete é calculada a uma taxa de 60 fps estável, mas a entrada do controlador de jogo é amostrada baseada na taxa de quadros computados, a taxa de quadros computados pode ser de 53 fps quando o jogo foi gravado, mas 52 fps quando o jogo é reprisado, o que pode fazer uma diferença entre se a bola de basquete é bloqueada no trajeto para dentro da cesta ou não, resultando em uma saída diferente. Portanto, usar o estado do jogo para gravar jogos de vídeo exige um projeto de software de jogo muito cuidadoso para garantir que a reprise, usando a informação de estado de jogo, produza exatamente a mesma saída.

Outra abordagem da técnica anterior para gravar jogos de vídeo é simplesmente gravar a saída de vídeo de um PC ou sistema de jogo de vídeo (por exemplo, para um gravador de VCR, DVD, ou para uma placa de captura de vídeo em um PC). O vídeo pode então ser retornado e reprisado, ou alternativamente, o vídeo gravado carregado para a Internet, tipicamente após ser comprimido. Uma desvantagem desta abordagem é que quando uma sequência de jogo 3D é reproduzida, o usuário é limitado a ver a sequência do único ponto de vista em que a sequência foi gravada. Em

outras palavras, o usuário não pode mudar o ponto de vista do cenário.

Adicionalmente, quando vídeo comprimido de uma sequência de jogo gravada jogada em um PC residencial ou console de jogo é disponibilizada para outros usuários na Internet, mesmo se o vídeo for comprimido em tempo real, pode ser impossível carregar o vídeo comprimido em tempo real para a Internet. A razão é porque muitas residências no mundo conectadas a Internet têm conexões de banda larga altamente assimétricas (por exemplo, DSL e modems a cabo tipicamente tem uma banda largura de banda de fluxo de descida muito mais alta do que a largura de banda de fluxo de subida). Sequências de vídeo de alta resolução comprimidas frequentemente têm larguras de banda mais altas do que a capacidade de largura de banda de fluxo de subida da rede, tornando impossível o carregamento em tempo real. Portanto, haveria um atraso significativo após a sequência de jogo ser jogada (Talvez minutos ou mesmo horas) antes que outro usuário na Internet fosse capaz de visualizar o jogo. Embora este atraso seja tolerável em certas situações (por exemplo, para assistir o desempenho de um jogador que ocorreu em um momento anterior), o mesmo elimina a possibilidade de assistir a um jogo ao vivo (por exemplo, um torneio de basquete jogado por jogadores campeões) ou com um recurso de "reapresentação instantânea" quando o jogo é jogado ao vivo.

Outra abordagem da técnica anterior permite que um espectador com um receptor de televisão assista a jogos ao vivo, mas apenas sob o controle da equipe de produção da televisão. Alguns canais de televisão tanto nos EUA como em outros países fornecem canais de visualização de jogos de vídeo onde o público de espectadores de televisão pode assistir certos usuários de jogos de vídeo (por exemplo, jogadores de alta classificação jogando torneios) em canais de jogos de vídeo. Isto é realizado tendo a saída de sistemas de jogo de vídeo (PCs e / ou consoles) transmitida dentro do equipamento de distribuição e processamento para o canal de televisão. Isto não é diferente de quando o canal de televisão está difundindo um jogo de basquete ao vivo no qual diversas câmeras fornecem transmissões ao vivo de diferentes ângulos em volta da quadra de basquete. O canal

de televisão então é capaz de usar seu equipamento de processamento e efeitos de áudio / vídeo para manipular a saída de vários sistemas de jogos de vídeo. Por exemplo, o canal de televisão pode sobrepor texto sobre o vídeo de um jogo de vídeo que indica o status de vários jogadores (exatamente como os mesmos podem sobrepor texto durante um jogo de basquete ao vivo), e o canal de televisão pode sobrepor áudio de um comentarista que pode discutir a ação que está ocorrendo durante os jogos. Adicionalmente, a saída do jogo de vídeo pode ser combinada com câmeras de gravação de vídeo dos jogadores reais dos jogos (por exemplo, mostrando suas reações de resposta ao jogo).

Um problema com esta abordagem é que estas transmissões de vídeo ao vivo tem que estar disponíveis para o equipamento de distribuição e processamento de vídeo do canal de televisão em tempo real a fim de ter a emoção de uma difusão ao vivo. Entretanto, como discutido anteriormente, frequentemente isto é impossível quando o sistema de jogo de vídeo está rodando de residência, especialmente se parte da difusão inclui vídeo ao vivo de uma câmera que captura vídeo do mundo real do jogador. Adicionalmente, em uma situação de torneio, existe uma preocupação de que um jogador em sua residência possa modificar o jogo e trapacear, como descrito previamente. Por estas razões, estas difusões de jogo de vídeo em canais de televisão são frequentemente organizadas com jogadores e sistemas de jogos de vídeo juntos em uma localização comum (por exemplo, em um estúdio de televisão ou em uma arena) onde o equipamento de produção de televisão possa receber transmissões de vídeo dos múltiplos sistemas de jogo de vídeo e potencialmente de câmeras ao vivo.

Embora estes canais de televisão de jogo de vídeo da técnica anterior possam fornecer uma apresentação muito emocionante para o público espectador de televisão que é uma experiência semelhante a um evento de esporte ao vivo, por exemplo, com os jogadores do jogo de vídeo apresentados como "atletas", tanto em termos de suas ações no mundo do jogo de vídeo world, como em termos de suas ações no mundo real, estes sistemas de jogos de vídeo são frequentemente limitados a situações onde

os jogadores estão em grande proximidade física uns aos outros. E, uma vez que canais de televisão são difundidos, cada canal de televisão difundido pode mostrar apenas um fluxo de vídeo contínuo, que é selecionado pela equipe de produção do canal de televisão. Devido a estas limitações e ao alto custo do tempo de difusão, equipamento de produção e equipes de produção, estes canais de televisão tipicamente mostram apenas jogadores de alta classificação jogando em torneios de alto nível.

Adicionalmente, um dado canal de televisão difundindo uma imagem de tela cheia de um jogo de vídeo para todo o público espectador da televisão mostra apenas um jogo de vídeo por vez. Isto limita severamente as escolhas de um espectador da televisão. Por exemplo, um espectador de televisão pode não estar interessado no(s) jogo(s) apresentados em um dado momento. Outro espectador pode estar interessado apenas em assistir o jogo jogado por um jogador particular que não é apresentado pelo canal de televisão em um dado momento. Em outros casos, um espectador pode estar interessado apenas em assistir como um jogador experiente se comporta em um nível particular em um jogo. Outros espectadores podem ainda querer controlar o ponto de vista pelo qual um jogo é visto, que é diferente daquele escolhido pelo time da produção, etc. Em resumo, um espectador de televisão pode ter uma miríade de preferências quando assiste jogos de vídeo que não são acomodadas pela difusão particular de uma rede de televisão, mesmo se diversos canais de televisão diferentes estiverem disponíveis. Por todas as razões mencionadas acima, canais de televisão de jogo de vídeo da técnica anterior têm limitações significativas na apresentação de jogos de vídeo para espectadores de televisão.

Outros problemas dos sistemas de jogos de vídeo e sistemas de software aplicativo da técnica anterior é que os mesmos são complexos, e comumente sofrem com erros, falhas e /ou comportamentos não intencionais e indesejados (coletivamente "erros"). Embora jogos e aplicativos tipicamente passem por um processo de depuração e ajuste fino (frequentemente chamado "Certificação de Qualidade de Software" ou SQA) antes da liberação, quase invariavelmente uma vez que o jogo ou aplicação é liberado para

um público amplo no campo os erros surgem. Infelizmente, é difícil para o desenvolvedor de software identificar e rastrear muitos dos erros após a liberação. Pode ser difícil para desenvolvedores de software ficarem cientes dos erros. Mesmo quando eles aprendem sobre um erro, pode haver apenas uma quantidade limitada de informação disponível para eles identificarem o que causou o erro. Por exemplo, um usuário pode ligar para m alinha de serviço de cliente para desenvolvedor do jogo e deixar uma mensagem expondo que quando joga o jogo, a tela começou a piscar, então mudou para uma cor sólida azul e o PC congelou. Isto fornece muito pouca informação útil para o time de SQA rastrear um erro. Alguns jogos ou aplicações que são conectados em linha algumas vezes fornecem mais informação em certos casos. Por exemplo, um processo "cão de guarda" pode ser usado algumas vezes para monitorar "falhas" no jogo ou aplicação. O processo cão de guarda pode coletar estatísticas sobre o status do processo do jogo ou aplicações (por exemplo, o status do uso da ilha de memória, quão longe o jogo ou aplicações têm progredido, etc.) quando o mesmo falha e em seguida carregar aquela informação para o time de SQA através da Internet. Mas em um jogo ou aplicação complexo, esta informação pode demorar muito tempo para decifrar a fim de determinar com precisão o que o usuário estava fazendo no momento da falha. Mesmo assim, pode ser impossível determinar qual sequência de eventos levou à falha.

Ainda outro problema associado com PCs e consoles de jogos é que os mesmos são muito sujeitos a problemas no serviço que incomodam muito o consumidor. Problemas de serviço também impactam o fabricante do PC ou console de jogo uma vez que eles tipicamente eles são solicitados a enviar uma caixa especial para despachar com segurança o PC ou console, e então incorrem no custo do conserto se o PC ou console está na garantia. O publicador do jogo ou software aplicativo também pode ser impactado pela perda de vendas (ou uso do serviço em linha) pelos PCs e / ou consoles ficando em estado de reparo.

A figura 1 ilustra um sistema de jogos de vídeo da técnica anterior tal como um Sony Playstation® 3, Microsoft Xbox 360®, Nintendo Wii™,

computador pessoa baseado em Windows ou Apple Macintosh. cada um destes sistemas inclui uma unidade central de processamento (CPU) para executar código de programa, tipicamente uma unidade de processamento gráfica (GPU) para executar operações gráficas avançadas, e múltiplas formas de entrada / saída (I / O) para se comunicar com dispositivos e usuários externos. Por simplicidade, estes componentes são combinados como uma única unidade 100. O sistema de jogos da técnica anterior da figura 1 também é mostrado incluindo um controlador de mídia ótica 104 (por exemplo, um controlador de DVD–ROM); um controlador de disco rígido 103 para armazenar código de programa e dados de jogo de vídeo; uma conexão de rede 105 para jogar jogos multijogadores, para descarregar jogos, correções, demonstrações ou outra mídia; uma memória de acesso randômico (RAM) 101 para armazenar código de programa sendo correntemente executado pela CPU / GPU 100; um controlador de jogo 106 para receber comandos de entrada do usuário durante o jogo; e um dispositivo de exibição 102 (por exemplo, um SDTV / HDTV ou um monitor de computador).

O sistema da técnica anterior mostrado na figura 1 sofre de diversas limitações. Primeiro, controladores óticos 104 e controladores de disco rígido 103 tendem a ter velocidades de acesso muito mais lentas quando comparados àquela da RAM 101. Quando trabalha diretamente da RAM 101, a CPU / GPU 100 pode, na prática, processar muito mais polígonos por segundo do que é possível quando o código de programa e dados são lidos diretamente a partir do controlador de disco rígido 103 ou controlador ótico 104 devido ao fato de que a RAM 101 tem geralmente uma largura de banda muito maior e não sofre com atrasos relativamente longos de busca dos mecanismos de disco. Mas apenas uma quantidade limitada de RAM é fornecida nestes sistemas da técnica anterior (por exemplo, 256 a 512Mbytes). Portanto, uma sequência "Carregando..." na qual a RAM 101 é periodicamente carregada com dados para o próximo cenário do jogo de vídeo é requerida frequentemente.

Alguns sistemas tentam sobrepor o carregamento do código de programa concorrentemente com a execução do jogo, mas isto pode ser

feito apenas quando existe uma sequência conhecida de eventos (por exemplo, se o carro está percorrendo uma estrada, a geometria para as edificações que se aproximam nos lados da estrada pode ser carregada enquanto o carro está viajando). Para mudanças complexas e / ou rápidas, este tipo de sobreposição usualmente não funciona. Por exemplo, no caso onde o usuário está no meio de uma batalha e a RAM 101 está completamente preenchida com dados que representam objetos dentro da visão naquele momento, se o usuário move a visão rapidamente para à esquerda para ver objetos que não estão atualmente carregados na RAM 101, resultará em uma descontinuidade na ação uma vez que não existe tempo suficiente para carregar os novos objetos do controlador de disco rígido 103 ou mídia ótica 104 para a RAM 101.

Outro problema com o sistema da figura 1 surge devido à limitações na capacidade de armazenamento dos controladores de disco rígido 103 e mídia ótica 104. Embora dispositivos de armazenamento possam ser fabricados com uma capacidade de armazenamento relativamente grande (por exemplo, 50 gigabytes ou mais), os mesmos ainda não fornecem capacidade de armazenamento suficiente para certos cenários encontrados nos jogos de vídeo atuais. Por exemplo, como mencionado previamente, jogo de vídeo de futebol pode permitir que o usuário escolha dentre dezenas de times, jogadores e estádios por todo o mundo. Para cada time, cada jogador e cada estádio são necessários uma grande quantidade de mapas de textura e mapas de ambiente para caracterizar as superfícies 3D no mundo (por exemplo, cada time tem uma camiseta única que requer um mapa de textura único).

Uma técnica usada para endereçar este último problema é o jogo pré-computar mapas de textura e ambiente uma vez que os mesmos são selecionados pelo usuário. Isto pode envolver uma quantidade de processos computacionalmente intensivos, incluindo a descompressão de imagens, mapeamento 3D, sombreamento, organização de estruturas de dados, etc. Como resultado, pode haver um atraso para o usuário enquanto o jogo de vídeo está realizando estes cálculos. Uma forma de reduzir estes atrasos,

em princípio, é executar todas estas computações – incluindo toda permutação de time, lista de jogadores, e estádio – quando o jogo foi originalmente desenvolvido. A versão liberada do jogo deve então incluir todos estes dados pré-processados armazenados na mídia ótica 104, ou em um ou mais servidores na Internet com apenas os dados pré-processados selecionados para um dado time, lista de jogadores, seleção de estádios descarregados através da Internet para o controlador de disco rígido 103 quando o usuário faz uma seleção. Como um caso prático, entretanto, estes dados pré-carregados de todas as permutações possíveis para jogar o jogo podem facilmente atingir terabytes de dados, que é muito mais do que a capacidade dos dispositivos atuais de mídia ótica. Além disso, os dados para um dado time, lista de jogadores, seleção de estádio podem ser facilmente centenas de megabytes de dados ou mais. Com uma conexão de rede residencial de, digamos, 10Mbps, seria mais demorado descarregar estes dados através da conexão de rede 105 do que seria para computar os dados localmente.

Portanto, a arquitetura de jogo da técnica anterior mostrada na figura 1 sujeita o usuário a atrasos significativos entre transições de cenários principais de jogos complexos.

Outro problema com a abordagem da técnica anterior tal como aquela mostrada na figura 1 é que ao longo dos anos os jogos de vídeo tendem a se tornar mais avançados e exigir mais capacidade de processamento da CPU / GPU. Portanto, mesmo assumindo uma capacidade ilimitada de RAM, as exigências de hardware dos jogos de vídeo vai além do nível de pico da capacidade de processamento disponível nestes sistemas. Como resultado, é requerido que os usuários atualizem seu hardware de jogos a cada poucos anos para manter o ritmo (ou jogar jogos mais novos com níveis de qualidade mais baixo). Uma consequência da tendência de jogos de vídeo cada vez mais avançados é que as máquinas para jogar jogos de vídeo para uso doméstico são tipicamente ineficientes economicamente porque seu custo normalmente é determinado pelas exigências da maior performance que o jogo pode suportar. Por exemplo, um XBox 360 pode ser usado para jogar um jogo como "Gears of War", que demanda uma

CPU, GPU de alta performance e centenas de megabytes de RAM, ou o XBox 360 pode ser usado para jogar Pac Man, um jogo dos anos 1970s que requer apenas kilobytes de RAM e uma CPU de performance muito baixa. Na verdade, um XBox 360 tem potência computacional suficiente para hospedar muitos jogos simultâneos Pac Man.

Máquinas de jogos de vídeo ficam tipicamente desligadas a maior parte das horas de uma semana. De acordo com um estudo da Nielsen Entertainment de julho de 2006 dos jogadores ativos de 13 anos ou mais, na média, jogadores ativos gastam quatorze horas / semana jogando jogos de vídeo na console, ou apenas 12% do total de horas de uma semana. Isto significa que a console de jogo de vídeo fica ociosa 88% do tempo, o que é um uso ineficiente de um recurso caro. Isto é particularmente significativo dado que as consoles de jogos frequentemente são subsidiadas pelos fabricantes para baixar o preço de compra (com a expectativa de que o subsídio será recebido de volta pelos direitos de uso de futuras compras de software de jogos de vídeo).

Consoles de jogos de vídeo também incorrem em custos associados com quase todos dispositivos eletrônicos para consumidor. Por exemplo, a eletrônica e mecanismos precisam ser alojados em um gabinete. O fabricante precisa oferecer um serviço de garantia. O varejista que vende o sistema precisa receber uma margem em cada venda do sistema e / ou na venda do software do jogo de vídeo. Todos estes fatores adicionam custo à console de jogo de vídeo, que têm que ser subsidiados pelo fabricante, repassado para o consumidor, ou ambos.

Adicionalmente, a pirataria é um problema importante para a indústria de jogos de vídeo. Os mecanismos de segurança utilizados em virtualmente a maior parte dos sistemas de jogos de vídeo têm sido "quebrados" ao longo dos anos, resultando em uma cópia não autorizada de jogos de vídeo. Por exemplo, o sistema de segurança do Xbox 360 foi quebrado em julho de 2006 e agora os usuários são capazes de descarregar cópias ilegais em linha. Jogos que são descarregáveis (por exemplo, jogos para PC ou Mac) são particularmente vulneráveis a pirataria. Em certas regiões do

mundo onde a pirataria é fracamente policiada não existe essencialmente nenhum mercado viável para software de jogo de vídeo individual porque os usuários podem comprar cópias pirateadas tão facilmente como cópias legais por uma mínima fração do custo. Também, em muitas partes do mundo o custo de uma console de jogo é um percentual tão alto da renda que mesmo que a pirataria fosse controlada, poucas pessoas poderiam ter um sistema de jogos do estado da técnica.

Adicionalmente, o mercado de jogos usados reduz venda para a indústria de jogos de vídeo. Quando um usuário cansa de um jogo, ele pode vender o jogo para uma loja que irá revender o jogo para outros usuários. Esta prática não autorizada mas comum reduz significativamente as vendas de publicadores de jogos. De maneira similar, uma redução nas vendas da ordem de 50% ocorre comumente quando existe uma transição de plataforma em um período de poucos anos. Isto se dá porque os usuários param de comprar jogos para as plataformas antigas quando eles sabem que uma nova plataforma está para ser liberada (por exemplo, quando o Playstation 3 estava próximo a ser liberado, os usuários pararam de comprar jogos para Playstation 2). Combinados, as perdas de vendas e sustos de desenvolvimento aumentados associados as novas plataformas podem ter um impacto adverso muito significativo na lucratividade dos desenvolvedores de jogos.

Novos consoles de jogos também são muito caras. O Xbox 360, o Nintendo Wii, e o Sony Playstation 3 são todos vendidos por centenas de dólares. Sistemas de jogos de computador pessoal de alta potência podem custar até US$ 8.000. Isto representa um investimento significativo para os usuários, particularmente considerando que o hardware se torna obsoleto após uns poucos anos e o fato de que muitos sistemas são comprados para crianças.

Uma abordagem para os problemas acima referenciados é de jogos em linha nos quais o código e dados de programa de jogo são hospedados em um servidor e entregues para as máquinas clientes sob demanda como vídeo comprimido e áudio transmitidos sobre uma rede de banda larga digital. Algumas companhias tais como a G-Cluster a Finlândia (agora uma

subsidiária da SOFTBANK Broadmedia japonesa) fornece estes serviços em linha atualmente. Serviços similares de jogos têm se tornado disponíveis em redes locais, tais como aquelas de dentro de hotéis e oferecidas por provedores de televisão a cabo e DSL. Um grande problema destes sistemas é o problema de latência, ou seja, o tempo que leva para o sinal se deslocar para e do servidor, que é tipicamente localizado em uma "estação transmissora" do operador". Jogos de vídeo de ação rápida (também conhecidos como jogos de vídeo de "espasmo") requerem uma latência muito baixa entre o tempo que o usuário executa uma ação com o controlador de jogo e o tempo em que a tela de exibição é atualizada mostrando o resultado da ação do usuário. A baixa latência é necessária para que o usuário tenha a percepção de que o jogo está respondendo "instantaneamente". Os usuários podem ser satisfeitos com diferentes intervalos de latência dependendo do tipo de jogo e do nível de conhecimento do usuário. Por exemplo, 100 ms de latência podem ser toleráveis para um jogo casual lento (como gamão) ou um jogo de interpretação de papel (role playing game) de ação lenta, mas em jogos de ação rápida uma latência acima de 70 ou 80 ms pode fazer com que o usuário se saia pior no jogo, e portanto é inaceitável. Por exemplo, em um jogo que requer tempo de reação rápido existe um forte declínio na precisão se a latência cresce de 50 para 100 ms.

Quando um servidor de jogo ou aplicação é instalado em um ambiente de rede controlado próximo, ou um onde o percurso de rede para o usuário é previsível e / ou pode tolerar picos de largura de banda, é muito mais fácil controlar a latência, tanto em termos de latência máxima como em termos de consistência da latência (por exemplo, de modo que o usuário observe um movimento uniforme a partir do fluxo continuo de vídeo através da rede). Este nível de controle pode ser alcançado entre uma estação de transmissão de rede de TV a cabo para uma residência de assinante de TV a cabo ou de um escritório central de DSL para uma residência de assinante de DSL, ou em um ambiente de Rede de área Local (LAN) de m escritório comercial a partir de um servidor ou de um usuário. Também, é possível obter conexões ponto a ponto privadas de graduação especial entre empre-

sas que tenham latência e largura de banda garantidas. Mas em um sistema de jogo ou aplicação que hospeda jogos em um centro de servidores conectado a Internet em geral e então transmite vídeo comprimido para o usuário através de uma conexão de banda larga, a latência ocorre a partir de muitos fatores, resultando em limitações severas na implementação de sistemas da técnica anterior.

Em uma residência conectada em banda larga típica, um usuário pode ter um modem DSL ou a cabo para serviço de banda larga. Estes serviços de banda larga incorrem em latência de ida e volta de até 25 ms (e algumas vezes mais) entre a residência do usuário e a Internet em geral. Adicionalmente, existem latências no percurso de ida e volta que ocorrem do encaminhamento de dados através da Internet para um centro de servidores. A latência através da Internet varia baseada na rota que os dados percorrem e nos atrasos em que os mesmos incorrem quando são encaminhados. Adicionalmente aos atrasos de encaminhamento, a latência de ida e volta também ocorre devido ao deslocamento à velocidade da luz através da fibra ótica que interconecta a maior parte da Internet. Por exemplo, para cada 1600 kilômetros (1000 milhas), são somados 22 ms na latência de ida e volta devido a velocidade da luz através da fibra ótica e outras sobrecargas.

Pode ocorrer latência adicional devido à taxa de dados dos dados transmitidos através da Internet. Por exemplo, se um usuário tem um serviço DSL que é vendido como "serviço DSL de 6Mbps", na prática, provavelmente o usuário terá menos do que 5Mbps de capacidade de enlace de descida, e provavelmente verá a conexão degradar periodicamente devido a vários fatores tais como congestionamento durante horas de pico no Multiplexador de Linha de Acesso de Assinante Digital (DSLAM). Uma questão similar pode ocorrer reduzindo a taxa de dados de um modem a cabo usado para uma conexão vendida como "serviço de modem a cabo de 6Mbps" para muito menos do que, se existir um congestionamento no circuito de cabo coaxial local compartilhado através da vizinhança, ou em qualquer outro local na rede do sistema de modem a cabo. Se os pacotes de dados a uma taxa estável de 4Mbps são transmitidos em formato de sentido

único em Protocolo de Datagrama de Usuário (UDP) a partir de um centro de servidores através de tais conexões, se tudo estiver funcionando bem, os pacotes de dados passarão sem incorrer em latência adicional, mas se existir congestionamento (ou outros impedimentos) e estiverem disponíveis apenas 3,5 Mbps para transmitir dados para o usuário, então em uma situação típica ou serão descartados pacotes, resultando dados perdidos, ou pacotes irão enfileirar no ponto de congestionamento, até que possam ser enviados, deste modo introduzindo latência adicional. Pontos diferentes de congestionamento têm diferentes capacidades de enfileiramento para manter pacotes atrasados, assim em alguns casos os pacotes que não fazê-lo através do congestionamento são descartados imediatamente. Em outros casos, alguns megabits de dados são enfileirados e eventualmente enviados. Mas, em quase todos os casos, as filas em pontos de congestionamento têm limites de capacidade, e uma vez que estes limites sejam excedidos, as filas irão transbordar e pacotes serão descartados. Portanto, para evitar a ocorrência de latência adicional (ou pior, perda de pacotes), é necessário evitar exceder a capacidade da taxa de dados do servidor do jogo ou aplicação para o usuário.

A latência também corre pelo tempo requerido para comprimir vídeo no servidor e descomprimir no dispositivo cliente. A latência ocorre adicionalmente enquanto um jogo de vídeo está calculando o próximo quadro a ser exibido. Os algoritmos de compressão de vídeo atualmente disponíveis sofrem ou de altas taxas de dados ou alta latência. Por exemplo, JPEG em movimento é um algoritmo de compressão apenas intra-quadro com perdas que é caracterizado por baixa latência. Cada quadro de vídeo é comprimido independentemente de cada outro quadro de vídeo. Quando um dispositivo cliente recebe um quadro de vídeo de JPEG em movimento comprimido, o mesmo pode imediatamente descomprimir o quadro e exibi-lo, resultando em baixa latência. Mas devido a cada quadro ser comprimido separadamente, o algoritmo é incapaz de explorar similaridades entre quadros sucessivos, e como resultado algoritmos de compressão de vídeo apenas intra-quadro sofrem com taxas de dados muito altas. Por exemplo,

60 fps (quadros por segundo) de vídeo JPEG em movimento 640x480 requerem 40Mbps (megabits por segundo) ou mais de dados. Estas altas taxas de dados para tais janelas de vídeo de baixa resolução seriam proibitivamente caras em muitas aplicações de banda larga (e certamente para a maior parte das aplicações baseadas em Internet). Adicionalmente, devido a cada quadro ser comprimido independentemente, é provável que artefatos nos quadros que podem resultar das perdas de compressão apareçam em diferentes lugares em quadros sucessivos. Isto pode resultar em o que aparece para um espectador como artefatos em movimento quando o vídeo é descomprimido.

Outros algoritmos de compressão, tais como MPEG2, H.264 ou VC9 da Microsoft Corporation como são usados em configurações da técnica anterior, podem obter altas taxas de compressão, mas ao custo de alta latência. Estes algoritmos usam compressão inter-quadro bem como intra-quadro. Periodicamente, estes algoritmos executam uma compressão apenas intra-quadro de um quadro. Este quadro é conhecido como o quadro chave (tipicamente referenciado como um quadro "I"). Então, estes algoritmos tipicamente comparam o quadro I tanto com quadros anteriores como com quadros sucessivos. Em vez de comprimir os quadros anteriores e quadros sucessivos independentemente, o algoritmo determina o que foi mudado na imagem do e quadro I para os quadros anterior e sucessivo, e então armazena estas mudanças como o que são chamados quadros "B" para as mudanças que precedem o quadro I e quadros "P" para as mudanças que seguem o quadro I. Isto resulta em taxas de dados muito mais baixas do que a compressão apenas intra-quadro. Mas, vem tipicamente através de um custo de maior latência. Um quadro I é tipicamente muito maior do que um quadro B ou P (frequentemente 10 vezes maior), e como resultado, o mesmo é proporcionalmente mais demorado para transmitir a uma dada taxa.

Considerando, por exemplo, uma situação onde os quadros I são 10X o tamanho dos quadros B e P, e que existem 29 quadros B + 30 quadros P = 59 inter-quadros para cada intra-quadro I, ou 60 quadros no

total para cada "Grupo de Quadros" (GOP). Assim, a 60 fps, existe 1 60-quadro GOP a cada segundo. Supondo que um canal de transmissão tenha uma taxa de dados de aproximadamente 2Mbps. Para obter a melhor qualidade de vídeo no canal, o algoritmo de compressão deve produzir um fluxo contínuo de dados de 2Mbps, e dadas as taxas acima, isto deve resultar em 2 Megabits (Mb) / (59+10) = 30.394 bits por inter-quadro e 303.935 bits por quadro I. Quando o fluxo contínuo de vídeo comprimido é recebido pelo algoritmo de descompressão, para que o vídeo passe de forma estável, cada quadro precisa ser descomprimido e exibido a um intervalo regular (por exemplo, 60 fps). Para alcançar este resultado, se qualquer quadro é sujeito a latência todos os quadros precisam ser atrasados por pelo menos aquela latência, assim o pior caso de latência de quadro definirá a latência para todos os quadros de vídeo. Os quadros I introduzem as maiores latências de transmissão uma vez que os mesmos são maiores, e um quadro I inteiro deve ter que ser recebido antes que o quadro I possa ser descomprimido e exibido (ou qualquer inter-quadro dependente do quadro I). Dado que a taxa de dados do canal é de 2Mbps, levará 303.935/2Mb = 145 ms para transmitir um quadro I.

Um sistema de compressão de vídeo inter-quadro como descrito acima usando uma grande porcentagem de largura de banda do canal de transmissão estará sujeito a longas latências devido ao grande tamanho de um quadro I relativo ao tamanho médio de um quadro. Ou, para colocar de outra forma, embora os algoritmos de compressão inter-quadro da técnica anterior obtém uma taxa de dados média por quadro mais baixa do que os algoritmos de compressão apenas intra-quadro (por exemplo, 2Mbps vs. 40Mbps), eles ainda sofrem de uma taxa de dados de pico por quadro alta (por exemplo, 303.935 * 60 = 18,2Mbps) devido aos quadros I. Tendo em mente, entretanto que a análise acima assume que os quadros P e B são muito menores do que os quadros I. Embora isto seja verdadeiro em geral, isto não é verdadeiro para quadros com alta complexidade de imagem não correlacionados com o quadro anterior, muito movimento, ou mudanças de cenário. Nestas situações, os quadros P ou B podem se tornar tão grandes

quanto os quadros I (se um quadro P ou B se torna maior do que um quadro I, um algoritmo de compressão sofisticado irá tipicamente "forçar" um quadro I e substituir o quadro P ou B por um quadro I). Assim, os picos de taxa de dados dimensionados por quadro I podem ocorrer a qualquer momento em um fluxo contínuo de vídeo digital. Portanto, com vídeo comprimido, quando a taxa de dados de vídeo médias se aproxima da capacidade de taxa de dados dos canais de transmissão (como é frequentemente o caso, dada as altas demandas de taxa de dados para vídeo) As altas taxas de dados de pico dos quadros I ou grandes quadros P ou B resultam em uma alta latência de quadro.

Naturalmente, a discussão acima apenas caracteriza a latência do algoritmo de compressão criada por quadros B, P ou I grandes em um GOP. Se quadros B são usados , a latência será ainda maior. A razão é porque antes de um quadro B poder ser exibido, todos os quadros B após o quadro B e o quadro I têm que ser recebidos. Portanto, em uma sequência de grupo imagem (GOP) tal como BBBBBIPPPPPBBBBBIPPPPP, onde existem 5 quadros B antes de cada quadro I, o primeiro quadro B não pode ser exibido pelo descompressor de vídeo até que os quadros B e quadro I subsequentes sejam recebidos. Assim, se o vídeo está sendo transmitido a 60fps (ou seja, 16,67ms/quadro), antes de o primeiro quadro B poder ser descomprimido, cinco quadros B e o quadro I levarão 16.67 * 6 = 100ms para ser recebidos, não importa quão rápida seja a largura de banda do canal, e isto com apenas 5 quadros B. Sequências de vídeo comprimido com 30 quadros B são bastante comuns. E, em uma largura de banda de canal baixa como 2Mbps, o impacto da latência causada pelo quadro I é grandemente aditiva ao impacto devido à espera pela chegada de quadros B. Portanto, em um canal de 2Mbps com uma grande quantidade de quadros B é muito fácil exceder 500ms de latência ou mais usando tecnologia de compressão de vídeo da técnica anterior. Se não são usados quadros B (ao custo de uma menor taxa de compressão para dado nível de qualidade), não ocorre a latência de quadro B, mas a latência causada pelos tamanhos de quadro de pico, descrita acima, ainda ocorre.

O problema é muito exacerbado pela natureza de muitos jogos de vídeo. Algoritmos de compressão de vídeo que utilizam a estrutura de GOP descrita acima têm sido muito otimizados para uso com material de vídeo ao vivo ou filmes para visualização passiva. Tipicamente, a câmera (ou uma câmera real, ou uma câmera no caso de uma animação gerada por computador) e cenário são relativamente estáveis, simplesmente porque se a câmera ou cenário se movem ao redor muito aos trancos, o material de vídeo ou filme é (a) tipicamente desprazeroso para assistir e (b) se está sendo assistido, usualmente o espectador não está seguindo a ação de perto quando a câmera arranca subitamente (por exemplo, se a câmera sofre um impacto quando está filmando uma criança apagando as velas em um bolo de aniversário e repentinamente desvia para longe do bolo e volta novamente, os espectadores estão tipicamente focados na criança e no bolo, e desconsideram a breve interrupção quando a câmera se move). N caso do vídeo de uma entrevista, ou uma teleconferência de vídeo, a câmera pode ser mantida em uma posição fixa e não se mover, resultando em muito poucos picos de dados. Mas jogos de vídeo 3D de muita ação são caracterizados por movimento constante (por exemplo, considerando uma corrida em 3D, onde o quadro inteiro está em movimento rápido pela duração da corrida, ou considerando atiradores em primeira pessoa, onde a câmera virtual está constante mente se movendo aos trancos). Estes jogos de vídeo podem resultar em sequências de quadro com picos grandes e frequentes onde o usuário pode precisar ver claramente o que está acontecendo durante estes movimentos súbitos. Assim, artefatos de compressão são muito menos toleráveis em jogos de vídeo de muita ação em 3D. Portanto, a saída de vídeo de muitos jogos de vídeo, por sua natureza, produz um fluxo contínuo de vídeo comprimido com picos muito altos e frequentes.

Dado que os usuários de jogos de vídeo de ação rápida têm pouca tolerância a alta latência, e dadas todas as causas de latência acima, até hoje tem existido limitações a jogos de vídeo hospedados em servidor que transmite vídeo pela Internet. Adicionalmente, usuários de aplicações que requerem um alto grau de interatividade sofrem com limitações similares

se as aplicações são hospedadas na Internet em geral e transmitem vídeo. Estes serviços requerem uma configuração de rede na qual os servidores hospedeiros são configurados diretamente em uma estação transmissora (no caso de banda larga por cabo) ou no escritório central (no caso de Linhas de Assinante Digitais (DSL)), ou dentro de uma LAN (ou em conexões privadas altamente graduadas) em uma configuração comercial, de modo que a rota e distância do dispositivo cliente para o servidor é controlada para minimizar a latência e picos podem ser acomodados sem incorrer em latência. LANs (tipicamente classificadas em 100Mbps a 1Gbps) e linhas alugadas com largura de banda adequada tipicamente podem suportar exigências de pico de largura de banda (por exemplo, pico de largura de banda de 18Mbps é uma pequena fração da capacidade de LAN de 100Mbps).

Exigências de pico de largura de banda também podem ser acomodadas pela infra-estrutura de banda larga residencial se são feitas acomodações especiais. Por exemplo, em um sistema de TV a cabo, pode ser dada uma largura de banda dedicada de tráfego de vídeo que trate os picos, tais como grandes quadros I. E em um sistema DSL, pode ser fornecido um modem DSL de maior velocidade, que permita picos altos, ou pode ser fornecida uma conexão de graduação especial que possa tratar taxas de dados mais altas. Mas, modem de cabo e infra-estrutura DSL convencionais conectados a Internet têm muito menos tolerância para exigência de largura de banda de pico para vídeo comprimido. Assim, serviços em linha que hospedam jogos de vídeo o aplicações em centros de servidores a longa distância dos dispositivos de clientes, e então transmitem saída de fluxo contínuo de vídeo comprimido output pela Internet através de conexões de banda larga residenciais convencionais sofrem de limitações de latência e largura de banda de pico significativas – particularmente com respeito a jogos e aplicações que requerem baixa latência (por exemplo, tiro em primeira pessoa e outros jogos de ação interativos multiusuários, ou applications que requerem um tempo de resposta rápido).

BREVE DESCRIÇÃO DOS DESENHOS

A presente invenção será mais entendida mais completamente a partir da descrição detalhada a seguir e a partir dos desenhos em anexo, que entretanto, não devem ser tomados para limitar o objeto revelado às modalidades específicas mostradas, mas apenas para explicação e entendimento.

A figura 1 ilustra uma arquitetura de um sistema de jogos da técnica anterior.

As figuraS 2a e b ilustram uma arquitetura de sistema de alto nível de acordo com uma modalidade.

A figura 3 ilustra taxas de dados real, classificada e requerida para comunicação entre um cliente e um servidor.

A figura 4a ilustra um serviço de hospedagem e um cliente empregado de acordo com uma modalidade.

A figura 4b ilustra latências exemplificativas associadas com comunicação entre um cliente e serviço de hospedagem.

A figura 4c ilustra um dispositivo cliente de acordo com uma modalidade.

A figura 4d ilustra um dispositivo cliente de acordo com outra modalidade.

figura 4e ilustra um diagrama de blocos exemplificativo do dispositivo cliente na figura 4c.

figura 4f ilustra um diagrama de blocos exemplificativo do dispositivo cliente na figura 4d.

A figura 5 ilustra uma forma exemplificativa de compressão de vídeo que pode ser empregada de acordo com uma modalidade.

A figura 6a ilustra uma forma exemplificativa de compressão de vídeo que pode ser empregada em outra modalidade.

A figura 6b ilustra picos de taxa de dados associados com transmissão de uma sequência de vídeo de baixa complexidade e baixa ação,

A figura 6c picos de taxa de dados associados com transmissão

de uma sequência de vídeo de alta complexidade e alta ação.

As figuraS 7a e b ilustram técnicas de compressão de vídeo exemplificativas empregadas em uma modalidade.

A figura 8 ilustra técnicas de compressão de vídeo exemplificativas adicionais empregadas em uma modalidade.

As figuraS 9a a 9c ilustram técnicas de processamento de taxa de quadro empregados em uma modalidade da invenção.

As figuraS 10a e b ilustram uma modalidade que empacota eficientemente recortes de imagem dentro de pacotes.

As figuraS 11a a 11d ilustram modalidades que empregam técnicas de correção de erro de encaminhamento.

A figura 12 ilustra uma modalidade que usa unidades de processamento multinúcleo para compressão.

As figuraS 13a e b ilustram posicionamento geográfico e comunicação entre serviços de hospedagem de acordo com várias modalidades.

A figura 14 ilustra latências exemplificativas associadas com communication entre um cliente e um serviço de hospedagem.

A figura 15 ilustra um exemplo de arquitetura de centro de servidores de serviço de hospedagem.

A figura 16 ilustra uma fotografia de exemplo de uma modalidade de uma interface de usuário que inclui uma pluralidade de janelas de vídeo ao vivo.

figura 17 ilustra a interface de usuário da figura 16 seguindo a seleção de uma janela de vídeo particular.

figura 18 ilustra a interface de usuário da figura 17 seguindo a aproximação da janela de vídeo particular para o tamanho inteiro da tela.

A figura 19 ilustra um exemplo de dados de vídeo de usuário colaborativo na tela de um jogo multijogador.

A figura 20 ilustra um exemplo de página de usuário para um jogador em um serviço de hospedagem.

A figura 21 ilustra um exemplo de propaganda interativa em 3D.

A figura 22 ilustra um exemplo de sequência de etapas para

produzir uma imagem foto-realista que tenha uma superfície texturizada da captura de superfície de uma performance ao vivo.

A figura 23 ilustra um exemplo de página de interface de usuário que permite a seleção de conteúdo de mídia linear.

A figura 24 é um gráfico que ilustra a quantidade de tempo que passa antes de da página de web versus a velocidade de conexão.

As figuraS 25a e b ilustram modalidades da invenção que empregam um canal de realimentação de informação do dispositivo cliente para o serviço de hospedagem.

As figuraS 26a e b ilustram uma modalidade na qual codifica recortes / quadros baseado no último recorte / quadro conhecido a ter sido recebido com sucesso.

As figuraS 27a e b ilustram uma modalidade na qual o estado de um jogo ou aplicação é portado de um primeiro serviço de hospedagem ou servidor para um segundo serviço ou servidor de hospedagem.

A figura 28 ilustra uma modalidade o estado de um jogo ou aplicação é portado usando dados de diferença.

A figura 29 ilustra uma modalidade da invenção que emprega um decodificador temporário no dispositivo cliente.

A figura 30 ilustra como "recortes I" são intercalados através de "quadros R" de acordo com uma modalidade da invenção.

As figuraS 31a a h ilustram modalidades da invenção que geram um fluxo contínuo de vídeo e / ou um ou mais fluxos contínuos de HQ.

DESCRIÇÃO DE MODALIDADES DE EXEMPLO

Na descrição a seguir são demonstrados detalhes específicos, tais como tipos de dispositivos, configurações de sistema, métodos de comunicação, etc., a fim de fornecer um entendimento completo da presente invenção. Entretanto, pessoas que tem conhecimentos básicos nas técnicas relevantes avaliarão que estes detalhes específicos podem não ser necessários para praticar as modalidades descritas.

As figuras 2a e b fornecem uma arquitetura de alto nível de duas modalidades nas quais jogos de vídeo e aplicações de software são hospe-

dadas por um serviço de hospedagem 210 e acessadas por dispositivos clientes 205 nas premissas de usuário 211 (deve ser observado que as "premissas de usuário" significa o lugar onde o usuário está localizado, incluindo locais externos se estiver usando um dispositivo móvel) através da Internet 206 (ou outra rede pública ou privada) sob um serviço de assinatura. Os dispositivos clientes 205 podem ser computadores de propósito geral tais como Microsoft Windows ou PCs baseados em Linux ou computadores Macintosh da Apple, Inc. com uma conexão por cabo ou sem fio à Internet com um dispositivo de exibição interno ou externo 222, ou os mesmos podem ser dispositivos clientes dedicados tais como um decodificador (com uma conexão por cabo ou sem fio à Internet) que fornece saída de vídeo e áudio para um conjunto de monitor ou TV 222, ou os mesmos podem ser dispositivos móveis, presumidamente com uma conexão sem fio à Internet.

Qualquer um destes dispositivos pode ter seus próprios dispositivos de entrada (por exemplo, teclados, botões, telas sensíveis ao toque, dispositivo apontador ou dispositivos sensíveis à inércia, câmeras de captura de vídeo e / ou câmeras de rastreamento de movimento, etc.), ou os mesmos podem usar dispositivos de entrada externos 221 (por exemplo, teclados, ratos (mice), controladores de jogo, dispositivos sensíveis à inércia, câmeras de captura de vídeo e / ou câmeras de rastreamento de movimento, etc.), conectados por cabos ou sem fio. como descrito em mais detalhes abaixo, o serviço de hospedagem 210 servidores de vários níveis de performance, incluindo aqueles com CPU / GPU de alta capacidade de processamento. Durante um jogo ou uso de uma aplicação no serviço de hospedagem 210, dispositivo cliente 205 residencial ou de escritório recebe entradas de teclado e / ou controlador do usuário, e em seguida transmite a entrada do controlador através da Internet 206 para o serviço de hospedagem 210 que executa o código do programa de jogo em resposta e gera quadros sucessivos de saída de vídeo (uma sequência de imagens de vídeo) para o jogo ou software aplicativo (por exemplo, se o usuário pressiona um botão que dirige um personagem na tela para se mover para a direita, o programa de jogo então cria uma sequência de imagens de vídeo que mostra o

personagem se movendo para a direita). Esta sequência de imagens de vídeo é então comprimida usando um compressor de vídeo de baixa latência, e o serviço de hospedagem 210 então transmite o fluxo contínuo de vídeo de baixa latência através da Internet 206. o dispositivo cliente residencial ou de escritório então decodifica o fluxo contínuo de vídeo comprimido e entrega as imagens de vídeo descomprimido em um monitor ou TV. Consequentemente, as exigências de hardware de computação e gráfico do dispositivo cliente 205 são significativamente reduzidas. O cliente 205 precisa apenas ter poder de processamento para encaminhar a entrada do teclado / controlador para a Internet 206 e decodifica e descomprime um fluxo contínuo de vídeo comprimido recebido da Internet 206, o que virtualmente qualquer computador pessoal é capaz de fazer hoje em software em sua CPU (por exemplo, uma CPU Core Duo da Intel Corporation rodando a aproximadamente 2GHz é capaz de descomprimir HDTV de 720p codificado usando compressores tais como H.264 e Windows Media VC9). E, no caso de quaisquer circuitos integrados (chips) dedicados de dispositivos clientes também poderem executar descompressão de vídeo para estes padrões em tempo real a um custo muito menor e com muito menos consumo de energia do que uma CPU de propósito geral tal como seria requerido por um PC moderno. Notavelmente, para executar a função de encaminhar a entrada do controlador e descompressão de vídeo,os dispositivos clientes residenciais 205 não requerem quaisquer unidades de processamento gráfico especializadas (GPUs), controlador ótico ou controlador de discos rígidos, tal como o sistema de jogo de vídeo da técnica anterior mostrado na figura 1.

Conforme os jogos e softwares aplicativos se tornam mais complexos e fotorrealísticos, os mesmos exigirão CPUs e GPUs de maior performance, mais RAM, e controladores de disco maiores e mais rápidos, e o poder de computação no serviço de hospedagem 210 pode ser continuamente atualizado, mas não é requerido que o usuário final atualize a plataforma cliente residencial ou do escritório 205 uma vez que suas exigências de processamento permanecerão constantes para uma resolução de exibição e taxa de quadros com um dado o algoritmo de descompressão de

vídeo. Portanto, as limitações de hardware e problemas de compatibilidade vistos atualmente não existem no sistema ilustrado nas figuras 2a e b.

Adicionalmente, devido ao jogo e software aplicativo executarem apenas em servidores no serviço de hospedagem 210, nunca existe uma cópia do jogo ou software aplicativo (seja na forma de mídia ótica, ou como software descarregado) na residência ou escritório do usuário ("escritório" como usado neste documento a menos que qualificado de outra forma deve incluir qualquer configuração não residencial, incluindo, por exemplo, salas escolares). Isto mitiga significativamente a probabilidade de um jogo ou software aplicativo ser copiado ilegalmente (pirateado), bem como mitiga a probabilidade de que um banco de dados valioso que pode ser usado por um jogo ou softwares aplicativos seja pirateado. Na verdade, se são requeridos servidores especializados (por exemplo, requer equipamento muito caro grande e ruidoso) para jogar o jogo ou software aplicativo que não são práticos para uso residencial ou de escritório, então mesmo se uma cópia pirateada do jogo ou software aplicativo for obtida, o mesmo não deve ser operável na residência ou escritório.

Em uma modalidade, o serviço de hospedagem 210 fornece ferramentas de desenvolvimento de software para os desenvolvedores de jogo ou software aplicativo (que refere-se geralmente a companhias de desenvolvimento de software, estúdios de jogo ou filme, ou publicadores de jogo ou softwares aplicativos) 220 que projetam jogos de vídeo de modo que os mesmos possam projetar jogos capazes de ser executados no serviço de hospedagem 210. Estas ferramentas permitem que desenvolvedores explorem características do serviço de hospedagem que normalmente não estão disponíveis em um PC ou console de jogo autônomos (por exemplo, acesso rápido a bancos de dados muito grandes de geometria complexa ("geometria" a menos que qualificada de outra forma deve ser usada neste documento para se referir a comportamentos de polígonos, texturas, montagem, iluminação e outros componentes e parâmetros que definem estruturas de dados 3D)).

São possíveis diferentes modelos de negócios sob esta arquite-

tura. Em um modelo, o serviço de hospedagem 210 coleta uma taxa de assinatura do usuário final e paga um direito de uso para os desenvolvedores 220, como mostrado na figura 2a. Em uma implementação alternativa, mostrada na figura 2b, os desenvolvedores 220 coletam uma taxe de assinatura diretamente do usuário e pagam o serviço de hospedagem 210 para hospedar o conteúdo do jogo ou aplicação. Estes princípios subjacentes não são limitados a qualquer modelo de negócio para fornecer jogos em linha ou hospedagem de aplicações.

Características de Vídeo Comprimido

Como discutido anteriormente, um problema significativo com fornecimento de serviços de jogos de vídeo ou serviços de software de aplicação em linha é o da latência. Uma latência de 70 a 80 ms (do ponto em que um dispositivo de entrada é acionado por um usuário até o ponto onde uma resposta é exibida no dispositivo de exibição) está no limite superior para jogos e aplicações que exigem um tempo de resposta rápido. Entretanto, isto é muito difícil de alcançar no contexto das arquiteturas mostradas nas figuras 2a e 2b devido a uma quantidade de restrições práticas e físicas.

Como indicado na figura 3, quando um usuário assina um serviço de Internet, a conexão é tipicamente classificada por uma taxa de dados máxima nominal 301 para a residência ou escritório do usuário. dependendo das políticas do provedor e capacidade do equipamento de encaminhamento, aquela taxa de dados máxima pode ser mais ou menos aplicada estritamente, mas tipicamente a taxa de dados disponível real é menor por uma de muitas razões diferentes. Por exemplo, pode haver muito tráfego na rede no escritório central de DSL ou no circuito de modem de cabo local, ou pode haver ruído no cabeamento que provoca perda de pacotes ou o provedor pode estabelecer uma quantidade máxima de bits por mês por usuário. Atualmente, a taxa de dados máxima serviços de cabo e DSL varia tipicamente de algumas centenas de Kilobits/segundo (Kbps) a 30 Mbps. Serviços celulares são tipicamente limitados a centenas de Kbps de fluxo de descida de dados. Entretanto, a velocidade de serviços de banda larga e a quantidade de usuários que assina serviços de banda larga aumentará

dramaticamente ao longo do tempo. Atualmente, alguns analistas estimam que 33% dos assinantes de banda larga nos EUA tenham uma taxa de dados para fluxo de descida de 2Mbps ou mais. Por exemplo, alguns analistas prevêem que em 2010, acima de 85% dos assinantes de banda larga nos EUA terá uma taxa de dados de 2Mbps ou mais.

Como indicado na figura 3, a taxa de dados máxima disponível atual 302 pode flutuar ao longo do tempo. Portanto, em um contexto de jogos e software aplicativo de baixa latência em linha algumas vezes é difícil prever a taxa de dados disponível para um fluxo contínuo de vídeo particular. Se a taxa de dados 303 requerida para sustentar um dado nível de qualidade a uma dada quantidade de quadros por segundo (fps) a uma dada resolução (por exemplo, 640 x 480 @ 60 fps) para uma certa complexidade de cenário e movimento sobe acima da taxa de dados máxima disponível atual 302 (como indicado pelo pico na figura 3), então podem ocorrer diversos problemas. Por exemplo, alguns serviços de Internet simplesmente perderão pacotes, resultando em perda de dados e imagens destorcidas / perdidas na tela de vídeo do usuário. Outros serviços irão armazenar temporariamente (ou seja, enfileirar) os pacotes adicionais e fornecer os pacotes para o cliente na taxa de dados disponível, resultando em um aumento na latência – um resultado inaceitável para muitos jogos de vídeo e aplicações. Finalmente, alguns provedores de serviço de Internet verão o aumento na taxa de dados como um ataque malicioso, tal como um ataque de recusa de serviço (uma técnica bem conhecida usada pelos hackers para desabilitar conexões de rede), e cortarão a conexão de Internet do usuário por um determinado período de. Portanto, as modalidades descritas neste documento tomam medidas para garantir que a taxa de dados requerida por um jogo de vídeo não exceda a taxa de dados máxima disponível.

Arquitetura de Serviço de Hospedagem

A figura 4a ilustra uma arquitetura do serviço de hospedagem 210 de acordo com uma modalidade. O serviço de hospedagem 210 pode ser localizado ou em um único centro de servidores, ou pode ser distribuída por uma pluralidade de centros de servidores (para fornecer conexões de

baixa latência para usuários que têm percursos de menor latência para certos centros de servidores do que para outros, para fornecer um balanceamento de carga entre os usuários, e para fornecer redundância no caso de um ou mais centros de servidores falharem). O serviço de hospedagem 210 pode incluir eventualmente centenas de milhares ou mesmo milhões de servidores 402, servindo uma base de usuários muito grande. Um sistema de controle de serviço de hospedagem 401 fornece controle geral para o serviço de hospedagem 210, e controla servidores, roteadores, sistemas de compressão de vídeo, sistemas de faturamento e contabilidade, etc. em uma modalidade, o sistema de controle de serviço de hospedagem 401 é implementado em um sistema baseado em Linux distribuído ligado à matrizes de RAID usadas para armazenar bancos de dados para informação de usuário, informação de servidor e estatísticas de sistema. Nas descrições acima, as várias ações implementadas no serviço de hospedagem 210, a menos que atribuídas a outros sistemas específicos, são iniciadas e controladas pelo sistema de controle de serviço de hospedagem 401.

O serviço de hospedagem 210 inclui uma quantidade de servidores 402 tais como aqueles disponíveis pela Intel, IBM e Hewlett Packard, e outros. Alternativamente, os servidores 402 podem ser montados em uma configuração personalizada de componentes, ou podem ser eventualmente integrados assim um servidor inteiro é implementado como um único circuito integrado. Embora este diagrama mostre uma pequena quantidade de servidores 402 para o propósito de ilustração, em um desenvolvimento atual podem haver tão poucos como apenas um servidor 402 ou tantos como milhões de servidores 402 ou mais. Os servidores 402 podem ser todos configurados da mesma forma (como exemplo de alguns parâmetros de configuração, com o mesmo tipo e performance de CPU; com ou sem uma GPU, e se com uma GPU, com o mesmo tipo e performance de GPU; com a mesma quantidade de CPUs e GPUs; com a mesma quantidade e tipo / velocidade de RAM; e com a mesma configuração de RAM), ou vários subconjuntos dos servidores 402 podem ter a mesma configuração (por exemplo, 25% dos servidores podem ser configurados de uma certa forma,

50% de uma forma diferente, e 25% ainda de outra forma), ou cada servidor 402 pode ser diferente.

Em uma modalidade, os servidores 402 são sem disco, ou seja, em vez de terem seu próprio armazenamento de massa (seja ele armazenamento ótico ou magnético, ou armazenamento baseado em semicondutor tal como memória Flash ou outros meios de armazenamento de massa servindo para uma função similar), cada servidos acessa armazenamento de massa compartilhado através de um painel passivo ou conexão de rede. Em uma modalidade, esta conexão rápida é uma Rede de Área de Armazenamento (SAN) 403 conectada a uma série de Matrizes redundantes de Discos Independentes (RAID) 405 com conexões entre dispositivos implementada usando Gigabit Ethernet. Como é conhecido pelos indivíduos versados na técnica, uma SAN 403 pode ser usada para combinar muitas matrizes de RAID 405, resultando em largura de banda extremamente alta — se aproximando ou potencialmente excedendo a largura de banda disponível a partir da RAM usada nas consoles de jogos e PCs. E, embora matrizes de RAID baseadas em mídia rotativa, tal como mídia magnética, frequentemente tenham latência de tempo de busca significativo, matrizes de RAID baseadas em armazenamento de semicondutor podem ser implementadas com muito menos latência de acesso. em outra configuração, alguns ou todos os servidores 402 fornecem parte ou todo o seu armazenamento de massa localmente. Por exemplo, um servidor 402 pode armazenar informação acessada frequentemente tal como seu sistema operacional e uma cópia de um jogo de vídeo ou aplicação em armazenamento baseado em Flash local de baixa latência, mas o mesmo pode utilizar o SAN para acessar matrizes de RAID 405 baseadas em mídia rotativa com latência de busca maior para acessar grandes bancos de dados de geometria ou informação de estado do jogo em bases menos frequentes.

Adicionalmente, em uma modalidade, o serviço de hospedagem 210 emprega lógica de compressão de vídeo de baixa latência 404 descrita em detalhes abaixo. A lógica de compressão de vídeo 404 pode ser implementada em software, hardware, ou qualquer combinação dos mesmos

(certas modalidades das quais são descritas abaixo). Lógica de compressão de vídeo 404 inclui lógica para comprimir áudio bem como material visual.

Em operação, enquanto joga um jogo de vídeo ou usa uma aplicação nas premissas de usuário 211 através de um teclado, rato, controlador de jogo ou outro dispositivo de entrada 421, a lógica de sinal de controle 413 no cliente 415 transmite sinais de controle 406a e b (tipicamente na forma de pacotes UDP) que representam pressionamentos de botões (e outros tipos de entradas de usuário) acionados pelo usuário para o serviço de hospedagem 210. Os sinais de controle de um dado usuário são encaminhadas para o servidor de aplicação apropriado (ou servidores, se múltiplos servidores são responsivos ao dispositivo de entrada de usuário) 402. Como ilustrado na figura 4a, sinais de controle 406a podem ser encaminhados para os servidores 402 através da SAN. Alternativa ou adicionalmente, sinais de controle 406b podem ser encaminhados diretamente para os servidores 402 sobre a rede do serviço de hospedagem (por exemplo, uma rede de área local baseada em Ethernet). Independentemente de como os mesmos são transmitidos, o servidor ou servidores executam o jogo ou software aplicativo em resposta aos sinais de controle 406a e b. Embora não ilustrados na figura 4a, vários componentes de rede tal como um firewall(s) e / ou porta(s) de ligação podem processar tráfego de entrada e de saída na borda do serviço de hospedagem 210 (por exemplo, entre o serviço de hospedagem 210 e a Internet 410) e / ou na borda das premissas de usuário 211 entre a Internet 410 e o cliente residencial ou de escritório 415. A saída gráfica ou de áudio do jogo ou software aplicativo executado — ou seja, novas sequências de imagem de vídeo — são fornecidas para a lógica de compressão de vídeo de baixa latência 404 que comprime as sequências de imagens de vídeo de acordo com as técnicas de compressão de vídeo de baixa latência, tais como aquelas descritas neste documento e transmite um fluxo contínuo de vídeo comprimido, tipicamente com áudio comprimido e não comprimido, de volta para o cliente 415 sobre a Internet 410 (ou como descrito abaixo, sobre um serviço de rede de alta velocidade otimizado que desvia da Internet geral). Lógica de descompressão de vídeo de baixa

latência 412 no cliente 415 em seguida descomprime os fluxos contínuos de vídeo e áudio e entrega o fluxo contínuo de vídeo descomprimido, e tipicamente toca o fluxo contínuo de áudio descomprimido em um dispositivo de exibição 422. Alternativamente, o áudio pode ser tocado em alto-falantes separados do dispositivo de exibição 422 ou não absolutamente. Deve ser observado que, apesar do fato de que o dispositivo de entrada 421 e dispositivo de exibição 422 serem mostrados como dispositivos autônomos nas figuras 2a e 2b, os mesmos podem ser integrados em dispositivos clientes tais como computadores portáteis ou dispositivos móveis.

Cliente residencial ou de escritório 415 (descrito anteriormente como cliente residencial ou de escritório 205 nas figuras 2a e 2b) pode ser um dispositivo muito barato e de baixa potência, com performance de computação ou gráfica muito limitada e pode ter armazenamento der massa local muito limitado ou nenhum. Ao contrário, cada servidor 402, acoplado a uma SAN 403 e múltiplas RAIDs 405 pode ser um sistema de computação de performance excepcionalmente alta, e na verdade, se múltiplos servidores são usados cooperativamente em uma configuração de processamento paralelo, quase não existe limite na quantidade de potência de processamento e gráfica que pode ser trazida para o suporte. E, devido a compressão de vídeo de baixa latência 404 e compressão de vídeo de baixa latência 412, perceptivamente para o usuário, a potência de computação dos servidores 402 está sendo fornecida para o usuário. Quando o usuário pressiona um botão no dispositivo de entrada 421, a imagem na exibição 422 é atualizada em resposta ao pressionamento do botão sem nem um atraso significativo perceptivamente, como se o jogo ou software aplicativo estivessem rodando localmente. Portanto, com um cliente residencial ou de escritório 415 que é um computador de performance muito baixa ou apenas um circuito integrado barato que implementa a descompressão de vídeo de baixa latência e lógica de controle de sinal 413, é fornecido para o usuário uma potência de computação efetivamente arbitrária a partir de uma localização remota que parece ser disponível localmente. Isto dá ao usuário potência para jogar os mais avançados jogos de vídeo de processador intensivo

(tipicamente novos) e as aplicações de performance mais alta.

A figura 4c mostra um dispositivo cliente 465 muito básico e barato. este dispositivo é uma modalidade de cliente residencial ou de escritório 415 das figuras 4a e 4b. O mesmo tem aproximadamente 5 centímetros de comprimento. O mesmo tem uma tomada Ethernet 462 que faz interface com um cabo Ethernet com Energia sobre Ethernet (PoE), a partir do qual o mesmo deriva sua energia e sua conectividade para a Internet. O mesmo é capaz de rodar Tradução de Endereço de Rede (NAT) dentro de uma rede que suporta NAT. Em um ambiente de escritório, muitos novos comutadores Ethernet têm PoE e trazem PoE diretamente para uma tomada Ethernet em um escritório. Em uma situação como esta, tudo que é requerido é um cabo Ethernet de uma tomada de parede para o cliente 465. Se a conexão Ethernet disponível não transportar energia (por exemplo, em uma residência com um modem DSL ou cabo, mas nenhum PoE), então existem "tijolos" de parede baratos (ou seja, fontes de energia) disponíveis que aceitam cabo Ethernet e fornecem Ethernet com PoE.

O cliente 465 contém lógica de controle de sinal 413 (da figura 4a) que é acoplada a uma interface sem fio Bluetooth, que faz a interface com dispositivos de entrada Bluetooth 479, tais como um teclado, rato, controlador de jogo e / ou microfone e / ou fone de ouvido. Também, uma modalidade de cliente 465 é capaz de fornecer vídeo a 120fps acoplado com um dispositivo de exibição 468 capaz de suportar 120fps de vídeo e sinal (tipicamente através de infravermelho) um par de óculos obturado 466 para fechar alternada mente um olho, depois o outro com cada quadro sucessivo.O efeito percebido pelo usuário é o de uma imagem estereoscópica 3D que "salta para fora" da tela de exibição. Um dispositivo de exibição 468 como este que suporta tal operação é o Samsung HL-T5076S. Uma vez que o fluxo contínuo de vídeo para cada olho é separado, em uma modalidade dois fluxos contínuo de vídeo independentes são comprimidos pelo serviço de hospedagem 210, os quadros são entrelaçados no tempo, e os quadros são descomprimidos como dois processos de descompressão independentes dentro do cliente 465.

O cliente 465 também contém lógica de descompressão de vídeo de baixa latência 412, que descomprime vídeo e áudio e fornece através do conector HDMI (Interface Multimídia de Alta Definição) 463 que conecta em uma SDTV (Televisão de Definição Padrão) ou HDTV (Televisão de Alta Definição) 468, fornecendo vídeo e áudio para a TV, ou para um monitor 468 que suporte HDMI. Se o monitor do usuário 468 não suporta HDMI, então pode ser usado um HDMI para DVI (Interface Visual Digital), mas o áudio será perdido. Sob o padrão HDMI, as capacidades de exibição (por exemplo resoluções suportadas, taxas de quadro) 464 são comunicadas do dispositivo de exibição 468, e esta informação é então passada de volta através da conexão de Internet 462 de volta para o serviço de hospedagem 210 assim o mesmo pode transmitir vídeo comprimido em um formato adequado para o dispositivo de exibição.

A figura 4d mostra um dispositivo cliente residencial ou de escritório 475 que é o mesmo que o dispositivo cliente residencial ou de escritório 465 mostrado na figura 4c exceto pelo fato de que tem mais interfaces externas. Também, o cliente 475 pode aceitar ou PoE para energia, ou a mesma pode escoar de um adaptador de fonte de alimentação externa (não mostrado) que conecta na parede. Usando entrada USB do cliente 475,a câmera de câmera 477 fornece vídeo comprimido para o cliente 475, que é carregado pelo cliente 475 para o serviço de hospedagem 210 para o uso descrito abaixo. A câmera embutida 477 é um compressor de baixa latência que utiliza as técnicas de compressão descritas abaixo.

Adicionalmente a ter um conector Ethernet para sua conexão a Internet, o cliente 475 também tem uma interface sem fio 802.11g para a Internet. Ambas interfaces são capazes de usar NAT dentro de uma rede que suporta NAT.

Também, adicionalmente a ter um conector HDMI para fornecer vídeo e áudio, o cliente 475 também tem um conector Dual Link DVI-I, que inclui saída analógica (e com um cabo adaptador padrão fornecerá saída VGA). O mesmo também tem saída analógica para vídeo composto e S-vídeo.

Para áudio, o cliente 475 tem tomadas esquerda / direita RCA estéreo analógicas, e para saída de áudio digital o mesmo tem uma saída TOSLINK.

Adicionalmente a uma interface sem fio Bluetooth para dispositivos de entrada 479, o mesmo tem tomadas USB para fazerem interface com dispositivos de entrada.

A figura 4e mostra uma modalidade da arquitetura interna do cliente 465. Todos ou alguns dispositivos mostrados no diagrama podem ser implementados em um Arranjo de Lógica Programável no Campo, um ASIC personalizado ou em diversos dispositivos discretos, ou projetados personalizados ou de prateleira.

Ethernet com PoE 497 conecta a Interface Ethernet 481. A energia 499 é derivada da Ethernet com PoE 497 e é conectada ao resto dos dispositivos no cliente 465. O barramento 480 é um barramento comum para comunicação entre dispositivos.

A CPU de controle 483 (quase qualquer CPU pequena, como uma CPU da série MIPS R4000 a 100MHz com RAM embutida é adequada) que executa um pequeno aplicativo de controle de cliente da Flash 476 implanta a pilha de protocolo para a rede (ou seja, interface Ethernet) e também comunica com o Serviço de Hospedagem 210, e configura todos os dispositivos no cliente 465. Também manuseia as interfaces com os dispositivos de entrada 469 e envia os pacotes de volta para o serviço de hospedagem 210 com dados de controlador de usuário, protegido por Correção de Erro Antecipada, se necessário. Ademais, a CPU de controle 483 monitora o tráfego de pacote (por exemplo, se os pacotes são perdidos ou atrasados, e também carimba com data/hora sua chegada). Essas informações são enviadas de volta para o serviço de hospedagem 210 de modo que possa monitorar constantemente a conexão de rede e ajustar o que envia consequentemente. A memória rápida 476 é inicialmente carregada no momento da fabricação com o programa de controle para a CPU de controle 483 e também com um número de série que é único para a unidade de Cliente 465 particular. Esse número de série permite que o serviço de hospedagem 210

para identificar de maneira única a unidade de cliente 465.

A interface de Bluetooth 484 se comunica com dispositivos de entrada 469 sem fio através de sua antena, interna ao cliente 465.

O descompressor de vídeo 486 é um descompressor de vídeo de baixa latência configurado para implantar a descompressão de vídeo descrita no presente documento. Um número grande de dispositivos de descompressão de vídeo existe, ou em circulação, ou como Propriedade Intelectual (IP) de um design que pode ser integrado em um FPGA ou um ASIC padrão. Uma empresa que oferece um IP para um decodificador H.264 é a Ocean Logic de Manly, NSW Austrália. A vantagem de usar IP é que as técnicas de compressão usadas no presente documento não se conformam aos padrões de compressão. Alguns descompressores padrão são flexíveis o suficiente para serem configurados para acomodar as técnicas de compressão do presente documento, mas alguns não podem. Porém, com IP, há uma flexibilidade completa no reprojeto do descompressor, conforme necessário.

A saída do descompressor de vídeo é acoplada ao subsistema de saída de vídeo 487, que acopla o vídeo à saída de vídeo da interface de HDMI 490.

O subsistema de descompressão de áudio 488 é implantado ou com o uso de um descompressor de áudio padrão que está disponível, ou pode ser implantado como IP, ou a descompressão de áudio pode ser implantada no processador de controle 483 que poderia, por exemplo, implantar o descompressor de áudio Vorbis (disponível em Vorbis.com).

O dispositivo que implanta a descompressão de áudio é acoplado ao subsistema de saída de áudio 489 que acopla o áudio à saída de áudio da interface de HDMI 490.

A figura 4f mostra uma modalidade da arquitetura interna do cliente 475. Conforme pode ser visto, a arquitetura é a mesma que a do cliente 465 exceto pelas interfaces adicionais e potência de CD externa opcional de um adaptador de abastecimento elétrico que se pluga à parede, e se for usado, substituía a potência que viria do Eternet PoE 497. A fun-

cionalidade que está em comum com o cliente 465 não será repetida abaixo, mas a funcionalidade adicional é como segue.

A CPU 483 se comunica com e configura os dispositivos adicionais.

O subsistema de WiFi 482 fornece acesso à Internet sem fio como uma alternativa à Eternet 497 através de sua antena. Os subsistemas de WiFi estão disponíveis junto a uma ampla gama de fabricantes, incluindo Atheros Communications of Santa Clara, CA.

O subsistema de USB 485 fornece uma alternativa para a comunicação do tipo Bluetooth para dispositivos e entrada de USB com fio 479. Os subsistemas de USB são padrão e prontamente disponíveis para FPGAs e ASICs, bem como frequentemente embutidos dispositivos em circulação que realizam outras funções, como descompressão de vídeo.

O subsistema de saída de vídeo 487 produz uma faixa mais ampla de saídas de vídeo que no cliente 465. Além do fornecimento de saída de vídeo de HDMI 490, o mesmo fornece DVI-I 491, S-vídeo 492, e vídeo compósito 493. Além disso, quando a interface DVI-I 491 é usada para vídeo digital, as capacidades de exibição 464 são passadas de volta para o dispositivo de exibição para a CPU de controle 483 de modo que possa notificar o serviço de hospedagem 210 as capacidades do dispositivo de exibição 478. Todas as interfaces fornecidas pelo subsistema de saída de vídeo 487 são interfaces padrão e prontamente disponíveis sob muitas formas.

O subsistema de saída de áudio 489 emite áudio digitalmente através da interface digital 494 (S/PDIF e / ou Toslink) e áudio sob a forma analógica através da interface analógica estéreo 495.

ANÁLISE DE LATÊNCIA DE CICLO

Naturalmente, para que os benefícios do parágrafo anterior sejam realizados, a latência de ida e volta entre a ação do usuário que usa o dispositivo de entrada 421 e tendo em vista a consequência daquela ação sobre o dispositivo de exibição 420 não deveria ser maior que 70 a 80 ms. Esta latência deve considerar todos os fatores no caminho do dispositivo de

entrada 421 nas premissas de usuário 211 para o serviço de hospedagem 210 e de volta novamente para as premissas de usuário 211 para o dispositivo de exibição 422. A figura 4b ilustra os vários componentes e redes sobre os quais os sinais devem percorrer, e acima destes componentes e redes está uma linha de tempo que lista latências exemplificadoras que se esperam em uma implantação prática. Notar que a figura 4b está simplificada de modo que apenas o roteamento do caminho crítico seja mostrado. Outro roteamento de dados usado para outros recursos do sistema está descrito abaixo. Setas com duas pontas (por exemplo, seta 453) indicam latência de ida e volta e setas com uma ponta (por exemplo, seta 457) indicam latência de um único trajeto, e "~" denota uma medição aproximada. Deve-se apontar que haverá situações de mundo real nas quais as latências listadas não podem ser alcançadas, mas em um número maior de casos no US, com o uso de DSL e conexões de modem a cabo em relação às premissas de usuário 211, estas latências podem ser alcançadas nas circunstâncias descritas no próximo parágrafo. Além disso, notar que, embora a conectividade sem fio celular com a Internet trabalhará certamente no sistema mostrado, a maioria os sistemas de dados celulares de US (como EVDO) incorre em latências muito elevadas e não seriam capazes de alcançar as latências mostradas na figura 4b. Entretanto, estes princípios subjacentes podem ser implantados em futuras tecnologias de celular que podem ser capazes de implantar este nível de latência. Adicionalmente, há cenários de aplicativos e jogos (por exemplo, jogos que não exigem tempo rápido de reação de usuário, como xadrez) onde a latência incorreu através de um sistema de dados de celular de US, embora notável pelo usuário, seria aceitável para o jogo ou aplicativo.

A partir do dispositivo de entrada 421 nas premissas do usuário 211, uma vez que o usuário aciona o dispositivo de entrada 421, um sinal de controle de usuário é enviado ao cliente 415 (que pode ser um dispositivo autônomo como um conversor digital, ou pode ser *software* ou *hardware* em execução em outro dispositivo como um computador pessoal ou um dispositivo móvel), e é distribuído em pacotes (no formato UDP em uma modali-

dade) e o pacote recebe um endereço de destino para alcançar o serviço de hospedagem 210. O pacote também irá conter informações para indicar de qual usuário os sinais de controle estão sendo originados. O(s) pacote(s) de sinal(is) de controle é/são então emitidos através do dispositivo de *Firewall*/Roteador/NAT (Tradução de Endereço de Rede) 443 para a interface WAN 442. A interface WAN 442 é o dispositivo de interface fornecido às premissas do usuário 211 por meio do ISP do usuário (Provedor de Serviços de Internet). A Interface WAN 442 pode ser um cabo ou modem DSL, um transceptor WiMax, um transceptor de fibra, uma interface de dados de celular, uma interface de protocolo IP por rede elétrica, ou qualquer outra dentre muitas interfaces com a Internet. Além disso, o dispositivo de *Firewall*/Roteador/NAT 443 (e, possivelmente, a interface WAN 442) pode ser integrado ao cliente 415. Um exemplo disso seria um telefone móvel, que inclui *software* para implantar a funcionalidade do cliente domiciliar ou corporativo 415, bem como os meios para roteamento e conexão com a Internet por meio de comunicação sem fio através de algum padrão (por exemplo, 802.11g).

A interface WAN 442, então, efetua o roteamento dos sinais de controle para o que poderia ser chamado na presente invenção de "ponto de presença" 441 do Provedor de Serviços de Internet (ISP) do usuário, que são as instalações que oferecem uma interface entre o transporte WAN conectado às premissas do usuário 211 e a Internet geral ou rede privada. As características do ponto de presença irão variar dependendo da natureza do serviço de Internet oferecido. Em DSL, este tipicamente será um escritório central de uma empresa de telefonia onde um DSLAM está localizado. Para modens a cabo, este tipicamente será um ponto central de recepção de sinais da multioperadora de sistemas a cabo (MSO). Para sistemas de celulares, este tipicamente será uma sala de controle associada à torre de celular. Mas seja qual for a natureza do ponto de presença, este, então, irá efetuar o roteamento do(s) pacote(s) de sinal(is) de controle para a Internet geral 410. O(s) pacote(s) de sinal(is) de controle será/serão então roteado(s) para a interface WAN 441 do serviço de hospedagem 210, através daquilo

que provavelmente será uma interface do transceptor de fibra. A WAN 441 irá, então, efetuar o roteamento dos pacotes de sinais de controle para a lógica de roteamento 409 (que pode ser implantada de diversas e diferentes maneiras, incluindo comutadores Ethernet e servidores de roteamento), que avalia o endereço do usuário e efetua o roteamento do(s) sinal(is) de controle para o servidor correto 402 do usuário específico.

O servidor 402, então, adota os sinais de controle como entrada para o jogo ou aplicativo de software que está operando no servidor 402 e usa os sinais de controle para processar o próximo quadro do jogo ou aplicativo. Uma vez que os próximos quadros I são gerados, o vídeo e o áudio são emitidos a partir do servidor 402 para o compressor de vídeo 404. O vídeo e o áudio podem ser emitidos a partir do servidor 402 para o compressor 404 através de vários meios. Para iniciar isto, o compressor 404 pode ser embutido no servidor 402, então, a compressão pode ser implantada localmente no interior do servidor 402. Ou, o vídeo e/ou áudio pode ser emitido na forma de pacotes através de uma conexão de rede tal como uma conexão de Ethernet para uma rede que é uma rede privada entre o servidor 402 e o compressor de vídeo 404, ou através de uma rede compartilhada, tal como SAN 403. Ou, o vídeo pode ser emitido através de um conector de saída de vídeo do servidor 402, tal como um conector DVI ou VGA e, então, capturado pelo compressor de vídeo 404. Ademais, o áudio pode ser emitido a partir do servidor 402 como áudio digital (por exemplo, através de um conector TOSLINK ou S/PDIF) ou como áudio analógico, que é digitalizado e encodificado por lógica de compressão de áudio dentro do compressor de vídeo 404.

Uma vez que o compressor de vídeo 404 capturou o quadro de vídeo e o áudio gerado durante aquele tempo de quadro do servidor 402, o compressor de vídeo irá comprimir o vídeo e o áudio com o uso de técnicas descritas abaixo. Uma vez que o vídeo e o áudio é comprimido, isto é disposto em pacotes com um endereço para enviá-lo de volta para o cliente do usuário 415, e é roteado para a Interface WAN 441, que, então, roteia os pacotes de vídeo e áudio através da Internet geral 410, que, então, roteia os

pacotes de vídeo e áudio para o ponto ISP de presença do usuário 441, que roteia os pacotes de vídeo e áudio para a interface WAN 442 nas premissas do usuários, que roteia os pacotes de vídeo e áudio para o dispositivo de Firewall/Roteador/NAT 443, que, então, roteia os pacotes de vídeo e áudio para o cliente 415.

O cliente 415 descomprime o vídeo e o áudio e, então, exibe o vídeo no dispositivo de exibição 422 (ou no dispositivo de exibição embutido no cliente) e envia o áudio para o dispositivo de exibição 422 ou para amplificador/alto-falantes separados ou amplificador/alto-falantes embutidos no cliente.

Para o usuário perceber que todo o processo descrito agora está perceptivamente sem defasagem, o atraso de ida e volta precisa ser menor que 70 ou 80 ms. Alguns dos atrasos de latência na trajetória de ida e volta descrita estão sob o controle do serviço de hospedagem 210 e/ou do usuário e outros não estão. Entretanto, com base na análise e no teste de um grande número de cenários do mundo real, as seguintes são medições apropriadas.

O tempo de transmissão de um caminho para enviar os sinais de controle 451 é tipicamente menor do que 1 ms, o roteiro de ida e volta através das premissas de usuário 452 é tipicamente realizado, usando prontamente comutadores Firewall/Router/NAT de grau de consumo disponíveis na Ethernet em cerca de 1 ms. O ISPs de usuário pode variar amplamente em seus atrasos de ida e volta 453, mas com DSL e fornecedores de modem a cabo, se vê tipicamente entre 10 e 25 ms. A latência de ida e volta na Internet geral 410 pode variar enormemente dependendo de como o tráfico é roteado e se existem quaisquer falhas na rota (e essas questões são discutidas abaixo), mas, tipicamente, a Internet geral fornece rotas justamente ótimas e a latência é amplamente determinada pela velocidade da luz através de fibra óptica, dada a distância para o destino. Conforme discutido adicionalmente abaixo, foram estabelecidas 1.000 milhas como uma distância aproximadamente maior do que a esperada para colocar um serviço de hospedagem 210 longe de premissas de usuário 211. Em 1.000 milhas

(2.000 milhas de ida e volta) o tempo de transito prático para um sinal através da Internet é de aproximadamente 22 ms. A Interface WAN 441 para o serviço de hospedagem 210 é tipicamente uma interface de alta velocidade de fibra de grau comercial com latência negligenciável. Dessa forma, a latência da Internet geral 454 é tipicamente entre 1 e 10 ms. A latência da rota de um caminho 455 através do serviço de hospedagem 210 pode ser alcançada em menos do que 1 ms. O servidor 402 tipicamente computará um novo quadro para um jogo ou um aplicativo em menos do que um tempo de quadro (que em 60 fps é 16,7 ms) de modo que 16 ms seja uma latência de um caminho máxima razoável 456 para o uso. Em uma implantação de hardware otimizada da compressão de vídeo e algoritmos de compressão de áudio descrito aqui, a compressão 457 pode ser finalizada em 1 ms. Em versões menos otimizadas, a compressão pode levar tanto quanto 6ms (naturalmente, mesmo as versões menos otimizadas poderiam levar mais tempo, mas essas implantações impactariam na latência geral de ida e volta e exigiriam que outras latências fossem mais curtas (por exemplo, a distância permitida através da Internet geral poderia ser reduzida) para manter o alvo de latência de 70 a 80 ms). As latências de ida e volta da Internet 454, ISP de Usuário 453 e Roteamento de Premissas do Usuário 452 foram ainda consideradas, de modo que o que resta é a latência da descompressão de vídeo 458 que, dependendo de se a descompressão de vídeo 458 é implantada em hardware dedicado ou se é implantada em software em um dispositivo cliente 415 (como um dispositivo móvel ou PC) essa pode variar dependendo do tamanho da exibição e do desempenho da CPU de descompressão. Tipicamente, a descompressão 458 leva entre 1 e 8ms.

Dessa forma, ao adicionar todas as latências de pior caso vistas na prática, pode-se determinar a pior latência de caso de ida e volta que pode ser esperada para ser experimentada por um usuário do sistema mostrado na figura 4a. Essas são: 1+1+25+22+1+16+6+8 = 80ms. E, de fato, na prática (com avisos discutidos abaixo), isso é aproximadamente a latência de ida e volta vista com o uso de versões de protótipo do sistema mostrado na figura 4a, usando PCs de Windows padrão como dispositivos

clientes e conexões de modem a cabo e DSL home dentro do US. Naturalmente, os cenários melhores do que o pior caso podem resultar em latências muito mais curtas, mas esses não podem ser confiáveis mediante o desenvolvimento de um serviço comercial que é usado amplamente.

A fim de alcançar as latências listadas nas figuras 4b sobre a Internet em geral requer que o compressor de vídeo 404 e descompressor de vídeo 412 da figura 4a no cliente 415 gere um fluxo de pacote com características muito particulares, de modo que a sequência de pacotes gerada através de todo o caminho desde o serviço de hospedagem 210 até o dispositivo de exibição 422 não está sujeito a atrasos ou perda de pacote excessiva e, em particular, se encaixa consistentemente com as restrições da largura de banda disponível ao usuário através da conexão à Internet do usuário através da interface WAN 442 e Firewall/Roteador/NAT 443. Adicionalmente, o compressor de vídeo deve criar um fluxo de pacote que é robusto o suficiente para tolerar a perda de pacote e reordenação de pacote inevitável que ocorre em transmissões de Internet e rede normais.

COMPRESSÃO DE VÍDEO DE BAIXA LATÊNCIA

Para atingir os objetivos precedentes, uma modalidade utiliza uma nova abordagem para a compressão de vídeo que diminui os requerimentos de largura de banda para a latência e o pico para a transmissão de vídeo. Antes da descrição destas modalidades, uma análise de técnicas de compressão de vídeo atuais serão fornecidas com respeito à figura 5 e figuras 6a a b. Naturalmente, estas técnicas podem ser empregadas de acordo com princípios fundamentais se o usuário é fornecido com largura de banda o suficiente para suportar a taxa de dados requisitada por estas técnicas. Note-se que a compressão de áudio não é tratada no presente documento a não ser para afirmar que a mesma é implementada simultaneamente e em sincronia com a compressão de vídeo. Existem técnicas de compressão de áudio anteriores que satisfazem os requerimentos para este sistema.

A figura 5 ilustra uma técnica anterior em particular para comprimir vídeo no qual cada quadro de vídeo individual 501 a 503 é comprimido

através de lógica de compressão 520 com o uso de um algoritmo de compressão particular para gerar uma série de quadros comprimidos 511 a 513. Uma modalidade desta técnica é "JPEG em movimento" na qual cada quadro é comprimido de acordo com um algoritmo de compressão de Grupo de Peritos Fotográfico Comum (JPEG), baseado no transformação discreta de cosseno (DCT). Vários tipos diferentes de algoritmos de compressão podem ser empregados, entretanto, embora ainda cumpra com esses princípios fundamentais (por exemplo, algoritmos de compressão baseados em ondas pequenas, como JPEG-2000).

Um problema com este tipo de compressão é que reduz a taxa de dados de cada quadro, mas não explora as similaridades entre os quadros sucessivos para reduzir uma taxa de dados do fluxo contínuo total de vídeo. Por exemplo, como ilustrado na figura 5, supondo uma taxa de quadro de 640x480x24bits/pixel = 640*480*24/8/1024=900 Kilobites/quadro (KB/quadro), para uma dada qualidade de imagem, moviemento JPEG pode apenas comprimir o fluxo por um fator de 10, resutando em um fluxo de dados de 90 KB/quadro. A 60 quadros/s, isto necessitaria de uma largura de banda de canal de 90 KB * 8 bits * 60 quadros/s = 42,2Mbps, o que seria uma largura de banda muito grande para quase todas as as conexões caseiras de internet nos Estados Unidos hoje em dia, e uma largura de banda muito grande para muitas conexões de internet de escritórios. De fato, sabendo-se que precisa-se de um fluxo constante de dados em uma largura de banda tão grande, e estaria apenas servindo um usuário, mesmo em um ambiente LAN de escritório, consumiria uma grande porcentagem de uma largura de banda Ethernet LAN 100Mbps e sobrecarregar muito a comutação de Ethernet que suporta a LAN. Deste modo, a compressão para vídeo de moviento é ineficiente quando comparada com outras técnicas de compressão (como aquelas descritas abaixo). Além disso, um quadro único de algoritmos de compressão como JPEG e JPEG-2000 que usa algoritmos de compressão em perda produz artefatos de compressão que podem não estar visíveis em imagens estáticas (por exemplo, um artefato dentro de uma vegetação densa na cena pode não aparecer como um artefato, uma vez

que os olhos não conhecem exatamente como a vegetação densa deve aparecer). Porém, uma vez que a cena está em movimento, um artefato pode se destacar uma vez que os olhos detectam que o artefato foi alterado de quadro a quadro, apesar do fato de o artefato estar em uma área da cena em que pode não estar visível em uma imagem estática. Isto resulta na percepção do "ruído de fundo" na sequência de quadros, de aparência similar ao ruído do tipo "neve" visível durante recepção de TV analógica marginal. Naturalmente, esse tipo de compressão pode ser usado ainda em certas modalidades descritas no presente documento, porém, em geral, para evitar ruído de fundo na cena, uma elevada taxa de dados (ou seja, uma baixa taxa de compressão) é necessária para uma determinada qualidade perceptiva.

Outros tipos de compactação, como H.264, ou Windows Media VC9, MPEG2 e MPEG4 são mais eficientes para compactar um fluxo de vídeo, pois esses exploram as similaridades entre quadros sucessivos. Essas técnicas contam com as mesmas técnicas gerais de compactação de vídeo. Assim, embora o padrão H.264 seja descrito, os mesmos princípios gerais se aplicam a vários outros algoritmos de compactação. Um grande número de compactadores e descompactadores H.264 está disponível, inclusive a biblioteca de software de fonte aberta x264 para compactar H.264 e as bibliotecas de software de fonte aberta FFmpeg para descompactar H.264.

As figuras 6a e 6b ilustram uma técnica de compactação da técnica anterior exemplificativa em que uma série de quadros de vídeos descompactados 501-503, 559-561 é compactada por lógica de compactação 620 em uma série de "quadros I" 611, 671; "P-quadros" 612-613; e "B-quadros" 670. O eixo geométrico vertical na figura 6a geralmente significa o tamanho resultante de cada quadro codificado (embora os quadros não sejam representados em escala). Como descrito acima, a codificação de vídeo utilizando I-quadros, B- quadros e P-quadros é bem entendida pelos elementos versados na técnica. Brevemente, um I-quadro 611 é uma compactação baseada em DCT de um quadro descompactado completo 501 (similar à imagem JPEG compactada como descrito acima). Os P-quadros 612-613

geralmente são significativamente menores em tamanho do que os I-quadros 611, pois esses tiram vantagem dos dados no I-quadro anterior ou P-quadro; ou seja, esses contêm dados indicando as alterações entre o quadro I anterior ou P-quadro. Os B-quadros 670 são similares àqueles de P-quadros exceto pelo fato de que os B-quadros utilizam o quadro no seguinte quadro de referência bem como potencialmente o quadro no quadro de referência anterior.

Para a discussão a seguir, supõe-se que a taxa de quadro desejada seja 60 quadros/segundo, que cada I-quadro seja aproximadamente 160 Kb, a média de P-quadro e B-quadro seja 16 Kb e que um novo I-quadro seja gerado a cada segundo. Com esse conjunto de parâmetros, a taxa de dados média poderia ser: 160 Kb + 16 Kb 59 = 1,1Mbps. Essa taxa de dados está bem situada dentro da taxa de dados máxima para muitas conexões de Internet de banda larga atuais a residências e escritórios. Essa técnica também tende a evitar o problema de ruído de fundo de codificação somente intraquadro, pois os P e B quadros rastreiam diferenças entre os quadros, então os artefatos de compactação não tendem a aparecer e desaparecer de quadro para quadro, reduzindo assim o problema de ruído de fundo descrito acima.

Um problema com os tipos anteriores de compactação é que embora a taxa de dados média seja relativamente baixa (por exemplo, 1,1Mbps), um único I-quadro pode levar vários tempos de quadro para realizar a transmissão. Por exemplo, utilizando as práticas da técnica anterior, uma conexão de rede de 2,2 Mbps (por exemplo, DSL ou modem a cabo com pico de 2,2Mbps de taxa de dados máxima disponível 302 da figura 3a) poderia ser tipicamente adequada para transmitir o vídeo em 1,1 Mbps com um quadro I de 160Kbps a cada 60 quadros. Isso poderia ser realizado ao enfileirar o descompactador 1 segundo de vídeo antes de descompactar o vídeo. Em 1 segundo, 1,1Mb de dados poderia ser transmitido, isso poderia ser facilmente acomodado por uma taxa de dados máxima disponível de 2,2Mbps, mesmo supondo que a taxa de dados disponível possa cair periodicamente até 50%. Infelizmente, essa abordagem da técnica anterior

poderia resultar em uma latência de 1 segundo para o vídeo devido ao buffer de vídeo de 1 segundo no receptor. Tal atraso é adequado para muitas aplicações da técnica anterior (por exemplo, a reprodução de vídeo linear), porém é uma latência longa para videogames de ação rápida que não podem tolerar mais de 70 a 80ms de latência.

Se uma tentativa for feita para eliminar o buffer de vídeo de 1 segundo, ainda não poderia resultar em uma redução adequada em latência para videogames de ação rápida. Para alguém, o uso de B-quadros, como anteriormente descrito, poderia precisar da recepção de todos os B-quadros que precedem um I-quadro bem como o I-quadro. Se for considerado que os 59 não-I quadros são mais ou menos divididos entre os P e B quadros, então poderia haver pelo menos 29 B-quadros e um I-quadro recebido antes que qualquer B-quadro possa ser exibido. Assim, independente da largura de banda disponível do canal, poderia ser necessário um atraso de 29+1=30 quadros de duração de 1/60 segundo cada, ou 500ms de latência. Claramente que é excessivamente longo.

Assim, outra abordagem poderia ser eliminar os B-quadros e utilizar apenas os I e P-quadros. (Uma consequência disso é que a taxa de dados poderia aumentar para um determinado nível de qualidade, porém em consideração à consistência nesse exemplo, continua-se a considerar que cada I-quadro possui 160Kb e o P-quadro médio possui 16Kb de tamanho, e assim a taxa de dados ainda é 1,1Mbps). Essa abordagem elimina a latência inevitável introduzida por B-quadros, visto que a decodificação de cada P-quadro depende apenas do quadro recebido anterior. Um problema que permanece com essa abordagem é que um I-quadro é muito maior do que um P-quadro médio, aquele em um canal de largura de banda baixa, como é típico na maioria das residências e em muitos escritórios, a transmissão do I-quadro acrescenta latência substancial. Isso é ilustrado na figura 6b. A taxa de dados de fluxo de vídeo 624 está abaixo da taxa de dados máxima disponível 621 exceto para os I-quadros, onde a taxa de dados pico exigida para os I-quadros 623 excede muito a taxa de dados máxima disponível 622 (e ainda a taxa de dados máxima classificada 621). A taxa de dados exigida

pelos P-quadros é menor do que a taxa de dados máxima disponível. Mesmo que a taxa de dados máxima disponível chegue ao máximo em 2,2Mbps essa permanece constante em sua taxa de pico de 2,2Mbps, serão necessários 160Kb/2,2Mb=71ms para transmitir o I-quadro, e se a taxa de dados máxima disponível 622 cair para 50% (1,1Mbps), serão necessários 142ms para transmitir o I-quadro. Então, a latência para transmitir o I-quadro será reduzida em algum lugar entre 71 a 142ms. Essa latência é aditiva às latências identificadas na figura 4b, que no pior caso adicionou 70 ms, então isso poderia resultar em uma latência de ida e volta total de 141 a 222ms a partir do ponto que o usuário atua o dispositivo de entrada 421 até uma imagem aparecer no dispositivo de exibição 422, essa é muito alta. E se a taxa de dados máxima disponível cair abaixo de 2,2Mbps, a latência irá aumentar adicionalmente.

Nota-se também que geralmente há consequências severas para "interferência" de um ISP com taxa de dados de pico 623 que excede muito a taxa de dados disponível 622. O equipamento em ISPs diferentes irão se comportar de forma diferente, porém os comportamentos a seguir são muito comuns entre DSL e ISPs de modem a cabo quando os pacotes forem recebidos em uma taxa de dados muito maior do que a taxa de dados disponível 622: (a) atrasar os pacotes ao colocar os mesmos na fila (latência de introdução), (b) descartar alguns ou todos os pacotes, (c) desabilitar a conexão durante um período de tempo (mais provavelmente devido ao ISP ser referido como um ataque malicioso, como ataque de "negação de serviço"). Assim, a transmissão de um fluxo de pacotes em taxa de dados total com características como aquelas mostradas na figura 6b não é uma opção viável. Os picos 623 podem ser enfileirados no serviço de hospedagem 210 e enviados em uma taxa de dados abaixo da taxa de dados máxima disponível, introduzindo a latência inaceitável descrita no parágrafo anterior.

Ademais, a sequência de taxa de dados de fluxo de vídeo 624 mostrada na figura 6b é uma sequência de taxa de dados de fluxo de vídeo muito "enfadonha" e poderia ser a classificação de sequência de taxa de

dados que alguém poderia esperar que resulte da compactação do vídeo de uma sequência de vídeo que não muda muito e possui muito pouco movimento (por exemplo, como é comum em teleconferência de vídeo onde as câmeras estão em uma posição fixa e possuem pouco movimento, e os objetos, na cena, por exemplo, pessoas conversando sentadas, mostram pouco movimento).

A sequência de taxa de dados de fluxo de vídeo 634 mostrada na figura 6c é uma sequência típica do que alguém poderia esperar ver do vídeo com muito mais ação, conforme deve ser gerado em um filme cinematográfico ou um videogame, ou em algum software de aplicativo. Nota-se que além dos picos de I-quadro 633, também há picos de P-quadro como 635 e 636 que são muito grandes e excedem a taxa de dados máxima disponível em muitas ocasiões. Embora esses picos de P-quadro não sejam tão grandes quanto os picos de I-quadro, esses ainda são muito grandes para serem transmitidos pelo canal em taxa de dados total, e conforme com os picos de I-quadro, esses picos de P-quadro devem ser lentamente transmitidos (com isso há cada vez mais latência).

Em um canal de largura de banda alta (por exemplo, uma LAN de 100Mbps, ou uma conexão privada de largura de banda alta de 100Mbps) a rede poderia ser capaz de tolerar grandes picos, como picos de I-quadro 633 ou picos de P-quadro 636, e em princípio, a baixa latência poderia ser mantida. Porém, tais redes são frequentemente compartilhadas entre muitos usuários (por exemplo, em um ambiente de trabalho), e tais dados "com picos" poderiam causar um impacto no desempenho da LAN, particularmente se o tráfego de rede for roteado até uma conexão compartilhada privada (por exemplo, de um centro de dados remoto até um escritório). Para começar, deve-se ter em mente que esse exemplo possui um fluxo de vídeo de resolução relativamente baixa de 640x480 pixels em 60fps. Os fluxos de HDTV de 1920x1080 em 60fps são facilmente manipulados por computadores e monitores modernos, e as exibições de resolução 2560x1440 em 60fps estão cada vez mais disponíveis (por exemplo, monitor da Apple, Inc.'s 30"). Uma sequência de vídeo de alta ação em 1920x1080 em 60fps pode

exigir 4,5 Mbps utilizando compactação H.264 para um nível de qualidade razoável. Se for adotado o pico de I-quadros em 10X a taxa de dados nominal, que poderia resultar em picos de 45Mbps, bem como o pico de P-quadro menor, porém ainda considerável. Se vários usuários estiverem recebendo fluxos de vídeo na mesma rede de 100Mbps (por exemplo, uma conexão de rede privada entre um escritório e um centro de dados), é fácil observar como os picos de fluxo de vídeo de vários usuários poderiam se alinhar, cobrindo a largura de banda da rede, e potencialmente cobrindo a largura de banda das placas traseiras dos comutadores que sustentam os usuários na rede. Mesmo no caso de uma rede Gigabit Ethernet, se usuários suficientes possuírem picos suficientes alinhados de uma só vez, isso poderia cobrir a rede ou comutadores de rede. E, uma vez que o vídeo de resolução 2560x1440 se torna mais comum, a taxa de dados de fluxo de vídeo média pode ser 9,5Mbps, resultando talvez em uma taxa de dados de pico de 95Mbps. É evidente que uma conexão de 100Mbps entre um centro de dados e um escritório (que hoje em dia é uma conexão excepcionalmente rápida) poderia ser completamente ocupada pelo tráfego de picos de um único usuário. Assim, mesmo que as LANs e conexões de rede privada possam ser mais tolerantes de vídeo de fluxo com picos, o vídeo de fluxo com altos picos não é desejado e deve exigir planejamento e acomodação especial por um departamento de IT do escritório.

Naturalmente, para aplicações de vídeo lineares padrão, essas questões não são um problema, pois a taxa de dados é "atenuada" no ponto de transmissão e os dados de cada quadro abaixo da taxa de dados disponível máxima 622, e um buffer nos armazenamentos de cliente uma sequência de I, P e B quadros antes de serem descompactados. Assim, a taxa de dados sobre a rede permanece próxima à taxa de dados média do fluxo de vídeo. Infelizmente, isso introduz latência, mesmo que os B-quadros não sejam usados, isso é inaceitável para aplicações de baixa latência como videogames e aplicações que exigem tempo de resposta rápido.

Uma solução da técnica anterior para mitigar os fluxos de vídeo que possuem altos picos é utilizar uma técnica geralmente referida como

codificação de "Taxa de Bits Constante" (CBR). Embora possa parecer que o termo CBR sugere que todos os quadros sejam compactados para possuir a mesma taxa de bits (ou seja, tamanho), o que geralmente refere-se é um paradigma de compactação onde uma taxa de bits máxima sobre um determinado número de quadros (nesse caso, 1 quadro) é permitida. Por exemplo, no caso da figura 6c, se uma restrição de CBR for aplicada à codificação que limitou a taxa de bits, por exemplo, a 70% da taxa de dados máxima classificada 621, então o algoritmo de compactação poderia limitar a compactação de cada quadro de modo que qualquer quadro que poderia ser normalmente compactado utilizando mais de 70% da taxa de dados máxima classificada 621 possa ser compactado com menos bits. O resultado disso é que os quadros que normalmente poderiam exigir mais bits para manter um determinado nível de qualidade poderiam ser "privados" de bits e a qualidade de imagem desses quadros poderia ser pior do que aquele de outros quadros que não exigem mais bits do que 70% da taxa de dados máxima classificada 621. Essa abordagem pode produzir resultados aceitáveis para determinados tipos de vídeo compactado onde há (a) pouco movimento ou alterações de cena são esperadas e (b) os usuários podem aceitar a degradação de qualidade periódica. Um bom exemplo de uma aplicação adequada para CBR é a teleconferência de vídeo visto que há poucos picos, e se a qualidade se degradar brevemente (por exemplo, se a câmera for deslocada, resultando em movimento de cena significativo e grandes picos, durante o deslocamento pode não haver bits suficientes para compactação de imagem de alta qualidade, que poderia resultar em qualidade de imagem degradada), é aceitável para a maioria dos usuários. Infelizmente, a CBR não é bem adequada para muitas outras aplicações que possuem cenas de alta complexidade ou um movimento muito grande e/ou quando um nível de qualidade razoavelmente constante for exigido.

A lógica de compactação de baixa latência 404 empregada em uma modalidade utiliza diversas técnicas diferentes para atender a faixa de problemas com o fluxo de vídeo compactado de baixa latência, enquanto mantém a alta qualidade. Primeiro, a lógica de compactação de baixa latên-

cia 404 gera apenas I-quadros e P-quadros, reduzindo assim a necessidade de esperar vários tempos de quadro para decodificar cada B-quadro. Ademais, como ilustrado na figura 7a, em uma modalidade, a lógica de compactação de baixa latência 404 subdivide cada quadro não compactado 701-760 em uma série de "blocos" e codifica individualmente cada bloco como um I-quadro ou um P-quadro. O grupo de I-quadros e P-quadros compactados é referido aqui como "quadros R" 711-770. No exemplo específico mostrado na figura 7a, cada quadro não compactado é subdividido em uma matriz 4 x 4 de 16 blocos. Entretanto, esses princípios subjacentes não são limitados a nenhum esquema de subdivisão particular.

Em uma modalidade, a lógica de compactação de baixa latência 404 divide o quadro de vídeo em inúmeros blocos, e codifica (ou seja, compacta) um bloco de cada quadro como um I-quadro (ou seja, o bloco é compactado como se fosse um quadro de vídeo separado de 1/16 o tamanho da imagem total, e a compactação usada para esse "mini" quadro é a compactação de I-quadro) e os blocos restantes como P-quadros (ou seja, a compactação usada para cada "mini" $1/16^{th}$ quadro é a compactação de P-quadro). Os blocos compactados como I-quadros e como P-quadros devem ser referidos como "blocos I" e "blocos P", respectivamente. Com cada quadro de vídeo sucessivo, o bloco que será codificado como um I-bloco é alterado. Assim, em um determinado tempo de quadro, apenas um bloco dos blocos no quadro de vídeo é um I-bloco, e o restante dos blocos consiste nos blocos P. Por exemplo, na figura 7a, o bloco 0 de quadro não compactado 701 é codificado como o I-I-tile$_0$ e o restante de 1 a 15 blocos é codificado como blocos P P_1 a P_{15} para produzir o quadro R 711. No próximo quadro de vídeo não compactado 702, o bloco 1 de quadro não compactado 701 é codificado como o I-I-tile$_1$ e os blocos restantes 0 e 2 a 15 são codificados como blocos P, P_0 e P_2 a P_{15}, para produzir o quadro R 712. Assim, os blocos I e os blocos P para os blocos são progressivamente intercalados em tempo sobre os sucessivos quadros. O processo continua até um bloco R 770 ser gerado com o último bloco na matriz codificada como um I-bloco (ou seja, I_{15}). O processo então começa gerando outro quadro R como quadro

711 (ou seja, codificando um I-bloco para o bloco 0) etc. Embora não ilustrado na figura 7a, em uma modalidade, o primeiro quadro R da sequência de vídeo de quadros R contém apenas os blocos (ou seja, de modo que os P-quadros subsequentes possuam dados de imagem de referência a partir dos quais calcula-se o movimento). Alternativamente, em uma modalidade, a sequência de inicialização utiliza o mesmo padrão de I-bloco normal, porém não inclui o blocos P daqueles blocos que ainda não foram codificados com um I-bloco. Em outras palavras, alguns blocos não são codificados com nenhum dado até o primeiro I-bloco chegar, evitando assim os picos de inicialização na taxa de dados de fluxo de vídeo 934 na **figura 9a**, que é explicada em mais detalhes abaixo. Ademais, como descrito abaixo, vários tamanhos e formatos diferentes podem ser usados para os blocos enquanto ainda estão de acordo com os princípios subjacentes.

A lógica de descompactação de vídeo 412 que é executada no cliente 415 descompacta cada bloco como se fosse uma sequência de vídeo separada de quadros pequenos I e P, e então apresenta cada bloco ao dispositivo de exibição de direção de buffer de quadro 422. Por exemplo, I_0 e P_0 de quadros R 711 a 770 são usados para descompactar e apresentar o bloco 0 da imagem de vídeo. Similarmente, I_1 e P_1 de quadros R 711 a 770 são usados para reconstruir o bloco 1, e assim por diante. Como mencionado acima, a descompactação de I-quadros e P-quadros é bem conhecida na técnica, e a descompactação de blocos I e blocos P pode ser realizada com múltiplas instâncias de um descompactador de vídeo que é executado no cliente 415. Embora pareça que os processos de multiplicação aumentam a carga computacional no cliente 415, isso de fato na ocorre devido ao fato de os próprios blocos serem proporcionalmente menores em relação ao número de processos adicionais, então o número de pixels exibido é o mesmo como se houvesse um processo e utilizando quadros I e P de tamanho total convencionais.

Essa técnica de quadro R mitiga significativamente os picos de largura de banda tipicamente associados aos I-quadros ilustrados nas figuras 6b e 6c, pois qualquer determinado quadro é geralmente constituído

de P-quadros que são tipicamente menores do que os I-quadros. Por exemplo, supondo-se novamente que um I-quadro típico seja 160Kb, então os I blocos de cada um dos quadros ilustrados na figura 7a poderiam ser aproximadamente 1/16 desse total ou 10Kb. Similarmente, supondo-se que um P-quadro típico tenha 16 Kb, então os P-quadros de cada um dos blocos ilustrados na figura 7a podem ter aproximadamente 1Kb. O resultado final é um quadro R de aproximadamente 10Kb + 15 * 1Kb = 25Kb. Então, cada sequência de 60 quadros poderia ter 25Kb * 60 = 1,5Mbps. Então, em 60 quadros/segundo, poderia ser exigido um canal capaz de sustentar uma largura de banda de 1,5Mbps, porém com picos muito menores devido ao fato de os I blocos serem distribuídos ao longo do intervalo de 60 quadros.

Nota-se que em exemplos anteriores com as mesmas taxas de dados presumidas para os I-quadros e P-quadros, a taxa de dados média era de 1,1Mbps. Isso porque nos exemplos anteriores, um novo I-quadro foi introduzido somente uma vez a cada 60 tempos de quadro, enquanto nesse exemplo, os 16 blocos que constituem um ciclo de I-quadro em 16 tempos de quadro, e com isso o equivalente de um I-quadro é introduzido a cada 16 tempos de quadro, resultando em uma taxa de dados média ligeiramente maior. Na prática, no entanto, a introdução de quadros I mais frequentes não aumenta a taxa de dados de maneira linear. Isso se deve ao fato de que um P-quadro (ou um bloco P) codifica principalmente a diferença do quadro anterior para o próximo. Então, se o quadro anterior for muito similar ao próximo quadro, o P-quadro será muito pequeno, se o quadro anterior for muito diferente do próximo quadro, o P-quadro será muito grande. Porém, devido ao fato de um P-quadro ser amplamente derivado do quadro anterior, em vez do quadro real, o quadro codificado resultante pode conter mais erros (por exemplo, artefatos visuais) do que um I-quadro com um número adequado de bits. E, quando um P-quadro acompanhar outro P-quadro, o que pode ocorrer é um acúmulo de erros que fica pior quando há uma sequência longa de P-quadros. Agora, um compactador de vídeo sofisticado irá detectar o fato que a qualidade da imagem está se degradando após uma sequência de P-quadros e, se necessário, esse irá alocar mais bits nos P-

quadros subsequentes para criar a qualidade ou, se esse for o curso de ação mais eficiente, substitui-se um P-quadro por um I-quadro. Então, quando sequências de P-quadros longas forem usadas (por exemplo, 59 P-quadros, como nos exemplos anteriores acima) particularmente quando a cena possui uma complexidade e/ou movimento muito grande, tipicamente, mais bits são necessários para os P-quadros à medida que são mais afastados de um I-quadro.

Ou, para considerar os P a partir do ponto de vista oposto, os P-quadros que acompanham estreitamente um I-quadro tendem a exigir menos bits do que os P-quadros que estão mais afastados de um I-quadro. Então, no exemplo mostrado na figura 7a, nenhum P-quadro está afastado mais do que 15 quadros de um I-quadro que precede o mesmo, enquanto no exemplo anterior, um P-quadro poderia estar 59 quadros afastado de um I-quadro. Assim, com I-quadros mais frequentes, os P-quadros são menores. Naturalmente, os tamanhos relativos exatos irão variar com base na natureza do fluxo de vídeo, porém no exemplo da figura 7a, se um bloco possuir 10Kb, os blocos P em média, podem possuir apenas 0,75kb de tamanho, resultando em 10Kb + 15 * 0,75Kb = 21,25Kb, ou em 60 quadros por segundo, a taxa de dados poderia possuir 21,25Kb * 60 = 1,3Mbps, ou taxa de dados cerca de 16% maior do que um fluxo com um I acompanhado por 59 P-quadros em 1,1Mbps. Novamente, os resultados relativos entre essas duas abordagens para a compactação de vídeo irão variar dependendo da sequência de vídeo, porém tipicamente, descobriu-se empiricamente que a utilização de quadros R exige cerca de 20% mais bits para um determinado nível de qualidade do que a utilização de sequências de I-quadro/P. Porém, naturalmente, os quadros R reduzem dramaticamente os picos que tornam as sequências de vídeo utilizáveis com muito menos latência do que as sequências de I-quadro/P.

Os quadros R podem ser configurados em uma variedade de formas diferentes, dependendo da natureza da sequência de vídeo, da confiabilidade do canal, e da taxa de dados disponível. Em uma modalidade alternativa, um número de blocos que não 16 é usado em uma configuração

4x4. Por exemplo, 2 blocos podem ser usados em uma configuração 2x1 ou 1x2, 4 blocos podem ser usados em uma configuração 2x2, 4x1 ou 1x4, 6 blocos podem ser usados em uma configuração 3x2, 2x3, 6x1 ou 1x6 ou 8 blocos podem ser usados em uma configuração 4x2 (como mostrado na figura 7b), 2x4, 8x1 ou 1x8. Nota-se que os blocos não precisam ser quadrados, nem o quadro de vídeo deve ser quadrado, ou ainda retangular. Os blocos podem ser divididos em qualquer formato que melhor se ajustar ao fluxo de vídeo e à aplicação usada.

Em outra modalidade, o ciclo dos blocos I e P não é limitado ao número de blocos. Por exemplo, em uma configuração 4x2 de 8 blocos, uma sequência de 16 ciclos ainda pode ser usada como ilustrado na figura 7b. Os quadros não compactados sequenciais 721, 722, 723 são divididos em 8 blocos, 0 a 7 e cada bloco é individualmente compactado. A partir do quadro R 731, apenas o bloco 0 é compactado como um I-bloco, e os blocos restantes são compactados como P blocos. Para o quadro R subsequente 732 todos os 8 blocos são compactados como P blocos, e então para o quadro R subsequente 733, o bloco 1 é compactado como um I-bloco e os outros blocos são compactados como P blocos. E, então o seqüenciamento continua durante 16 quadros, com um I-bloco gerado apenas a cada outro quadro, então o último I-bloco é gerado para o bloco 7 durante o 15° tempo de quadro (não mostrado na figura 7b) e durante o 16° tempo de quadro R o quadro 780 é compactado utilizando todos os P blocos. Então, a sequência começa novamente com o bloco 0 compactado como um I-bloco e os outros blocos compactados como P blocos. Como na modalidade anterior, o primeiro quadro de toda a sequência de vídeo poderia ser tipicamente todos os I blocos, para fornecer uma referência para P blocos daquele ponto em diante. O ciclo dos I blocos e P blocos ainda não precisa ser um múltiplo par do número de blocos. Por exemplo, com 8 blocos, cada quadro com um I-bloco pode ser acompanhado por 2 quadros com todos os P blocos, antes de outro I-bloco ser usado. Ainda em outra modalidade, alguns blocos podem ser seqüenciados com blocos mais frequentemente do que outros blocos se, por exemplo, considera-se que determinadas áreas da tela exigem

mais movimento de I blocos frequentes, enquanto outras são mais estáticas (por exemplo, mostrando uma pontuação de um jogo) exigindo blocos menos frequentes. Ademais, embora cada I-quadro seja ilustrado nas figuras 7a-b como um único I-bloco, múltiplos I blocos podem ser codificados em um único quadro (dependendo da largura de banda do canal de transmissão). Em contrapartida, determinados quadros ou sequências de quadro podem ser transmitidos sem os I blocos (ou seja, apenas os P blocos).

O motivo no qual as abordagens do parágrafo anterior funcionam bem é que embora não tenha I blocos distribuídos para cada quadro único, pode parecer que resulte em picos maiores, o comportamento do sistema não é tão simples. Visto que cada bloco é compactado separadamente dos outros blocos, à medida que os blocos ficam menores, a codificação de bloco pode se tornar menos eficiente, pois o compactador de um determinado bloco não é capaz de explorar características de imagem similares e movimento similar dos outros blocos. Assim, a divisão da tela em 16 blocos geralmente irá resultar em uma codificação menos eficiente do que a divisão da tela em 8 blocos. Porém, se a tela for dividida em 8 blocos e isso fizer com que os dados de um quadro total I sejam introduzidos a cada 8 quadros em vez de a cada 16 quadros, irá resultar em um total de taxa de dados muito maior. Então, ao introduzir um quadro total I a cada 16 quadros em vez de a cada 8 quadros, a taxa de dados total é reduzida. Também, ao utilizar 8 blocos maiores em vez de 16 blocos menores, a taxa de dados total é reduzida, isso também mitiga até determinado ponto os picos de dados causados pelos blocos maiores.

Em outra modalidade, a lógica de compactação de vídeo de baixa latência 404 nas figuras 7a e 7b controla a alocação de bits nos vários blocos nos quadros R mediante pré-configuração por configurações, com base em características conhecidas da sequência de vídeo que será compactada, ou automaticamente, com base em uma análise contínua da qualidade de imagem em cada bloco. Por exemplo, em alguns jogos de corrida, a frente do carro do jogador (que está relativamente estática na cena) ocupa uma grande parte da metade inferior da tela, enquanto a metade superior da

tela é completamente preenchida com a estrada em sentido contrário, edifícios e paisagem, que quase sempre está em movimento. Se a lógica de compactação 404 alocar um número igual de bits em cada bloco, então os blocos sobre a metade inferior da tela (blocos 4 a 7) no quadro não compactado 721 na figura 7b, serão geralmente compactados com qualidade superior aos blocos na metade superior da tela (blocos 0 a 3) no quadro não compactado 721 na figura 7b. Se for considerado que esse jogo particular, ou essa cena particular do jogo possui tais características, então os operadores do serviço de hospedagem 210 podem configurar a lógica de compactação 404 para alocar mais bits nos blocos na parte superior da tela do que nos blocos na parte inferior da tela. Ou, a lógica de compactação 404 pode avaliar a qualidade da compactação dos blocos após os quadros serem compactados (utilizando uma ou mais das várias métricas de qualidade de compactação, como Razão Máxima Entre Sinal-Ruído (PSNR)) e se for determinado que sobre uma determinada janela de tempo, alguns blocos estão consistentemente produzindo melhores resultados de qualidade, então essa aloca gradualmente mais bits nos blocos que estão produzindo resultados de qualidade inferior, até os vários blocos atingirem um nível de qualidade similar. Em uma modalidade alternativa, a lógica de compactador 404 aloca bits para atingir uma qualidade superior em um bloco particular ou grupo de blocos. Por exemplo, a mesma pode proporcionar uma melhor aparência perceptual geral para possuir maior qualidade no centro da tela do que nas bordas.

Em uma modalidade, para aprimorar a resolução de algumas regiões do fluxo de vídeo, a lógica de compactação de vídeo 404 utiliza blocos menores para codificar áreas do fluxo de vídeo com complexidade e/ou movimento de cena relativamente maior do que as áreas do fluxo de vídeo com complexidade e/ou movimento de cena relativamente menor. Por exemplo, como ilustrado na figura 8, blocos menores são empregados em torno de um personagem em movimento 805 em uma área de um quadro R 811 (potencialmente acompanhada por uma série de quadros R com os mesmos tamanhos de bloco (não mostrados)). Então, quando o personagem

805 se move para uma nova área da imagem, blocos menores são usados em torno dessa nova área dentro de outro quadro R 812, como ilustrado. Como mencionado acima, vários tamanhos e formatos diferentes podem ser empregados como "blocos" enquanto ainda estão de acordo com esses princípios subjacentes.

Embora os blocos I/P cíclicos descritos acima reduzam substancialmente os picos na taxa de dados de um fluxo de vídeo, esses não eliminam os picos completamente, particularmente no caso de imagem de vídeo altamente complexa ou que muda rapidamente, como ocorre com imagens em movimento, videogames, e algum software de aplicação. Por exemplo, durante uma transição repentina de cena, um quadro complexo pode ser acompanhado por outro quadro complexo que é completamente diferente. Mesmo que vários blocos I possam ter precedido a transição de cena em apenas alguns tempos de quadro, esses não ajudam nessa situação, pois o material do novo quadro não tem relação com os blocos I anteriores. Em tal situação (e em outras situações em que mesmo que nada mude, grande parte da imagem muda), o compactador de vídeo 404 irá determinar que muitos, se não todos, os blocos P são codificados de maneira mais eficiente que os blocos I, e o resultado é um pico muito grande na taxa de dados daquele quadro.

Como anteriormente discutido, é simplesmente o caso que com a maior parte das conexões de Internet a nível de consumidor (e muitas conexões de escritório), simplesmente não é viável "obstruir" dados que excedem a taxa de dados máxima disponível mostrada como 622 na figura 6c, juntamente com a taxa de dados máxima classificada 621. Nota-se que a taxa de dados máxima classificada 621 (por exemplo, "DSL de 6Mbps") é essencialmente um número de marketing para usuários considerando a aquisição de uma conexão de Internet, porém geralmente não garante um nível de desempenho. Para os propósitos dessa aplicação, é irrelevante, visto que a única preocupação é a taxa de dados máxima disponível 622 no momento em que o vídeo é transmitido através da conexão. Consequentemente, nas figuras 9a e 9c, como descrito aqui uma solução para o proble-

ma de formação de pico, a taxa de dados máxima classificada é omitida do gráfico, e apenas a taxa de dados máxima disponível 922 é mostrada. A taxa de dados de fluxo de vídeo não deve exceder a taxa de dados máxima disponível 922.

Para atender isso, a primeira coisa que o compactador de vídeo 404 faz é determinar uma taxa de dados de pico 941, que é uma taxa de dados que o canal é capaz de manipular constantemente. Essa taxa pode ser determinada por inúmeras técnicas. Uma tal técnica é enviar gradualmente um fluxo de teste de taxa de dados cada vez maior do serviço de hospedagem 210 para o cliente 415 nas figuras 4a e 4b, e fazer com que o cliente forneça um retorno ao serviço de hospedagem para o nível de perda de e latência. À medida que a perda de pacote e/ou latência começa a mostrar um forte aumento, que é uma indicação que a taxa de dados máxima disponível 922 está sendo atingida. Após isso, o serviço de hospedagem 210 pode reduzir gradualmente a taxa de dados do fluxo de teste até o cliente 415 relatar que um período de tempo razoável no qual o fluxo de teste foi recebido com um nível de perda de pacote e latência aceitável é quase mínimo. Isso estabelece uma taxa de dados máxima de pico 941, que será então usada como uma taxa de dados de pico para transmitir o vídeo. Ao longo do tempo, a taxa de dados de pico 941 irá flutuar (por exemplo, se outro usuário em uma família começar a usar excessivamente a conexão de Internet), e o cliente 415 precisará monitorar constantemente o mesmo para observar se a perda de pacote ou latência aumentou, indicando que a taxa de dados máxima disponível 922 está caindo abaixo da taxa de dados de pico anteriormente estabelecida 941, e nesse caso, a taxa de dados de pico 941. Similarmente, se ao longo do tempo o cliente 415 notar que a perda de pacote e latência permanecem em níveis ótimos, o mesmo pode solicitar que o compactador de vídeo aumenta lentamente a taxa de dados para observar se a taxa de dados máxima disponível aumentou (por exemplo, se outro usuário em uma família parou de usar excessivamente a conexão de Internet), e novamente aguarda até a perda de pacote e/ou latência maior indicar que a taxa de dados máxima disponível 922 foi excedida, e nova-

mente um nível inferior pode ser encontrado para a taxa de dados de pico 941, porém há um que talvez seja maior do que o nível antes de testar uma taxa de dados aumentada. Então, ao utilizar essa técnica (e outras técnicas como essa) uma taxa de dados de pico 941 pode encontrada, e ajustada periodicamente quando necessário. A taxa de dados de pico 941 estabelece a taxa de dados máxima que pode ser usada pelo compactador de vídeo 404 para o fluxo de vídeo até o usuário. A lógica para determinar a taxa de dados de pico pode ser implementada nas premissas de usuário 211 e/ou no serviço de hospedagem 210. Nas premissas de usuário 211, o dispositivo de cliente 415 realiza os cálculos para determinar a taxa de dados de pico e transmitir essas informações novamente para o serviço de hospedagem 210; no serviço de hospedagem 210, um servidor 402 no serviço de hospedagem realiza os cálculos para determinar a taxa de dados de pico com base nas estatísticas recebidas do cliente 415 (por exemplo, perda de pacote, latência, taxa de dados máxima, etc).

A figura 9a mostra um exemplo de taxa de dados de fluxo de vídeo 934 que possui complexidade e/ou movimento de cena substancial que foi gerada utilizando as técnicas compactação de I/P bloco cíclico anteriormente descritas e ilustradas nas figuras 7a, 7b e 8. O compactador de vídeo 404 foi configurado para emitir o vídeo compactado em uma taxa de dados média que está abaixo da taxa de dados de pico 941, e nota-se que, na maior parte do tempo, a taxa de dados de fluxo de vídeo permanece abaixo da taxa de dados de pico 941. Uma comparação de taxa de dados 934 com a taxa de dados de fluxo de vídeo 634 mostrada na figura 6c criada utilizando I/P/B ou I/ P quadros mostra que a compactação de I/P bloco cíclico produz uma taxa de dados muito mais suave. Ainda, no quadro 2x pico 952 (que é aproximadamente 2x a taxa de dados de pico 942) e no quadro 4x pico 954 (que é aproximadamente 4x a taxa de dados de pico 944), a taxa de dados excede a taxa de dados de pico 941, que é inaceitável. Na prática, mesmo com um vídeo de alta ação de videogames que mudam rapidamente, os picos que ultrapassam a taxa de dados de pico 941 ocorrem em menos de 2% dos quadros, os picos que ultrapassam 2x a taxa

de dados de pico 942 ocorrem raramente, e os picos que ultrapassam 3x a taxa de dados de pico 943 ocorrem quase nunca. Porém, quando esses ocorrem (por exemplo, durante uma transição de cena), a taxa de dados exigida por esses é necessária para produzir uma imagem de vídeo de qualidade satisfatória.

Uma maneira para resolver esse problema é simplesmente configurar o compactador de vídeo 404 de modo que sua saída de taxa de dados máxima seja a taxa de dados de pico 941. Infelizmente, a qualidade de saída de vídeo resultante durante os quadros de pico é insatisfatória visto que o algoritmo de compactação é "privado" de bits. O resultado é o surgimento de artefatos de compactação quando há transições repentinas ou movimento rápido, e a tempo, o usuário vem a perceber que os artefatos sempre surgem quando há mudanças repentinas ou movimento rápido, e esses se tornam um tanto desagradáveis.

Embora o sistema visual humano seja muito sensível a artefatos visuais que aparecem durante mudanças repentinas ou movimento rápido, o mesmo não é muito sensível à detecção de uma redução na taxa de quadro em tais situações. Na verdade, quando tais mudanças repentinas ocorrerem, parece que o sistema visual humano está preocupado em rastrear as mudanças, e não percebe se a taxa de quadro cai brevemente de 60fps para 30fps, e então retorna imediatamente para 60fps. E, no caso de uma transição muito dramática, como uma mudança de cena repentina, o sistema visual humano não percebe se a taxa de quadro cai para 20fps ou ainda 15fps, e então retorna imediatamente para 60fps. Desde que a redução de taxa de quadro ocorra apenas raramente, para um observador humano, parece que o vídeo está sendo continuamente executado em 60fps.

Essa propriedade do sistema visual humano é explorada pelas técnicas ilustradas na figura 9b. Um servidor 402 (das figuras 4a e 4b) produz um fluxo de saída de vídeo não compactado em uma taxa de quadro constante (em 60fps em uma modalidade). Uma linha de tempo mostra cada saída de quadro 961-970 a cada 1/60 segundo. Cada quadro de vídeo não compactado, começando com o quadro 961, é emitido para o compactador

de vídeo de baixa latência 404, que compacta o quadro em menos de um tempo de quadro, produzindo para o primeiro quadro um quadro compactado 1 981. Os dados produzidos para o quadro compactado 1 981 podem ser maiores ou menores, dependendo de muitos fatores, como anteriormente descrito. Se os dados forem pequenos o suficiente para que possam ser transmitidos para o cliente 415 em um tempo de quadro (1/60 segundo) ou menos na taxa de dados de pico 941, então esses são transmitidos durante o tempo de transmissão (tempo xmit) 991 (o comprimento da seta indica a duração do tempo de transmissão). No próximo tempo de quadro, o servidor 402 produz um quadro não compactado 2 962, o mesmo é compactado para compactar o quadro 2 982, e o mesmo é transmitido para o cliente 415 durante o tempo de transmissão 992, que é menor do que um tempo de quadro na taxa de dados de pico 941.

Então, no próximo tempo de quadro, o servidor 402 produz um quadro não compactado 3 963. Quando o mesmo for compactado pelo compactador de vídeo 404, o quadro compactado resultante 3 983 possui mais dados que podem ser transmitidos na taxa de dados de pico 941 em um tempo de quadro. Então, o mesmo é transmitido durante o tempo de transmissão (2x pico) 993, que refere-se a todo o tempo de quadro time e parte do próximo tempo de quadro. Agora, durante o próximo tempo de quadro, o servidor 402 produz outro quadro não compactado 4 964 e emite o mesmo para o compactador de vídeo 404, porém os dados são ignorados e ilustrados com 974. Isso se deve ao fato de o compactador de vídeo 404 ser configurado para ignorar os quadros de vídeo adicionalmente não compactados que chegam enquanto ainda estão transmitindo um quadro compactado anterior. Naturalmente, o descompactador de vídeo do cliente 415 não conseguirá receber o quadro 4, porém o mesmo continua simplesmente a exibir no dispositivo de exibição 422 o quadro 3 para 2 tempos de quadro (ou seja, reduz brevemente a taxa de quadro de 60fps para 30fps).

Para o próximo quadro 5, o servidor 402 emite o quadro não compactado 5 965, é compactado no quadro compactado 5 985 e transmitido dentro de 1 quadro durante o tempo de transmissão 995. O descom-

pactador de vídeo do cliente 415 descompacta o quadro 5 e exibe o mesmo no dispositivo de exibição 422. Depois, o servidor 402 emite o quadro não compactado 6 966, o compactador de vídeo 404 compacta o mesmo no quadro compactado 6 986, porém desta vez os dados resultantes são muito grandes. Os I-quadros compactados são transmitidos durante o tempo de transmissão (4x o pico) 996 na taxa de dados de pico 941, porém leva quase 4 tempos de quadro para transmitir o quadro. Durante os próximos 3 tempos de quadro, o compactador de vídeo 404 ignora 3 quadros do servidor 402, e o descompactador do cliente 415 mantém o quadro 6 estacionário no dispositivo de exibição 422 para 4 tempos de tempos de quadro (ou seja, reduz brevemente a taxa de quadro de 60fps para 15fps). Então por fim, o servidor 402 emite o quadro 10 970, o compactador de vídeo 404 compacta o mesmo no quadro compactado 10 987, e é transmitido durante o tempo de transmissão 997, e o descompactador do cliente 415 descompacta o quadro 10 e exibe o mesmo no dispositivo de exibição 422 e novamente o vídeo reinicia em 60fps.

Nota-se que embora o compactador de vídeo 404 ignore quadros de vídeo do fluxo de vídeo gerado pelo servidor 402, o mesmo não remove dados de áudio, independente de qual áudio são originados, e continua a compactar os dados de áudio quando os quadros de vídeo forem ignorados e transmite os mesmos ao cliente 415, que continua a descompactar os dados de áudio e fornecer o áudio para qualquer dispositivo que for usado pelo usuário para reproduzir o áudio. Assim, o áudio continua no mesmo ritmo durante períodos quando os quadros são ignorados. O áudio compactado consome uma porcentagem relativamente pequena de largura de banda, comparado com o vídeo compactado, e como resultado não possui um impacto importante sobre a taxa de dados total. Embora não seja ilustrado em nenhum diagrama de taxa de dados, há sempre uma capacidade de taxa de dados reservada para o fluxo de áudio compactado dentro da taxa de dados de pico 941.

O exemplo descrito na figura 9b foi selecionado para ilustrar como a taxa de quadro cai durante picos de taxa de dados, porém não é

ilustrado que quando as técnicas de I/P bloco cíclico anteriormente descritas forem usadas, tais picos de taxa de dados, e os quadros ignorados consequenciais forem raros, mesmo durante sequências de alta complexidade/ação de cena como aquelas que ocorrem em videogames, imagens em movimento e algum software de aplicação. Consequentemente, as taxas de quadro reduzidas são pouco frequentes e breves, e o sistema visual humano não as detecta.

Se o mecanismo de redução de taxa de quadro descrito acima for aplicado à taxa de dados de fluxo de vídeo ilustrada na figura 9a, a taxa de dados de fluxo de vídeo resultante é ilustrada na figura 9c. Nesse exemplo, o pico 2x 952 foi reduzido para pico nivelado 2x 953, e pico 4x 955 foi reduzido pico 4x nivelado 955, e toda a taxa de dados de fluxo de vídeo 934 permanece na ou abaixo da taxa de dados de pico 941.

Assim, utilizando-se as técnicas descritas acima, um fluxo de vídeo de alta ação pode ser transmitido com baixa latência através da Internet geral e através de uma conexão de Internet a nível de consumidor. Ademais, em um ambiente de escritório em uma LAN (por exemplo, 100Mbs Ethernet ou 802.11g sem fio) ou em uma rede privada (por exemplo, conexão de 100Mbps entre um centro de dados e um escritório) um fluxo de vídeo de alta ação pode ser transmitido sem picos de modo que múltiplos usuários (por exemplo, transmitindo 1920x1080 em 60fps em 4,5Mbps) possa utilizar a LAN ou conexão de dados privada compartilhada sem ter picos sobrepostos cobrindo a rede ou as placas traseiras dos comutadores de rede.

AJUSTE DE TAXA DE DADOS

Em uma modalidade, o serviço de hospedagem 210 avalia inicialmente a taxa de dados máxima disponível 622 e a latência do canal para determinar uma taxa de dados apropriada para o fluxo de vídeo e então em resposta ajusta dinamicamente a taxa de dados. Para ajustar a taxa de dados, o serviço de hospedagem 210 pode, por exemplo, modificar a resolução de imagem e/ou o número de quadros/segundo do fluxo de vídeo que será enviado para o cliente 415. Também, o serviço de hospedagem pode

ajustar o nível de qualidade do vídeo compactado. Quando se altera a resolução do fluxo de vídeo, por exemplo, de uma resolução 1280 x 720 para 640 x 360, a lógica de descompactação de vídeo 412 no cliente 415 pode ampliar a escala da imagem para manter o mesmo tamanho de imagem na tela de exibição.

Em uma modalidade, em uma situação onde o canal foi completamente abandonado, o serviço de hospedagem 210 pausa o jogo. No caso de um jogo para vários jogadores, o serviço de hospedagem relata aos outros usuários que o usuário abandonou o jogo e/ou pausa o jogo para os outros usuários.

PACOTES IGNORADOS OU ATRASADOS

Em uma modalidade, se os dados forem perdidos devido à perda de pacote entre o compactador de vídeo 404 e o cliente 415 nas figuras 4a ou 4b, ou devido ao fato de um pacote ser recebido fora de ordem que chega muito tarde para ser descompactado e cumpre as exigências de latência do quadro descompactado, a lógica de descompactação de vídeo 412 é capaz de mitigar os artefatos visuais. Em uma implementação de I/P-quadro de fluxo, se houver um pacote perdido/atrasado, toda a tela sofre o impacto, fazendo potencialmente com que a tela congele completamente durante um período de tempo ou mostre outros artefatos visuais em tela plana. Por exemplo, se um pacote perdido/atrasado causar a perda de um I-quadro, então o descompactador não irá dispor de uma referência para todos os P-quadros que acompanham até um novo I-quadro ser recebido. Se um P-quadro for perdido, então causará um impacto sobre os P-quadros para toda a tela que os acompanha. Dependendo de quanto tempo levará antes de um I-quadro surgir, isso causará um impacto visual maior ou menor. Utilizando-se I/P blocos intercalados como mostrado nas figuras 7a e 7b, é muito menos provável que um pacote perdido/atrasado cause impacto sobre toda a tela visto que isso afetará apenas os blocos contidos no pacote afetado. Se cada dado do bloco for enviado dentro de um pacote individual, então se um pacote for perdido, isso afetará apenas um bloco. Naturalmente, a duração do artefato visual irá depender da possibilidade de um pacote

de I bloco ser perdido e, se um P bloco for perdido, quantos I-quadros levarão até um I bloco aparecer. Porém, uma vez que os blocos diferentes na tela estão sendo atualizados com I-quadros muito frequentemente (potencialmente a cada quadro), mesmo que um bloco na tela seja afetado, outros blocos podem não ser. Ademais, se algum evento causar uma perda de vários pacotes de uma só vez (por exemplo, um pico de energia próximo a uma linha DSL que interrompe brevemente o fluxo de dados), então alguns blocos serão afetados mais do que outros, porém devido ao fato de alguns blocos serem rapidamente renovados com um novo I bloco, esses serão apenas brevemente afetados. Também, com uma implementação de I/P-quadro de fluxo, não só I-quadros são mais os I quadros mais críticos, porém os I-quadros são extremamente grandes, então se houver um evento que origine um pacote ignorado/atrasado, há uma probabilidade maior que um I-quadro seja afetado (ou seja, se qualquer parte de um I-quadro for perdida, é improvável que o I-quadro possa ser descompactado) do que um I bloco muito menor. Devido a todos esses motivos, a utilização de I/P blocos resulta em muito menos artefatos visuais quando os pacotes forem ignorados/atrasados do que com I/P-quadros.

Uma modalidade tenta reduzir o efeito de pacotes perdidos ao armazenar de forma inteligente os blocos compactados dentro dos pacotes TCP (protocolo de controle de transmissão) ou pacotes UDP (protocolo de datagrama de usuário). Por exemplo, em uma modalidade, os blocos são alinhados com os limites de pacote sempre que possível. A figura 10a ilustra como os blocos devem ser acondicionados dentro de uma série de pacotes 1001-1005 sem implementar esse recurso. Especificamente, na figura 10a, os blocos cruzam os limites de pacote e são armazenados de maneira ineficiente de modo que a perda de um único pacote resulta na perda de múltiplos quadros. Por exemplo, se os pacotes 1003 ou 1004 forem perdidos, três blocos são perdidos, resultando em artefatos visuais.

Em contrapartida, a figura 10b ilustra a lógica de armazenamento de bloco 1010 para armazenar os blocos de maneira inteligente dentro de pacotes para reduzir o efeito de perda de pacote. Primeiro, a lógica de arma-

zenamento de bloco 1010 alinha os blocos com os limites de pacote. Assim, os blocos T1, T3, T4, T7, e T2 são alinhados com os limites de pacotes 1001-1005, respectivamente. A lógica de armazenamento de bloco também tenta ajustar os blocos dentro de pacotes da maneira mais eficiente possível, sem cruzar os limites de pacote. Com base no tamanho de cada bloco, os blocos T1 e T6 são combinados em um pacote 1001; T3 e T5 são combinados em um pacote 1002; os blocos T4 e T8 são combinados em um pacote 1003; o bloco T8 é adicionado ao pacote 1004; e o bloco T2 é adicionado ao pacote 1005. Assim, sob esse esquema, uma única perda de pacote irá resultar na perda de não mais que 2 blocos (em vez de 3 blocos como ilustrado na figura 10a).

Um benefício adicional da modalidade mostrada na figura 10b é que os blocos são transmitidos em uma ordem diferente em que esses são exibidos dentro da imagem. Desse modo, se os pacotes adjacentes forem perdidos a partir do mesmo evento interferindo na transmissão, isso irá afetar áreas que não estão próximas umas às outras na tela, criando um artefato menos visível no monitor.

Uma modalidade emprega técnicas de correção antecipada de erros (FEC) para proteger determinadas partes do fluxo de vídeo contra erros de canal. Conforme conhecido na técnica, as técnicas FEC como Reed-Solomon e Viterbi geram e anexam informações de dados de correção de erros a dados transmitidos através de um canal de comunicação. Se ocorrer um erro nos dados subjacentes (por exemplo, um I-quadro), então a FEC pode ser usada para corrigir o erro.

Os códigos FEC aumentam a taxa de dados da transmissão, então de modo ideal, esses são apenas usados quando for necessário. Se os dados que estão sendo enviados não pudessem resultar em um artefato visual muito visível, pode ser preferido não utilizar códigos FEC para proteger os dados. Por exemplo, um P bloco que precede imediatamente um I bloco que foi perdido irá criar apenas um artefato visual (ou seja, o bloco na tela não será atualizado) para 1/60 de segundo na tela. Tal artefato visual é dificilmente detectável pelo olho humano. Visto que os P blocos estão mais

atrás de um I bloco, a perda de um P bloco se torna cada vez mais visível. Por exemplo, se um padrão de ciclo de bloco for um I bloco seguido por 15 P blocos antes de um I bloco estar disponível novamente, então se o P bloco que acompanha imediatamente um I bloco for perdido, isso irá resultar no fato de que o bloco mostra uma imagem incorreta durante 15 tempos de quadro (em 60 fps, que poderiam ser 250ms). O olho humano irá detectar facilmente uma interrupção em um fluxo durante 250ms. Então, um P bloco mais atrás pertence a um novo I bloco (ou seja, quanto mais próximo um P bloco acompanha um I bloco), mais visível será o artefato. Como anteriormente discutido, embora, em geral, quanto mais próximo um P bloco acompanha um I bloco, menores serão os dados para aquele P bloco. Assim, os P blocos que acompanham os I blocos não só são mais críticos de se proteger contra perda, como também são menores em tamanho. E, em geral, quanto menores forem os dados que precisam ser protegidos, menor será o código FEC que precisa ser protegido.

Então, como ilustrado na figura 11a, em uma modalidade, devido à importância de I blocos no fluxo de vídeo, apenas os I blocos são fornecidos com códigos FEC. Assim, a FEC 1101 contém um código de correção de erros para I bloco 1100 e a FEC 1104 contém um código de correção de erros para I bloco 1103. Nessa modalidade, nenhuma FEC é gerada para os P blocos.

Em uma modalidade ilustrada na figura 11b, os códigos FEC também são gerados para P que são mais prováveis de causar artefatos visuais se forem perdidos. Nessa modalidade, as FECs 1105 fornecem códigos de correção de erros para os primeiros 3 P blocos, porém não para os P que acompanham. Em outra modalidade, os códigos FEC são gerados para P blocos que são menores em tamanho de dado (que tenderá a autoselecionar os P blocos que ocorrem logo após um I bloco, que são os mais críticos de se proteger).

Em outra modalidade, em vez de enviar um código FEC com um bloco, o bloco é transmitido duas vezes, cada vez em um pacote diferente. Se um pacote for perdido/atrasado, o outro pacote é usado.

Em uma modalidade, mostrada na figura 11c, os códigos FEC 1111 e 1113 são gerados para pacotes de áudio, 1110 e 1112, respectivamente, transmitidos do serviço de hospedagem simultaneamente com o vídeo. É particularmente importante manter a integridade do áudio em um fluxo de vídeo, pois o áudio distorcido (por exemplo, ruído impulsivo ou ruído de fundo) irá resultar em uma experiência particularmente indesejada de usuário. Os códigos FEC ajudam a garantir que o conteúdo de áudio seja apresentado no computador de cliente 415 sem distorção.

Em outra modalidade, em vez de enviar um código FEC com dados de áudio, os dados de áudio são transmitidos duas vezes, cada vez em um pacote diferente. Se um pacote for perdido/atrasado, o outro pacote é usado.

Ademais, em uma modalidade ilustrada na figura 11d, os códigos FEC 1121 e 1123 são usados para comandos de entrada de usuário 1120 e 1122, respectivamente (por exemplo, pressionamentos de botão) transmitidos a montante do cliente 415 para o serviço de hospedagem 210. Isso é importante, pois a ausência de um pressionamento de botão ou um movimento de mouse em um videogame ou um aplicativo poderia resultar em uma experiência de usuário indesejada.

Em outra modalidade, em vez de enviar um código FEC com dados de comando de entrada de usuário, os dados de comando de entrada de usuário são transmitidos duas vezes, cada vez em um pacote diferente. Se um pacote for perdido/atrasado, o outro pacote é usado.

Em uma modalidade, o serviço de hospedagem 210 avalia a qualidade do canal de comunicação com o cliente 415 para determinar se deve-se utilizar FEC e, se for o caso, quais as partes dos comandos de vídeo, áudio e usuário às quais a FEC deve ser aplicada. A avaliação da "qualidade" do canal pode incluir funções como avaliação de perda de pacote, latência, etc, como descrito acima. Se o canal não for particularmente confiável, então o serviço de hospedagem 210 pode aplicar FEC a todos os I blocos, P blocos, comandos de áudio e usuário. Em contrapartida, se o canal for confiável, então o serviço de hospedagem 210 pode aplicar FEC apenas

a comandos de áudio e usuário, ou pode não aplicar FEC a áudio ou vídeo, ou pode não utilizar FEC de modo algum. Várias outras permutações da aplicação de FEC podem ser empregadas enquanto ainda estão de acordo com os princípios subjacentes. Em uma modalidade, o serviço de hospedagem 210 monitora continuamente as condições do canal e, consequentemente, altera a política de FEC.

Em outra modalidade, com referência às figuras 4a e 4b, quando um pacote for perdido/atrasado resultando na perda de dados de bloco ou se, talvez devido a uma perda de pacote particularmente prejudicial, a FEC é incapaz de corrigir os dados de bloco perdidos, o cliente 415 avalia quantos quadros são deixados antes de um novo I bloco ser recebido e compara o mesmo com a latência de ida e volta do cliente 415 do serviço de hospedagem 210. Se a latência de ida e volta for menor do que o número de quadros antes de um novo I bloco estar para chegar, então o cliente 415 envia uma mensagem ao serviço de hospedagem 210 solicitando um novo I bloco. Essa mensagem é roteada até o compactador de vídeo 404, e em vez de gerar um P bloco para o bloco cujos dados foram perdidos, essa gera um I bloco. Visto que o sistema mostrado nas figuras 4a e 4b é projetado para fornecer uma latência de ida e volta que é tipicamente menor que 80ms, isso resulta em um bloco que está sendo corrigido dentro de 80ms (em 60fps, os quadros possuem 16,67ms de duração, assim, em tempos de quadro totais, 80ms de latência poderiam resultar em um bloco corrigido dentro de 83,33ms, que é 5 tempos de quadro - uma interrupção visível, porém muito menos visível do que, por exemplo, uma interrupção de 250ms durante 15 quadros). Quando o compactador 404 gerar tal I bloco fora de sua ordem cíclica comum, se o I bloco puder fazer com que a largura de banda daquele quadro exceda a largura de banda disponível, então o compactador 404 irá atrasar os ciclos dos outros blocos de modo que os outros blocos recebam P blocos durante tal tempo de quadro (mesmo que um bloco se deva normalmente a um I bloco durante aquele quadro), e então começando com o próximo quadro o ciclo comum irá continuar, e o bloco que normalmente poderia receber um I bloco no quadro anterior irá receber um I bloco. Embora essa

ação atrase brevemente a fase do ciclo de R quadro, isso não será normalmente visualmente evidente.

IMPLEMENTAÇÃO DE COMPACTADOR/DESCOMPACTADOR DE VÍDEO E ÁUDIO

A figura 12 ilustra uma modalidade particular em que um multinúcleo e/ou multiprocessador 1200 é usado para compactar 8 blocos em paralelo. Em uma modalidade, um processador dual, um sistema de computador quad core Xeon CPU que executa em 2,66 GHz ou mais é usado, com cada núcleo implementando o compactador de fonte aberta x264 H.264 como um processo independente. Entretanto, várias outras configurações de hardware/software podem ser usadas enquanto ainda estão de acordo com esses princípios subjacentes. Por exemplo, cada núcleo de CPU pode ser substituído por um compactador H.264 implementado em um FPGA. No exemplo mostrado na figura 12, os núcleos 1201-1208 são usados para processar simultaneamente os I blocos e P blocos como oito segmentos independentes. Como conhecido na técnica, os sistemas atuais de multinúcleo e multiprocessador são inerentemente capazes de multi-segmentação quando integrados com sistemas de operação de multi-segmentação como Microsoft Windows XP Professional Edition (edição de 64 bits ou 32 bits) e Linux.

Na modalidade ilustrada na figura 12, visto que cada um dos 8 núcleos é responsável por apenas um bloco, o mesmo opera independentemente de outros núcleos, cada um executando uma instanciação separada de x264. Uma placa de captura DVI baseada em PCI Express x1, como Sendero Video Imaging IP Development Board de Microtronix de Oosterhout, The Netherlands é usado para capturar o vídeo não compactado em resolução 640x480, 800x600, ou 1280x720, e FPGA na placa utiliza Acesso de Memória Direto (DMA) para transferir o vídeo capturado através do barramento DVI no sistema RAM. Os blocos são dispostos em uma disposição 4x2 1205 (embora esses sejam ilustrados como blocos quadrados, nessa modalidade esses são resolução 160x240). Cada instanciação de x264 é configurada para compactar um dos 8 160x240, e esses são sincronizados de modo que, após uma compactação de I bloco inicial, cada núcleo entre

em um ciclo, cada quadro fora de fase com o outro, para compactar um I bloco acompanhado por sete P blocos, e ilustrado na figura 12.

Cada tempo de quadro, os blocos compactados resultantes são combinados em um fluxo de pacote, utilizando as técnicas anteriormente descritas, e então os blocos compactados são transmitidos para um cliente de destino 415.

Embora não ilustrado na figura 12, se a taxa de dados dos 8 blocos combinados exceder uma taxa de dados de pico 941 especificada, então todos os processos 8 x264 são suspensos por quantos tempos de quadro forem necessários até os dados dos 8 blocos combinados serem transmitidos.

Em uma modalidade, o cliente 415 é implementado como software em um PC que executa 8 instanciações de FFmpeg. Um processo de recepção recebe os 8 blocos, e cada bloco é roteado até uma instanciação FFmpeg, que descompacta o bloco e apresenta o mesmo a um local de bloco adequado no dispositivo de exibição 422.

O cliente 415 recebe entrada de teclado, mouse, ou controlador de jogo dos drivers de dispositivo de entrada do PC e transmite a mesma para o servidor 402. O servidor 402 então aplica os dados de dispositivo de entrada recebidos e aplica os mesmos ao jogo ou aplicativo que é executado no servidor 402, que é um PC que executa Windows utilizando uma CPU Intel 2.16GHz Core Duo. O servidor 402 então produz um novo quadro e emite o mesmo através de sua saída DVI, a partir de um sistema gráfico baseado em placa-mãe, ou através de uma saída de DVI da placa NVIDIA 8800GTX PCI Express.

Simultaneamente, o servidor 402 emite o áudio produzido pelo jogo ou aplicações através de sua saída de áudio digital (por exemplo, S/PDIF), que é acoplado à entrada de áudio digital no PC baseado em dual quad-core Xeon que está implementando a compactação de vídeo. Um compactador de áudio de fonte aberta Vorbis é usado para compactar o áudio simultaneamente com o vídeo utilizando qualquer núcleo disponível para o encadeamento de processo. Em uma modalidade, o núcleo que

conclui a compactação de seu bloco primeiro, executa a compactação de áudio. O áudio compactado é então transmitido juntamente com o vídeo compactado, e é descompactado no cliente 415 utilizando um descompactador de áudio Vorbis.

Distribuição Central do Servidor de Serviço de Hospedagem

O vidro atravessante de luz, tal como uma fibra óptica, viaja em alguma fração da velocidade da luz em um vácuo, e, logo, pode-se determinar uma velocidade exata de propagação para luz em fibra óptica. Porém, na prática, permitindo-se tempo para retardos de roteamento, ineficiências de transmissão, e outro overhead, observa-se que as latências ótimas na Internet refletem velocidades de transmissão mais próximas a 50% da velocidade da luz. Portanto, uma latência ótima de ida e volta de 1000 milhas é aproximadamente igual a 22ms, e uma latência ótima de ida e volta de 3000 milhas é igual a cerca de 64ms. Portanto, um único servidor em uma costa dos EUA estará muito afastado para servir clientes na outra costa (que pode estar até 3000 afastada) com a latência desejada. No entanto, conforme ilustrado na figura 13a, se a central de servidor 1300 do serviço de hospedagem 210 estiver localizada no centro dos EUA (por exemplo, Kansas, Nebraska, etc.), de tal modo que a distância par qualquer ponto na parte continental dos EUA seja de aproximadamente 1500 milhas ou menor, a latência de ida e volta da Internet pode tão baixa quanto 32 ms. Referindo-se à figura 4b, nota-se que embora as piores latências permitidas ao usuário ISP 453 seja igual a 25ms, tipicamente, observam-se latências mais próximas a 10-15ms com sistemas DSL e modem a cabo. Da mesma forma, a figura 4b supõe uma distância máxima a partir do local do usuário 211 até a central de hospedagem 210 de 1000 milhas. Portanto, com uma latência típica de ida e volta de usuário ISP igual a 15ms usada e uma distância máxima da Internet igual a 1500 milhas para uma latência de ida e volta de 32ms, a latência total de ida e volta a partir do ponto que um usuário ativa o dispositivo de entrada 421 e observa uma resposta no dispositivo de exibição 422 é igual a 1+1+15+32+1+16+6+8 = 80ms. Logo, o tempo de resposta de 80ms pode ser tipicamente alcançado por uma distância de Internet de

1500 milhas. Isto permitiria que qualquer local de usuário com uma latência de usuário curta o suficiente 453 na parte continental dos EUA acesse uma única central de servidor que esteja centralmente localizada.

Em outra modalidade, ilustradas na figura 13b, as centrais de servidor do serviço de hospedagem 210, HS1-HS6, estão estrategicamente posicionadas nos Estados Unidos (ou em outra região geográfica), com determinadas centrais de servidor de serviço de hospedagem maiores posicionadas próximas a grandes centros populacionais (por exemplo, HS2 e HS5). Em uma modalidade, as centrais de servidor HS1-HS6 trocam informações através de uma rede 1301 que pode ser a Internet ou uma rede privada ou uma combinação de ambas. Com múltiplas centrais de servidor, os serviços podem ser proporcionados em uma latência inferior aos usuários que tenham uma alta latência de usuário ISP 453.

Embora a distância na Internet seja obviamente um fator que contribui para a latência de ida e volta através da Internet, algumas vezes, surgem outros fatores que não sejam amplamente relacionados à latência. Algumas vezes, um fluxo de pacote é roteado através da Internet até um local distante e volta novamente, resultando em latência a partir do loop longo. Algumas vezes, existe um equipamento de roteamento na trajetória que não esteja operando apropriadamente, resultando em um retardo da transmissão. Algumas vezes, existe um tráfego sobrecarregando uma trajetória que introduz retardo. E, algumas vezes, existe uma falha que evita que o ISP do usuário roteie até um determinado destino. Portanto, embora a Internet geralmente proporcione conexões a partir de um ponto para outro com uma rota claramente confiável e ótima e uma latência que seja amplamente determinada pela distância (especialmente com conexões de longa distância que resultam em um roteamento fora da área local do usuário), tal confiabilidade e latência não são garantidas de forma alguma e, frequentemente, não podem ser obtidas a partir do local de um usuário até um determinado destino na Internet.

Em uma modalidade, quando um cliente de usuário 415 se conectar inicialmente ao serviço de hospedagem 210 para jogar um videogame

ou usar um aplicativo, o cliente se comunica com cada uma das centrais de servidor de serviço de hospedagem HS1-HS6 disponíveis mediante a inicialização (por exemplo, utilizando-se as técnicas descritas anteriormente). Se a latência for baixa o suficiente para uma conexão particular, então, utiliza-se tal conexão. Em uma modalidade, o cliente se comunica com todas, ou com um subconjunto das centrais de servidor de serviço de hospedagem e aquela com a conexão de menor latência é selecionada. O cliente pode selecionar a central de serviço com a conexão de menor latência ou as centrais de serviço podem identificar aquela com a conexão de menor latência e proporcionar estas informações (por exemplo, sob a forma de um endereço da Internet) ao cliente.

Se uma central de servidor de serviço de hospedagem particular estiver sobrecarregada e/ou o jogo ou aplicativo do usuário puder tolerar a latência à outra central de servidor de serviço de hospedagem menos carregada, então, o cliente 415 pode ser redirecionado à outra central de servidor de serviço de hospedagem. Nessa situação, o jogo ou aplicativo que o usuário está executando seria pausado no servidor 402 na central de servidor sobrecarregada do usuário, e os dados de estado de jogo ou aplicativo seriam transferidos a um servidor 402 em outra central de servidor de serviço de hospedagem. Então, o jogo ou aplicativo seria retomado. Em uma modalidade, o serviço de hospedagem 210 aguardaria até que o jogo ou aplicativo tenha alcançado um ponto de pausa natural (por exemplo, entre os níveis em um jogo, ou após o usuário iniciar uma operação "salvar" no aplicativo) para realizar a transferência. Ainda em outra modalidade, o serviço de hospedagem 210 aguardaria até que a atividade do usuário cesse durante um período especifico de tempo (por exemplo, 1 minuto) e, então, iniciaria a transferência neste momento.

Conforme descrito anteriormente, em uma modalidade, o serviço de hospedagem 210 se subscreve a um serviço de desvio de Internet 440 da figura 14 para tentar proporcionar uma latência prometida a seus clientes. Os serviços de desvio de Internet, conforme o uso em questão, são serviços que proporcionam rotas de rede privada de um ponto para outro na Internet

com as características prometidas (por exemplo, latência, taxa de dados, etc.). Por exemplo, se o serviço de hospedagem 210 estiver recebendo uma grande quantidade de tráfego a partir dos usuários que utilizam o serviço DSL da AT&T oferecido em São Francisco, ao invés de rotear aos escritórios centrais baseados em São Francisco da AT&T, o serviço de hospedagem 210 pode arrendar uma conexão de dados privados de alta capacidade a partir de um provedor de serviço (talvez o próprio AT&T ou outro provedor) entre os escritórios centrais sediados em São Francisco e uma ou mais centrais de servidor para serviço de hospedagem 210. Então, se as rotas a partir de todas as centrais de servidor de serviço de hospedagem HS1-HS6 através da Internet geral até um usuário em São Francisco usarem o resultado AT&T DSL em uma latência muito alta, então, a conexão de dados privados poderia ser usada no lugar. Embora as conexões de dados privados sejam genericamente mais dispendiosas do que as rotas através da Internet geral, contanto que estas permaneçam uma pequena porcentagem das conexões de serviço de hospedagem 210 aos usuários, o impacto geral de custos será baixo, e os usuários passarão por uma experiência de serviço mais consistente.

Geralmente, as centrais de servidor têm duas camadas de fonte de energia de backup no caso de queda de energia. Tipicamente, a primeira camada é a fonte de energia de backup de baterias (ou de uma fonte de energia disponível imediatamente alternativa, tal como um volante que é mantido funcionando e fixado a um gerador), que proporciona energia imediatamente quando a rede elétrica falhar e manter a central de servidor funcionando. Se a queda de energia for breve, e a rede elétrica retornar rapidamente (por exemplo, dentro de um minuto), então, as baterias são tudo o que seria necessário para manter a central de servidor funcionando. Porém, se a queda de energia ocorrer por um período mais longo de tempo, então, tipicamente, inicializam-se os geradores (por exemplo, movidos a diesel) que substituem as baterias e podem funcionar até que tenham combustível. Esses geradores são extremamente dispendiosos visto que devem ser capazes de produzir mais energia do que a central de servidor normal-

mente obtém a partir da rede elétrica.

Em uma modalidade, cada um dos serviços de hospedagem HS1-HS5 compartilha dados de usuário entre si, de tal modo que se uma central de servidor tiver uma queda de energia, a mesma pode pausar os jogos e aplicações que estiverem em processo, e, então, transferir os dados de estado de jogo ou aplicativo a partir de cada servidor 402 aos servidores 402 em outras centrais de servidor, e, então, notificará o cliente 415 de cada usuário a direcionar as comunicações ao novo servidor 402. Visto que situações como estas ocorrem raramente, pode ser aceitável transferir um usuário a uma central de servidor de serviço de hospedagem que não seja capaz de proporcionar uma latência ótima (isto é, o usuário simplesmente precisará tolerar uma latência superior pela duração da queda de energia), que proporcionará uma faixa muito mais de opções para transferir usuários. Por exemplo, dadas as diferenças de fuso-horário nos EUA, os usuários na Costa Leste podem ir dormir às 11:30 PM enquanto os usuários na Costa Oeste às 8:30 PM estão atingido o uso de pico de videogames. Se houver uma queda de energia em uma central de servidor de serviço de hospedagem na Costa Oeste neste horário, podem não existir servidores suficientes na Costa Oeste 402 em outras centrais de servidor de serviço de hospedagem para manipular todos os usuários. Nessa situação, alguns usuários podem ser transferidos para as centrais de servidor de serviço de hospedagem na Costa Leste que tenham servidores disponíveis 402, e a única consequência aos usuários seria uma latência maior. Uma vez que os usuários forem transferidos a partir da central de servidor que perdeu energia, a central de servidor pode, então, iniciar um desligamento sistemático de seus servidores e equipamentos, de tal modo que todo o equipamento seja desligado antes que as baterias (ou outro backup de energia imediato) se esgotem. Desta forma, podem-se evitar os custos de um gerador para a central de servidor.

Em uma modalidade, durante horários de carregamento pesado do serviço de hospedagem 210 (seja devido ao carregamento de usuário de pico, ou pelo fato de um ou mais centrais de servidor falharem) os usuários

são transferidos a outras centrais de servidor com base nos requerimentos de latência do jogo ou aplicativo que eles estiverem usando. Logo, os usuários que usam os jogos ou aplicativos que requerem baixa latência dariam preferência às conexões de servidor com baixa latência quando existir um suprimento limitado.

Recursos de Serviço de Hospedagem

A figura 15 ilustra uma modalidade de componentes de uma central de servidor para serviço de hospedagem 210 utilizada nas descrições de recurso a seguir. Assim como no serviço de hospedagem 210 ilustrado na figura 2a, os componentes desta central de servidor são controlados e coordenados por um sistema de controle 401 do serviço de hospedagem 210 exceto onde qualificado em contrário.

O tráfego de entrada da internet 1501 proveniente dos clientes de usuário 415 é direcionado ao roteamento de entrada 1502. Tipicamente, o tráfego de entrada da internet 1501 entrará na central de servidor através de uma conexão de alta velocidade por fibra óptica à Internet, porém, qualquer meio de conexão de rede com largura de banda adequada, confiabilidade de latência baixa será suficiente. O roteamento de entrada 1502 é um sistema de rede (a rede pode ser implementada como uma rede Ethernet, uma rede de canal de fibra, ou através de qualquer outro meio de transporte) tendo comutações e servidores de roteamento que suportam as comutações que recolhem os pacotes de chegada e roteiam cada pacote ao servidor de aplicativo/jogo ("app/jogo") apropriado 1521-1525. Em uma modalidade, um pacote que é distribuído a um servidor particular de app/jogo representa um subconjunto dos dados recebidos a partir do cliente e/ou pode ser convertido/alterado por outros componentes (por exemplo, componente de rede, tais como gateways e roteadores) na central de dados. Em alguns casos, os pacotes serão roteados a mais de um servidor 1521-1525 em um período, por exemplo, se um jogo ou aplicativo estiver sendo executado de uma vez em paralelo em múltiplos servidores. As matrizes RAID 1511-1512 são conectados à rede de roteamento de entrada 1502, de tal modo que os servidores de app/jogo 1521-1525 possam ler e gravar às matrizes RAID 1511-

1512. Além disso, uma matriz RAID 1515 (que pode ser implementada como múltiplas matrizes RAID) também é conectada ao roteamento de entrada 1502 e os dados provenientes da matriz RAID 1515 podem ser lidos a partir dos servidores de app/jogo 1521-1525. O roteamento de entrada 1502 pode ser implementado em uma ampla faixa de arquiteturas de rede da técnica anterior, incluindo uma estrutura em árvore de comutações, com o tráfego de entrada da internet 1501 em sua raiz; em uma estrutura em malha que interconecta todos os vários dispositivos; ou como uma série de sub-redes interconectadas, com tráfego concentrado entre os dispositivos de intercomunicação diferenciado do tráfego entre outros dispositivos. Um tipo de configuração de rede é um SAN que, embora tipicamente usado para dispositivos de armazenamento, também pode ser usado para transferência de dados em alta velocidade entre dispositivos. Da mesma forma, os servidores de app/jogo 1521-1525 podem ter múltiplas conexões de rede ao roteamento de entrada 1502. Por exemplo, um servidor 1521-1525 pode ter uma conexão de rede a uma sub-rede conectada às Matrizes RAID 1511-1512 e outra conexão de rede a uma sub-rede conectada a outros dispositivos.

Os servidores de app/jogo 1521-1525 podem ser todos configurados igualmente, alguns diferentemente, ou todos diferentemente, conforme descrito previamente em relação aos servidores 402 na modalidade ilustrada na figura 4a. Em uma modalidade, cada usuário, ao utilizar o serviço de hospedagem, está tipicamente usando pelo menos um servidor de app/jogo 1521-1525. Por motivos de simplicidade de explicação, deve-se supor que um determinado usuário está usando o servidor de app/jogo 1521, porém, múltiplos servidores podem ser usados por um usuário, e múltiplos usuários podem compartilhar um único servidor de app/jogo 1521-1525. A entrada de controle do usuário, enviada a partir do cliente 415 conforme descrito previamente é recebida como tráfego de entrada de Internet 1501, e roteada através do roteamento de entrada 1502 ao servidor de app/jogo 1521. O servidor de app/jogo 1521 usa a entrada de controle do usuário como uma entrada de controle ao jogo ou aplicativo executado no servidor, e computa o próximo quadro de vídeo e o áudio associado a este. Então, o servidor de

app/jogo 1521 emite o vídeo/áudio não-compactado 1529 à compactação de vídeo compartilhado 1530. O servidor de app/jogo pode emitir o vídeo não-compactado através de qualquer meio, incluindo uma ou mais conexões Gigabit Ethernet, porém, em uma modalidade, o vídeo é emitido através de uma conexão DVI e o áudio e outras informações de estado de compactação e canal de comunicação são emitidas através de uma conexão de Barramento Serial Universal (USB).

A compactação de vídeo compartilhado 1530 compacta o vídeo e o áudio não-compactados a partir dos servidores de app/jogo 1521-1525. A compactação pode ser totalmente implementada em hardware, ou em um software executado em hardware. Pode existir um compactador dedicado para cada servidor de app/jogo 1521-1525, ou se os compactadores forem rápidos o suficiente, um determinado compactador pode ser usado para compactar o vídeo/áudio a partir de mais de um servidor de app/jogo 1521-1525. Por exemplo, em 60fps um tempo de quadro de vídeo é igual a 16,67ms. Se um compactador for capaz de compactar um quadro em 1ms, então, tal compactador pode ser usado para compactar o vídeo/áudio a partir de até 16 servidores de app/jogo 1521-1525 adotando-se a entrada proveniente de um servidor após o outro, com o compactador salvando o estado de cada processo de compactação de vídeo/áudio e comutando o contexto à medida que realiza um ciclo entre os fluxos de vídeo/áudio a partir dos servidores. Isto resulta em economias substanciais de custos em hardware de compactação. Visto que diferentes servidores completarão os quadros em diferentes momentos, em uma modalidade, os recursos de compactador se encontram em um pool compartilhado 1530 com um meio de armazenamento compartilhado (por exemplo, RAM, Flash) que serve para armazenar o estado de cada processo de compactação, e quando um quadro de servidor 1521-1525 estiver completo e pronto para que seja compactado, um meio de controle determines qual recurso de compactação está disponível em tal momento, proporciona ao recurso de compactação o estado do processo de compactação do servidor e o quadro de vídeo/áudio não-compactado a ser comprimido.

Nota-se que parte do estado para cada processo de compactação de servidor inclui informações sobre a própria compactação, tais como os dados de buffer de quadro descompactado do quadro anterior que podem ser usados como uma referência para P blocos, a resolução da saída de vídeo; a qualidade da compactação; a estrutura lado a lado; a alocação de bits por blocos; a qualidade de compactação, o formato de áudio (por exemplo, estéreo, sistema surround, Dolby® AC-3). No entanto, o estado de processo de compactação também inclui informações de estado de canal de comunicação referentes aos dados à taxa de pico de dados 941 e se um quadro anterior (conforme ilustrado na figura 9b) está sendo atualmente emitido (e como resultado o quadro atual deve ser ignorado), e potencialmente se existem características de canal que devem ser consideradas na compactação, tal como uma perda excessiva de pacote, que afete as decisões para a compactação (por exemplo, em termos da frequência de I blocos, etc). À medida que a taxa de pico de dados 941 ou outras características de canal se alteram com o passar do tempo, conforme determinado por um servidor de app/jogo 1521-1525 que suporta cada um dos dados de monitoramento de usuário enviados a partir do cliente 415, o servidor de app/jogo 1521-1525 envia informações relevantes à compactação de hardware compartilhado 1530.

A compactação de hardware compartilhado 1530 também empacota o vídeo/áudio compactado utilizando-se meios como aqueles descritos anteriormente, e, se apropriado, aplicar códigos FEC, duplicar determinados dados, ou adotar outras etapas para garantir adequadamente a capacidade do fluxo de dados de vídeo/áudio a ser recebido pelo cliente 415 e descompactado com uma maior qualidade e confiabilidade possível.

Alguns aplicativos, tais como aqueles descritos mais adiante, requerem que a saída de vídeo/áudio de um determinado servidor de app/jogo 1521-1525 esteja disponível em múltiplas resoluções (ou em outros múltiplos formatos) simultaneamente. Se o servidor de app/jogo 1521-1525 notificar o recurso de compactação de hardware compartilhado 1530, então, o áudio/vídeo não-compactado 1529 de tal servidor de app/jogo 1521-1525

será simultaneamente compactado em diferentes formatos, diferentes resoluções, e/ou em diferentes estruturas de correção de pacote/erro. Em alguns casos, alguns recursos de compactação podem ser compartilhados entre múltiplos processos de compactação que compactam o mesmo vídeo/áudio (por exemplo, em muitos algoritmos de compactação, existe uma etapa por meio da qual a imagem é escalonada em múltiplos tamanhos antes de aplicar a compactação. Se for necessário que imagens com tamanhos diferentes sejam emitidas, então, esta etapa pode ser usada para servir vários processos de compactação de uma vez). Em outros casos, os recursos de compactação separada serão necessários para cada formato. Em qualquer caso, o vídeo/áudio compactado 1539 de todas as várias resoluções e formatos necessários para um determinado servidor de app/jogo 1521-1525 (seja este um ou muitos) será emitido de uma vez ao roteamento de saída 1540. Em uma modalidade, a saída do vídeo/áudio compactado 1539 se encontra em um formato UDP, logo, este consiste em um fluxo unidirecional de pacotes.

A rede de roteamento de saída 1540 compreende uma série de servidores e comutadores de roteamento que direcionam cada fluxo de vídeo/áudio compactado ao(s) usuário(s) destinado(s) ou outros destinos através da interface de tráfego de saída de Internet 1599 (que, tipicamente, conectaria uma interface de fibra à Internet) e/ou de volta ao buffer de retardo 1515, e/ou de volta ao roteamento de entrada 1502, e/ou para fora através de uma rede privada (não mostrada) para distribuição de vídeo. Nota-se que (conforme descrito anteriormente) o roteamento de saída 1540 pode emitir um determinado fluxo de vídeo/áudio a múltiplos destinos de uma vez. Em uma modalidade, este é implementado utilizando-se multicast de Protocolo de Internet (IP) no qual se radiodifunde um determinado fluxo UDP destinado a realizar um fluxo contínuo em múltiplos destinos de uma vez, e a radiodifusão é repetida pelos servidores e comutadores de roteamento no roteamento de saída 1540. Os múltiplos destinos da radiodifusão podem ser clientes de múltiplos usuários 415 através da Internet, a múltiplos servidores de app/jogo 1521-1525 através do roteamento de entrada 1502,

e/ou a um ou mais buffers de retardo 1515. Portanto, a saída de um determinado servidor 1521-1522 é compactada em um ou múltiplos formatos, e cada fluxo compactado é direcionado a um ou a múltiplos destinos.

Além disso, em outra modalidade, se múltiplos servidores de app/jogo 1521-1525 forem simultaneamente usados por um usuário (por exemplo, em uma configuração de processamento paralelo para criar uma saída em 3D de uma cena complexa) e cada servidor estiver produzindo parte da imagem resultante, a saída de vídeo de múltiplos servidores 1521-1525 pode ser combinada pela compactação de hardware compartilhado 1530 em um quadro combinado, e a partir deste ponto em diante, tratada conforme descrito anteriormente como se fosse proveniente de um único servidor de app/jogo 1521-1525.

Nota-se que em uma modalidade, uma cópia (em pelo menos uma resolução do vídeo, ou em uma resolução superior, observada pelo usuário) de todo o vídeo gerado pelos servidores de app/jogo 1521-1525 é gravada no buffer de retardo 1515 durante pelo menos alguns minutos (15 minutos em uma modalidade). Isto permite que cada usuário "rebobine" o vídeo a partir de cada sessão com a finalidade de rever ações ou proezas anteriores (no caso de um jogo). Portanto, em uma modalidade, cada fluxo de saída de vídeo/áudio compactado 1539 sendo roteado a um cliente de usuário 415 também está sendo realizado multicast a um buffer de retardo 1515. Quando o vídeo/áudio for armazenado em um buffer de retardo 1515, um diretório no buffer de retardo 1515 proporciona uma referência cruzada entre o endereço de rede do servidor de app/jogo 1521-1525 que seja a fonte do vídeo/áudio retardado e o local no buffer de retardo 1515 onde o vídeo/áudio retardado pode ser encontrado.

Jogos ao Vivo, Instantaneamente Visualizáveis e Jogáveis

Os servidores de app/jogo 1521-1525 podem não apenas ser usados para executar um determinado aplicativo ou videogame para um usuário, mas eles também podem ser usados para criar os aplicativos de interface de usuário para o serviço de hospedagem 210 que suporta navegação através do serviço de hospedagem 210 e outros recursos. Uma captu-

ra de tela de tal aplicativo de interface de usuário é mostrada na figura 16, uma tela "Game Finder". Esta tela de interface de usuário particular permite que um usuário veja 15 jogos que estão sendo jogados ao vivo (ou retardados) por outros usuários. Cada uma das janelas de vídeo "em miniatura", tal como 1600 é uma janela de vídeo ao vivo em movimento mostrando o vídeo do jogo de um usuário. A vista mostrada na miniatura pode ser a mesma vista que o usuário está vendo, ou pode ser uma vista retardada (por exemplo, se um usuário estiver jogando um jogo de luta, um usuário pode não desejar que outros usuários vejam onde ele está se escondendo e pode escolher retardar qualquer vista de seu jogo durante um período de tempo de digamos 10 minutos). A vista também pode ser uma vista de câmera de um jogo que seja diferente de qualquer vista do usuário. Através de seleções de menu (não mostradas nesta ilustração), um usuário pode escolher uma seleção de jogos para visualizar de uma vez, com base em uma variedade de critérios. Como uma pequena amostragem de escolhas exemplificadoras, o usuário pode selecionar uma seleção aleatória de jogos (tal como aquela mostrada na figura 16), todos de um tipo de jogos (todos sendo jogados por diferentes jogadores), apenas os jogadores mais bem classificados de um jogo, jogadores em um determinado nível no jogo, ou jogadores com pior classificação (por exemplo, se o jogador estiver aprendendo o básico), jogados que sejam "amigos" (ou rivais), jogos que tenham o maior número de espectadores, etc.

Em geral, nota-se que cada usuário decidirá se o vídeo de seu jogo ou aplicativo pode ser visualizado por terceiros e, se puder, quais terceiros, e quanto pode ser visualizado por outros, se apenas for visualizável com um retardo.

O servidor de app/jogo 1521-1525 que estiver gerando a tela de interface de usuário mostrada na figura 16 adquire os 15 feeds de vídeo/áudio enviando-se uma mensagem ao servidor de app/jogo 1521-1525 para cada usuário cujo jogo está sendo solicitado. A mensagem é enviada através do roteamento de entrada 1502 ou outra rede. A mensagem incluirá o tamanho e o formato do vídeo/áudio solicitado, e identificará o usuário vi-

sualizando a tela de interface de usuário. Um determinado usuário pode escolher selecionar o modo de "privacidade" e não permitir quaisquer outros usuários visualizem o vídeo/áudio de seu jogo (seja a partir de seu ponto de vista ou a partir de outro ponto de vista), ou conforme descrito no parágrafo anterior, um usuário pode escolher permitir a visualização de vídeo/áudio de seu jogo, porém, retardar o vídeo/áudio visualizado. Um usuário servidor de app/jogo 1521-1525 que recebe e aceita uma solicitação para permitir que seu vídeo/áudio seja visualizado reconhecerá como tal ao servidor de solicitação, e também notificará a compactação de hardware compartilhado 1530 da necessidade de gerar um fluxo de vídeo compactado adicional no formato solicitado ou tamanho de tela (supondo que o formato e tamanho de tela sejam diferentes daqueles já gerados), e também indicará o destino para o vídeo compactado (isto é, o servidor de solicitação). Se o vídeo/áudio solicitado for apenas retardado, então, o servidor de solicitação de app/jogo 1521-1525 será notificado, e adquirirá o vídeo/áudio retardado a partir de um buffer de retardo 1515 procurando-se o local do vídeo/áudio no diretório em um buffer de retardo 1515 e o endereço de rede do servidor de app/jogo 1521-1525 que seja a fonte do vídeo/áudio retardado. Uma vez que todas essas solicitações tiverem sido geradas e manipuladas, até 15 fluxos de vídeo ao vivo em miniatura serão roteados a partir do roteamento de saída 1540 até o roteamento de entrada 1502 ao servidor de app/jogo 1521-1525 que gera a tela de interface de usuário, e serão descompactados e exibidos pelo servidor. Os fluxos de vídeo/áudio retardados podem estar em um tamanho de tela muito grande, e, se estiverem, o servidor de app/jogo 1521-1525 descompactará os fluxos e reduzir a escala dos fluxos de vídeo ao tamanho em miniatura. Em uma modalidade, as solicitações por áudio/vídeo são enviadas (e gerenciadas por) um serviço de "gerenciamento" central similar ao sistema de controle de serviço de hospedagem da figura 4a (não mostrado na figura 15) que, então, redireciona as solicitações ao servidor de app/jogo apropriado 1521-1525. Ademais, em uma modalidade, nenhuma solicitação pode ser requerida devido ao fato de as miniaturas serem "enviadas por push" aos clientes daqueles usuários que as permitem.

Os áudios dos 15 jogos todos mixados simultaneamente podem criar uma cacofonia de som. O usuário pode escolher mixar todos os sons juntos desta forma (talvez apenas captar a sensação do "ruído" criado por todas as ações sendo visualizadas), ou o usuário pode escolher apenas escutar o áudio de um jogo de uma vez. A seleção de um único jogo é realizada movendo-se a caixa de seleção amarela 1601 (que aparece como um contorno retangular preto na renderização em preto e branco da figura 16) para um determinado jogo (o movimento da caixa amarela pode ser realizada utilizando-se as teclas direcionais em um teclado, movendo-se um mouse, movendo-se um joystick, ou apertando-se botões direcionais em outro dispositivo, tal como um telefone móvel). Uma vez que um único jogo for selecionado, reproduz-se apenas o áudio deste jogo. Da mesma forma, as informações de jogo 1602 são mostradas. No caso deste jogo, por exemplo, o logotipo do publicador (por exemplo, "EA" para "Electronic Arts") e o logotipo do jogo, "por exemplo, Need for Speed Carbon" e uma barra horizontal laranja (renderizada na figura 16 como uma barra com faixas verticais) indica em termos relativos o número de pessoas jogando ou visualizando o jogo neste momento particular (muitos, neste caso, logo, o jogo é "Hot"). Proporcionam-se "Stats" adicionais (isto é, estatísticas), indicando que existem 145 jogadores ativamente jogando 80 instanciações diferentes do Jogo Need for Speed (isto é, pode ser jogado por um jogador individual ou por jogadores múltiplos), e existem 680 espectadores (dentre os quais este usuário é um). Nota-se que estas estatísticas (e outras estatísticas) são coletadas pelo sistema de controle de serviço de hospedagem 401 e são armazenadas em matrizes RAID 1511-1512, para manter registros da operação do serviço de hospedagem 210 e para cobrar apropriadamente os usuários e os publicadores pagantes que proporcionam conteúdos. Algumas dessas estatísticas são gravadas devido às ações pelo sistema de controle de serviço 401, e algumas são reportadas ao sistema de controle de serviço 401 pelo servidor de app/jogo individual 1521 - 1525. Por exemplo, o servidor de app/jogo 1521-1525 executando este aplicativo Game Finder envia mensagens ao sistema de controle de serviço de hospedagem 401 quando

os jogos estiverem sendo visualizados (e quando sua visualização for cessada) de tal modo que possa atualizar as estatísticas de quantos jogos se encontram em visualização. Algumas dessas estatísticas estão disponíveis para aplicativos de interface de usuário, tal como este aplicativo Game Finder.

Se o usuário clicar um botão de ativação em seu dispositivo de entrada, ele verá o vídeo em miniatura na caixa amarela maximizada enquanto continua a jogar o vídeo ao vivo em tela cheia. Este efeito é mostrado no processo da figura 17. Nota-se que a janela de vídeo 1700 foi ampliada em tamanho. Objetivando implementar este efeito, o servidor de app/jogo 1521-1525 solicita que o servidor de app/jogo 1521-1525 que executa o jogo selecionado tenha uma cópia do fluxo de vídeo para um tamanho em tela cheia (na resolução do dispositivo de exibição do usuário 422) do jogo roteado ao mesmo. O servidor de app/jogo 1521-1525 que executa o jogo notifica o compactador de hardware compartilhado 1530 que uma cópia em miniatura do jogo não é mais necessária (a não ser que outro servidor de app/jogo 1521-1525 requeira tal miniatura), e, então, ordena que o mesmo envie uma cópia de tamanho em tela cheia do vídeo ao servidor de app/jogo 1521-1525 aplicando zoom ao vídeo. O usuário que joga o jogo pode ter ou não um dispositivo de exibição 422 tendo a mesma resolução do usuário que maximiza o jogo. Além disso, outros espectadores do jogo podem ter ou não dispositivos de exibição 422 tendo a mesma resolução do usuário que maximiza o jogo (e pode ter diferentes meio de reprodução de áudio, por exemplo, estéreo ou sistema surround). Portanto, o compactador de hardware compartilhado 1530 determina se já foi gerado um fluxo de vídeo/áudio compactado adequado que satisfaça os requerimentos do usuário que solicita o fluxo de vídeo/áudio e se não existir, o mesmo notifica o roteamento de saída 1540 a rotear uma cópia do fluxo ao servidor de app/jogo 1521-1525 aplicando zoom ao vídeo, se não compactar outra cópia do vídeo que seja adequada para tal usuário e instrui o roteamento de saída a enviar o fluxo de volta ao roteamento de entrada 1502 e ao servidor de app/jogo 1521-1525 que aplica zoom ao vídeo. Este servidor, que agora

recebe uma versão em tela cheia do vídeo selecionado o descompactará e gradualmente ampliará a escala para tamanho máximo.

A figura 18 ilustra como a tela se parece após o jogo ter sido completamente maximizado até o modo em tela cheia e o jogo é mostrado em uma resolução máxima do dispositivo de exibição do usuário 422 conforme indicado pela imagem apontada pela seta 1800. O servidor de app/jogo 1521-1525 que executa o aplicativo Game Finder envia mensagens aos outros servidores de app/jogo 1521-1525 que proporcionaram miniaturas que não sejam mais necessárias e mensagens ao servidor de controle de serviço de hospedagem 401 que os outros jogos não estão sendo mais visualizados. Neste ponto, a única exibição que está gerando é uma sobreposição 1801 no topo da tela que fornece informações e controles de menu ao usuário. Nota-se que à medida que este jogo foi progredido, a audiência aumentou para 2.503 espectadores. Com tantos espectadores, limitam-se como sendo muitos espectadores com dispositivos de exibição 422 que tenham uma resolução igual ou quase igual (cada servidor de app/jogo 1521-1525 tem a capacidade de escalonar o vídeo para ajustar a instalação).

Devido ao fato de o jogo mostrado ser um jogo de múltiplos jogadores, o usuário pode decidir entrar no jogo em algum momento. O serviço de hospedagem 210 pode permitir ou não que o usuário entre no jogo por uma variedade de razões. Por exemplo, o usuário pode ter que pagar para jogar o jogo e escolher por não fazê-lo, o usuário pode não ter ranking suficiente para entrar neste jogo em particular (por exemplo, não seria competitivo aos outros jogadores), ou a conexão de Internet do usuário pode não ter uma latência baixa o suficiente para permitir que o usuário jogue (por exemplo, não existe uma restrição de latência para visualizar jogos, logo, um jogo que estiver sendo jogado distante (de fato, em outro continente) pode ser visualizado sem questões de latência, porém, para um jogo a ser jogado, a latência deve ser baixa o suficiente para que o usuário (a) aproveite o jogo, e (b) esteja em iguais condições aos outros jogadores que podem ter conexões de latência inferior). Se o usuário tiver permissão para jogar, então, o servidor de app/jogo 1521-1525 que forneceu a interface de usuário Game

Finder ao usuário solicitará que o servidor de controle de serviço de hospedagem 401 inicie (isto é, localize e inicialize) um servidor de app/jogo 1521-1525 que seja adequadamente configurado para jogar o jogo em particular para carregar o jogo a partir de uma matriz RAID 1511-1512, e, então, o servidor de controle de serviço de hospedagem 401 instruirá o roteamento de entrada 1502 a transferir os sinais de controle a partir do usuário ao servidor de app/jogo que agora esteja hospedando o jogo e instruirá a compactação de hardware compartilhado 1530 a comutar de compactar o vídeo/áudio a partir do servidor de app/jogo que estava hospedando o aplicativo Game Finder para compactar o vídeo/áudio a partir do servidor de app/jogo que agora esteja hospedando o jogo. A sincronização vertical do serviço de app/jogo Game Finder e o novo servidor de app/jogo que hospeda o jogo não estão sincronizados, e, como resultado, provavelmente ocorrerá uma diferença de tempo entre as duas sincronizações. Devido ao fato de o hardware de compactação de vídeo compartilhado 1530 começar a compactação do jogo mediante um servidor de app/jogo 1521-1525 que completa um quadro de vídeo, o primeiro quadro proveniente do novo servidor pode ser completo antes de um tempo de quadro completo do servidor antigo, que pode ser acontecer antes de o quadro anteriormente compactado completar sua transmissão (por exemplo, considera-se o tempo de transmissão 992 da figura 9b: se o quadro não-compactado 3 963 tiver completado metade de um tempo de quadro antecipadamente, isto afetaria o tempo de transmissão 992). Nesta situação, o hardware de compactação de vídeo compartilhado 1530 ignorará o primeiro quadro do novo servidor (por exemplo, como o Quadro 4 964 é ignorado 974), e o cliente 415 manterá o último quadro do servidor antigo por um tempo extra de quadro, e o hardware de compactação de vídeo compartilhado 1530 começará a compactar o próximo vídeo de tempo de quadro do novo servidor de app/jogo que hospeda o jogo. Visualmente, ao usuário, a transição de um servidor de app/jogo para outro será contínua. Então, o servidor de controle de serviço de hospedagem 401 notificará o servidor de app/jogo 1521-1525 que estava hospedando o Game Finder a comutar para um estado ocioso, até que este

seja novamente necessário.

Então, o usuário está apto a jogar o jogo. E, o que é excepcional é que o jogo será jogado de modo instantaneamente perceptual (visto que será carregado no servidor de app/jogo 1521-1525 a partir de uma matriz RAID 1511-1512 em velocidade de gigabit/segundo), e o jogo será carregado em um servidor exatamente adequado para o jogo junto a um sistema operacional exatamente configurado para o jogo com as unidades ideais, *configuração de registro (no caso do Windows), e sem outros aplicativos* executados no servidor que podem competir com a operação do jogo.

Da mesma forma, à medida que o progride através do jogo, cada um dos segmentos do jogo será carregado no servidor em velocidade de gigabit/segundo (isto é, 1 gigabyte carrega em 8 segundos) a partir da matriz RAID 1511-1512, e por causa da vasta capacidade de armazenamento da matriz RAID 1511-1512 (visto que é um recurso compartilhado entre muitos usuários, pode ser muito grande, ainda pode ter uma boa relação custo-benefício), configuração de geometria ou outra configuração de segmento de jogo pode ser pré-computada e armazenada na matriz RAID 1511-1512 e carregada de modo extremamente rápido. Ademais, devido ao fato de a configuração de hardware e as capacidades computacionais de cada servidor de app/jogo 1521-1525 serem conhecidas, os sombreados de pixel e vértice podem ser pré-computados.

Portanto, o jogo será inicializado quase instantaneamente, será executado em um ambiente ideal, e os segmentos subsequentes serão carregados quase instantaneamente.

Porém, além dessas vantagens, o usuário será capaz de ver terceiros jogando o jogo (através do Game Finder, previamente descrito e por outros meios) e ambos decidem se o jogo está interessante, e, se estiver, aprendem dicas assistindo aos outros. E, o usuário será capaz de demonstrar o jogo instantaneamente, sem precisar aguardar por um download grande e/ou instalação, e o usuário será capaz de jogar o jogo instantaneamente, talvez em uma base experimental por uma pequena taxa, ou em longo prazo. E, o usuário será capaz de jogar o jogo em um PC

Windows, em um Macintosh, em um aparelho de televisão, em casa, quando estiver viajando, e mesmo em um telefone móvel, com uma conexão sem fio com latência baixa o suficiente (embora a latência não seja uma questão para os meros espectadores). E, isto pode ser realizado sem nunca ter posse física do jogo.

Conforme mencionado anteriormente, o usuário pode decidir não permitir que seu jogo seja visualizável por terceiros, permitir que seu jogo seja visualizável após um retardo, permitir que seu jogo seja visualizável por usuários selecionados, ou permitir que seu jogo seja visualizável por todos os usuários. Independentemente, o vídeo/áudio será armazenado, em uma modalidade, durante 15 minutos em um buffer de retardo 1515, e o usuário será capaz de "rebobinar" e assistir a seu jogo anterior, e pausar, o reproduzir em câmera lenta, avançar rapidamente, etc., assim como seria capaz assistir TV com um Gravador de Vídeo Digital (DVR). Embora neste exemplo, o usuário esteja jogando um jogo, a mesma capacidade "DVR" se encontra disponível se o usuário estiver utilizando um aplicativo. Isto pode ser útil na revisão de um trabalho anterior e em outras aplicações, conforme descrito mais adiante. Além disso, se o jogo tiver sido projetado com a capacidade de rebobinagem com base nas informações de estado do jogo, de tal modo que a vista de câmera possa ser alterada, etc., então, esta capacidade "3D DVR" também será suportada, porém, exigirá que o jogo seja projetado para suportá-la. A capacidade "DVR" utilizando-se um buffer de retardo 1515 funcionará com qualquer jogo ou aplicativo, limitado naturalmente, ao vídeo que foi gerado quando o jogo ou aplicativo foi usado, porém, no caso de jogos com capacidade 3D DVR, o usuário pode controlar um "sobrevoo" em 3D de um segmento previamente jogado, e ter o buffer de retardo 1515 gravado no vídeo resultante e ter o estado do jogo do segmento de jogo gravado. Portanto, um "sobrevoo" particular será gravado como vídeo compactado, porém, visto que o estado do jogo também será gravado, um sobrevoo diferente será possível em uma data posterior do mesmo segmento do jogo.

Conforme descrito mais adiante, os usuários no serviço de

hospedagem 210 terão uma User Page, onde eles podem postar informações sobre eles mesmos e outros dados. Dentre as coisas que os usuários serão capazes de postar encontram-se os segmentos de vídeo de um jogo que eles salvaram. Por exemplo, se o usuário tiver superado um desafio particularmente difícil em um jogo, o usuário pode "rebobinar" até um pouco antes do local onde ele obteve sua grande realização no jogo, e, então, instruir o serviço de hospedagem 210 a salvar um segmento de vídeo de certa duração (por exemplo, 30 segundos) na User Page do usuário para que outros usuários assistam. Objetivando implementar isto, esta consiste simplesmente em uma questão do servidor de app/jogo 1521-1525 que o usuário está usando para reproduzir o vídeo armazenado em um buffer de retardo 1515 a uma matriz RAID 1511-1512 e, então, indexar este segmento de vídeo na User Page do usuário.

0Se o jogo tiver a capacidade de 3D DVR, conforme descrito anteriormente, então, as informações de estado do jogo requeridas para o 3D DVR também podem ser gravadas pelo usuário e tornadas disponíveis para a User Page do usuário.

No caso onde um jogo é projetado para que tenha "espectadores" (isto é, usuários que sejam capazes de viajar através do mundo 3D e observar a ação sem participar no mesmo) além dos jogadores ativos, então, o aplicativo Game Finder permitirá que os usuários entrem nos jogos como espectadores assim como jogados. A partir de um ponto de vista de implementação, não existe diferença ao sistema de hospedagem 210 se um usuário for um espectador ao invés de um jogador ativo. O jogo será carregado em um servidor de app/jogo 1521-1525 e o usuário estará controlando o jogo (por exemplo, controlando uma câmera virtual que visualiza neste mundo). A única diferença será a experiência de jogo do usuário.

Colaboração de Múltiplos Usuários

Outro recurso do serviço de hospedagem 210 é a capacidade de múltiplos usuários colaborarem enquanto assistem a um vídeo ao vivo, mesmo se utilizarem dispositivos amplamente discrepantes para visualização. Isto é útil tanto ao se jogar os jogos como ao se utilizar os aplicativos.

Muitos PCs e telefones móveis são equipados com câmeras de vídeo e apresentam a capacidade de realizar uma compactação de vídeo em tempo real, particularmente quando a imagem for pequena. Da mesma forma, estão disponíveis câmeras pequenas que possam ser conectadas a uma televisão, e não seja difícil implementar uma compactação em tempo real em software ou utilizando-se um entre os muitos dispositivos de compactação de hardware para compactar o vídeo. Da mesma forma, muitos PCs e todos os telefones móveis têm microfones, e fones de ouvido estão disponíveis com microfones.

Essas câmeras e/ou microfones, combinados com a capacidade de compactação local de vídeo/áudio (particularmente empregando as técnicas de compactação de vídeo com baixa latência descritas no presente documento) permitirão que um usuário transmita vídeo e/ou áudio a partir dos locais do usuário 211 ao serviço de hospedagem 210, junto aos dados de controle do dispositivo de entrada. Quando essas técnicas forem empregadas, então, uma capacidade ilustrada na figura 19 é obtenível: um usuário pode ter seu vídeo e áudio 1900 aparecendo na tela no jogo ou aplicativo de outro usuário. Este exemplo é um jogo de múltiplos jogadores, onde parceiros de equipe colaboram em uma corrida de carros. Um vídeo/áudio do usuário pode ser seletivamente visualizável / escutável apenas por seus parceiros de equipe. E, visto que efetivamente não existiria latência, utilizando-se das técnicas descritas anteriormente, os jogadores seriam capazes de conversar ou realizar movimentos entre si em tempo real sem um retardo perceptível.

Esta integração de vídeo/áudio é realizada tendo-se o vídeo e/ou áudio compactados a partir da câmera/microfone de um usuário chegando como um tráfego de entrada da internet 1501. Então, o roteamento de entrada 1502 roteia o vídeo e/ou áudio aos servidores de app/jogo 1521-1525 que têm permissão para ver/escutar ao vídeo e/ou áudio. Então, os usuários dos respectivos servidores de app/jogo 1521-1525 que escolhem usar o vídeo e/ou áudio os descompactam e integram conforme desejado para aparecer no jogo ou aplicativo, tal como ilustrado por 1900.

O exemplo da figura 19 mostra como tal colaboração é usada em um jogo, porém, essa colaboração pode ser uma ferramenta imensamente poderosa para aplicativos. Considerando-se uma situação onde um grande edifício esta sendo projetado para a cidade de Nova York por arquitetos em Chicago para um construtor imobiliário baseado em Nova York, porém, a decisão envolve um investidor financeiro que está viajando e calha de mo mesmo estar em um aeroporto em Miami, e uma decisão precisa ser tomada sobre determinados elementos de projeto do edifício em termos de como este se combina com os edifícios próximos ao mesmo, para satisfazer tanto o investidor como o construtor imobiliário. Supõe-se que a firma de arquitetura tenha um monitor de alta resolução com uma câmera conectada a um PC em Chicago, o construtor imobiliário tem um laptop com uma câmera em Nova York, e o investidor tem um telefone móvel com uma câmera em Miami. A firma de arquitetura pode usar o serviço de hospedagem 210 para hospedar um aplicativo de projeto arquitetônico poderoso que seja capaz de renderizar de modo altamente realístico em 3D, e pode fazer uso de um grande banco de dados dos edifícios na cidade de Nova York, assim como um banco de dados do edifício sob projeto. O aplicativo de projeto arquitetônico executar em um, ou se requerer uma grande quantidade de potência computacional, em muitos, dos servidores de app/jogo 1521-1525. Cada um dos 3 usuários em locais discrepantes se conectará ao serviço de hospedagem 210, e terá uma vista simultânea da saída de vídeo do aplicativo de projeto arquitetônico, porém, estará apropriadamente dimensionado pela compactação de hardware compartilhado 1530 para o determinado dispositivo e características de conexão de rede que cada usuário tem (por exemplo, a firma de arquitetura pode ver uma exibição 2560x1440 60fps através de uma conexão comercial à Internet de 20Mbps, o construtor imobiliário em Nova York pode ser uma imagem de 1280x720 60fps por uma conexão DSL de 6 Mbps em seu laptop, e o investidor pode ver uma imagem de 320x180 60fps por uma conexão de dados via celular de 250Kbps em seu telefone móvel. Cada parte escutará a voz das outras partes (a chamada de conferência será manipulada por qualquer entre os muitos pacotes de

software de chamada de conferência amplamente disponíveis no(s) servidor(es) de app/jogo 1521-1525) e, através do acionamento de um botão em um dispositivo de entrada de usuário, um usuário será capaz de fazer com que o vídeo apareça utilizando-se sua câmera local. À medida que a reunião procede, os arquitetos serão capazes de mostrar como o edifício se parece à medida que o gira e o leva para próximo a outro edifício na área, com uma renderização em 3D extremamente fotorrealistica, e o meso vídeo será visível a todas as partes, na resolução de cada dispositivo de exibição. Não importará que nenhum dos dispositivos locais usados por qualquer parte seja incapaz de manusear animação em 3D com tal realismo, deixado sozinho transferindo por download ou até mesmo armazenando um vasto banco de dados requerido para renderizar os edifícios vizinhos na cidade de Nova York. A partir do ponto de vista de cada um dos usuários, apesar da distância, e apesar dos dispositivos locais discrepantes, eles simplesmente terão uma experiência contínua com um grau incrível de realismo. E, quando uma parte desejar que seu rosto seja visto para melhor transmitir seu estado emocional, ela pode fazê-lo. Além disso, se o construtor imobiliário ou o investidor desejarem assumir o controle do programa arquitetônico e usar seu próprio dispositivo de entrada (seja um teclado, mouse, teclado numérico ou tela sensível ao toque), eles podem, e responderão sem latência perceptual (supondo-se que sua conexão de rede não tenha uma latência irracional). Por exemplo, no caso do telefone móvel, se o telefone móvel estiver conectado a uma rede WiFi no aeroporto, este terá uma latência muito baixa. Porém, se utilizar as redes de dados celulares disponíveis atualmente nos EUA, provavelmente sofrerão de um tempo de retardo considerável. Ainda, para a maioria dos propósitos da reunião, onde o investidor está vendo os arquitetos controlarem o edifício ou conversando em teleconferência, até mesmo a latência celular deve ser aceitável.

Finalmente, ao fim da chamada de conferência colaborativa, o construtor imobiliário e o investidor farão seus comentários e sairão do serviço de hospedagem, a firma de arquitetura será capaz de "rebobinar" o vídeo da conferência que foi gravado em um buffer de retardo 1515 e analisar

os comentários, expressões faciais e/ou ações aplicadas ao modelo em 3D do edifício realizadas durante a reunião. Se existirem segmentos particulares que eles desejam salvar, estes segmentos de vídeo/áudio podem ser movidos do buffer de retardo 1515 para uma matriz RAID 1511-1512 para um armazenamento arquivado e reprodução posterior.

Da mesma forma, a partir de uma perspectiva de custos, se os arquitetos apenas precisarem usar a potência computacional e o grande banco de dados da cidade de Nova York durante uma chamada de conferência de 15 minutos, eles precisam apenas pagar pelo tempo que os recursos foram usados, ao invés de precisar ter estações de trabalho fortes e adquirir uma cópia cara de um grande banco de dados.

Serviços Comunitários Predominantemente Visuais

O serviço de hospedagem 210 permite uma oportunidade sem precedentes para estabelecer serviços comunitários predominantemente visuais na Internet. A figura 20 mostra uma User Page exemplificadora para um jogador no serviço de hospedagem 210. Assim como o aplicativo Game Finder, o User Page é um aplicativo que é executado em um dos servidores de app/jogo 1521-1525. Todas as miniaturas e janelas de vídeo nesta página mostram vídeos constantemente em movimento (se os segmentos forem curtos, eles executam loop).

Utilizando-se uma câmera de vídeo ou carregando-se vídeo, o usuário (cujo nome de usuário é "KILLHAZARD") é capaz de postar um vídeo dele mesmo 2000 que outros usuários possam ver. O vídeo é armazenado em uma matriz RAID 1511-1512. Da mesma forma, quando outros usuários visitarem a User Page de KILLHAZARD, se KILLHAZARD estiver usado o serviço de hospedagem 210 no momento, o vídeo ao vivo 2001 de qualquer coisa que ele estiver fazendo (supondo-se que ele permite que usuários entrem em sua User Page para vê-lo) será mostrado. Isto será realizado pelo servidor de app/jogo 1521-1525 que hospeda o aplicativo de User Page solicitado a partir do sistema de controle de serviço 401 se KILLHAZARD estiver ativo, e, se estiver, o servidor de app/jogo 1521-1525 que ele está usando. Então, utilizando-se os mesmos métodos usados pelo

aplicativo Game Finder, um fluxo de vídeo compactado em uma resolução e formato adequados será enviado ao servidor de app/jogo 1521-1525 que executa o aplicativo User Page e o mesmo será exibido. Se um usuário selecionar a janela com o jogo ao vivo de KILLHAZARD, e, então, clicar apropriadamente em seu dispositivo de entrada, a janela será maximizada (novamente utilizando-se os mesmos métodos dos aplicativos Game Finder, e o vídeo ao vivo preencherá a tela, na resolução do dispositivo de exibição do usuário espectador 422, apropriada para as características da conexão de Internet do usuário espectador.

Uma vantagem principal disto em relação às abordagens da técnica anterior é que o usuário visualizando a User Page é capaz de ver um jogo jogado ao vivo que o usuário não possui, e pode muito bem não ter um computador local ou console de jogo capaz de jogar o jogo. Isto oferece uma grande oportunidade para o usuário veja o usuário mostrado na User Page "em ação" jogando jogos, e uma oportunidade para aprender sobre um jogo que o usuário espectador pode desejar tentar ou se aperfeiçoar.

Os videoclipes gravados por câmera ou carregados dos amigos de KILLHAZARD 2002 também são mostrados na User Page, e abaixo de cada videoclipe tem um texto que indica que o amigo está online jogando um jogo (por exemplo, six_shot está jogando o jogo "Eragon" (mostrado aqui como Game4) e MrSnuggles99 está Offline, etc.). Clicando-se em um item de menu (não mostrado), os videoclipes do amigo trocam de vídeos gravados ou carregados para vídeo ao vivo de o que os amigos que estão atualmente jogando os jogos no serviço de hospedagem 210 estão fazendo neste momento em seus jogos. Logo, torna-se um grupamento Game Finder para os amigos. Se o jogo de um amigo for selecionado e o usuário clicar no mesmo, este será maximizado em tela cheia, e o usuário será capaz de assistir ao jogo jogado ao vivo em tela cheia.

Novamente, o usuário assistindo ao jogo do amigo não possui uma cópia do jogo, nem recursos de console de computação/jogo local para jogar o jogo. A visualização do jogo é efetivamente instantânea.

Conforme previamente descrito, quando um usuário jogar um

jogo no serviço de hospedagem 210, o usuário é capaz de "rebobinar" o jogo e encontrar um segmento de vídeo que ele deseja salvar, e, então, salva o segmento de vídeo em sua User Page. Estes são denominados como "Brag Clips™". Os segmentos de vídeo 2003 são todos Brag Clips 2003 salvos por KILLHAZARD a partir de jogos anteriores que ele jogou. O número 2004 mostra quantas vezes um Brag Clip foi visualizado, e quando o Brag Clip for visualizado, os usuários têm uma oportunidade de classificá-los, e os ícones com formato de fechadura de número laranja (mostrados como contornos pretos) 2005 indicam o quão alta é a classificação. Os Brag Clips 2003 executam loop constantemente quando um usuário visualizar a User Page, junto ao restante do vídeo na página. Se o usuário selecionar e clicar em um dos Brag Clips 2003, ele maximiza para apresentar o Brag Clip 2003, junto aos controles de DVR de modo a permitir que o clipe seja reproduzido, pausado, rebobinado, avançado rapidamente, saltado, etc.

A reprodução de Brag Clip 2003 é implementada pelo servidor de app/jogo 1521-1525 que carrega o segmento de vídeo compactado armazenado em uma matriz RAID 1511-1512 quando o usuário gravou o Brag Clip e o descompacta e o reproduz.

Os Brag Clips 2003 também podem ser segmentos de vídeo "3D DVR" (isto é, uma sequência de estado de jogo a partir do jogo que pode ser reproduzido novamente e permite que o usuário altere o ponto de vista da câmera) a partir de jogos que suportam tal capacidade. Neste caso, as informações de estado de jogo são armazenadas, além de uma gravação de vídeo compactado do "sobrevoo" particular que o usuário realizou quando o segmento de jogo foi gravado. Quando a User Page estiver sendo visualizada, e todas as miniaturas e janelas de vídeos estiverem constantemente executando loop, um Brag Clip 3D DVR 2003 constantemente executará loop do Brag Clip 2003 que foi gravado como um vídeo compactado quando o usuário gravou o "sobrevoo" do segmento de jogo. Porém, quando um usuário selecionar um Brag Clip 3D DVR 2003 e clicar no mesmo, além dos controles de DVR para permitir que o Brag Clip de vídeo compactado seja reproduzido, o usuário será capaz de clicar em um botão que fornece a eles

uma capacidade de 3D DVR para o segmento de jogo. Eles serão capazes de controlar por si próprios uma câmera de "sobrevoo" durante o segmento de jogo, e, se desejarem (e o usuário que possui a página de usuário permitir), serão capazes de gravar um Brag Clip de "sobrevoo" alternativo em uma forma de vídeo compactado que estará disponível a outros espectadores da página de usuário (seja imediatamente, ou após o proprietário da página de usuário tiver uma chance de analisar o Brag Clip).

Esta capacidade de Brag Clip 3D DVR 2003 é habilitada ativando-se o jogo que está para reproduzir novamente as informações de estado de jogo gravadas em outro servidor de app/jogo 1521-1525. Visto que o jogo pode ser ativado quase instantaneamente (conforme descrito anteriormente), não é difícil ativá-lo, com seu jogo limitado ao estado de jogo gravado pelo segmento Brag Clip, e, então, permitir que o usuário realize um "sobrevoo" com uma câmera enquanto grava o vídeo compactado em um buffer de retardo 1515. Uma vez que o usuário tiver completado a realização do "sobrevoo", o jogo é desativado.

A partir do ponto de vista do usuário, a ativação de um "sobrevoo" com um Brag Clip 3D DVR 2003 não é mais difícil do que controlar os controles DVR de um Brag Clip linear 2003. Estes podem não saber nada sobre o jogo ou nem mesmo como jogá-lo. Estes consistem apenas em um operador de câmera virtual visualizando em um mundo 3D durante um segmento de jogo gravado por outro.

Os usuários também serão capazes de realizar overdub em seu próprio áudio em Brag Clips que seja gravado a partir de microfones ou carregado. Desta forma, os Brag Clips podem ser usados para criar animações personalizadas, utilizando-se os personagens e as ações dos jogos. Esta técnica de animação é comumente conhecida como "machinima".

À medida que os usuários progridem através dos jogos, eles atingirão diferentes níveis de habilidade. Is jogos jogados reportarão as realizações ao sistema de controle de serviço 401, e estes níveis de habilidade serão mostrados nas User Pages.

Propagandas Animadas Interativas

As propagandas online realizaram uma transição de texto, para imagens imóveis, para vídeo, e agora para segmentos interativos, tipicamente implementados utilizando-se clientes finos de animação como Adobe Flash. A razão pela qual os clientes finos de animação serem usados é que os usuários tipicamente têm pouca paciência para serem retardados ao privilegio de terem um produto ou serviço estabelecido a eles. Da mesma forma, os clientes finos são executados em PCs de desempenho muito baixo e, como tal, o anunciante pode ter um alto grau de confidência que a propaganda interativa funcionará de modo apropriado. Infelizmente, os clientes finos de animação, tal como Adobe Flash são limitados ao grau de interatividade e à duração da experiência (para atenuar o tempo de download e ser operável em quase todos os dispositivos de usuário, incluindo PCs de baixo desempenho e Macs sem GPUs ou CPUs de alto desempenho).

A figura 21 ilustra uma propaganda interativa onde o usuário deve selecionar as cores exteriores e interiores de um carro enquanto o carro gira em uma sala de exibição, enquanto o traçamento de raios em tempo real mostra como o carro se parece. Então, o usuário escolhe um avatar para dirigir o carro, e, então, o usuário pode pegar o carro para uma volta seja em uma pista de corrida, ou em um local exótico, tal como Mônaco. O usuário pode selecionar um motor maior, ou melhores pneus, e, então, pode ver como a configuração alterada afeta a capacidade do carro para acelerar ou se manter na pista.

Naturalmente, a propaganda consiste efetivamente em um videogame em 3D sofisticado. Porém, tal propaganda a ser exibida em um PC ou em um console de videogame talvez requeira um download de 100MB e, no caso do PC, pode requerer a instalação de drivers especiais, e pode não ser executada se o PC não tiver uma capacidade computacional CPU ou GPU. Portanto, essas propagandas são impraticáveis em configurações da técnica anterior.

No serviço de hospedagem 210, tais propagandas são iniciadas quase instantaneamente, e são perfeitamente rodadas, sem importar quais sejam as capacidades do cliente de usuário 415. Logo, elas são iniciadas

mais rapidamente do que as propagandas interativas de cliente fino, são vastamente mais ricos em experiência, e altamente confiáveis.

Geometria de Fluxo Contínuo durante Animação em Tempo Real

A matriz RAID 1511-1512 e o roteamento de entrada 1502 podem proporcionar taxas de dados que sejam tão rápidas e com latências tão baixas que seja possível projetar videogames e aplicativos que dependem da matriz RAID 1511-1512 e do roteamento de entrada 1502 para distribuir confiavelmente a geometria de saída no meio do jogo ou em um aplicativo durante uma animação em tempo real (por exemplo, um sobrevoo com um banco de dados complexo.)

Através dos sistemas da técnica anterior, tal como um sistema de videogame mostrado na figura 1, os dispositivos de armazenamento em massa disponíveis, particularmente em dispositivos domiciliares práticos, são muito lentos para geometria de fluxo contínuo durante o jogo exceto em situações onde a geometria requerida for de alguma forma previsível. Por exemplo, em um jogo de direção onde existe uma pista específica, a geometria para edifícios que surgem em vista pode ser razoavelmente bem prevista e os dispositivos de armazenamento em massa podem buscar antecipadamente ao local onde a geometria de entrada está localizada.

Porém, em uma cena complexa com alterações imprevisíveis (por exemplo, em uma cena de batalha com personagens complexos) se a RAM no PC ou sistema de videogame for completamente preenchida com geometria para objetos atualmente em vista, e, então, o usuário gira repentinamente seu personagem para ver o que tem atrás do personagem, se a geometria não tiver sido pré-carregada em RAM, então, pode haver um retardo antes de o mesmo poder ser exibido.

No serviço de hospedagem 210, as matrizes RAID 1511-1512 podem realizar fluxo contínuo em dados em excesso de velocidade Gigabit Ethernet, e com uma rede SAN, é possível alcançar uma velocidade de 10 gigabit/segundo em uma Ethernet de 10 Gigabit ou por outras tecnologias de rede. 10 gigabits/segundo carregarão um gigabyte de dados em menos de um segundo. Em um tempo de quadro de 60fps (16,67ms), aproximadamen-

te 170 megabits (21MB) de dados podem ser carregados. Naturalmente, girar a mídia mesmo em uma configuração RAID ainda incorrerá latências maiores do que um tempo de quadro, porém, o armazenamento RAID baseado em Flash será eventualmente tão grande quanto as matrizes RAID de mídia e não incorrerá tal latência alta. Em uma modalidade, utiliza-se armazenamento em cache por gravação massiva em RAM para proporcionar acesso de latência muito baixa.

Portanto, com uma velocidade de rede suficientemente alta, e um armazenamento em massa com latência suficientemente baixa, a geometria pode realizar fluxo contínuo em servidores de app/jogo 1521-1525 tão rapidamente quanto os CPUs e/ou GPUs podem processar os dados em 3D. Logo, no exemplo dado previamente, onde um usuário gira seu personagem repentinamente e olha para trás, a geometria para todos os personagens atrás pode ser carregada antes de o personagem completar a rotação, e, portanto ao usuário, parecerá que ele se encontra em um mundo foto-realístico que seja tão real quando uma ação ao vivo.

Conforme previamente discutido, uma das últimas fronteiras em animação computacional fotorrealística é o rosto humano, e por causa da sensibilidade dos olhos humanos a imperfeições, o menos erro de um rosto foto-real pode resultar em uma reação negativa do espectador. A figura 22 mostra como um desempenho ao vivo capturado utilizando-se a Tecnologia de Captura de Realidade Contour™ (assunto dos pedidos co-pendentes: "Apparatus and method for capturing the motion of a performer," Ser. No. 10/942.609, depositado em 15 de setembro de 2004; "Apparatus and method for capturing the expression of a performer," Ser. No. 10/942.413 depositado em 15 de setembro de 2004; "Apparatus and method for improving marker identification within a motion capture system," Ser. No. 11/066.954, depositado em 25 de fevereiro de 2005; "Apparatus and method for performing motion capture using shutter synchronization," Ser. No. 11/077.628, depositado em 10 de março de 2005; "Apparatus and method for performing motion capture using a random pattern on capture surfaces," Ser. No. 11/255.854, depositado em 20 de outubro de 2005; "System and method for performing

motion capture using phosphor application techniques," Ser. No. 11/449.131, depositado em 7 de junho de 2006; "System and method for performing motion capture by strobing a fluorescent lamp," Ser. No. 11/449.043, depositado em 7 de junho de 2006; "System and method for three dimensional capture of stop-motion animated characters," Ser. No. 11/449.127, depositado em 7 de junho de 2006", sendo que cada um desses é atribuído ao requerente do presente pedido CIP) resulta em uma superfície capturada bastante suave, então, uma superfície rastreada por contagem de polígonos (isto é, o movimento poligonal segue o movimento do rosto precisamente). Finalmente, quando o vídeo do desempenho ao vivo for mapeado na superfície rastreada para produzir uma superfície texturizada, produz-se um resultado foto-real.

Muito embora uma tecnologia GPU atual seja capaz de renderizar o número de polígonos na superfície rastreada e texturizar e iluminar a superfície em tempo real, se os polígonos e texturas forem alterados a cada tempo de quadro (que produzirá os resultados mais foto-reais), rapidamente consumirão toda a RAM disponível de um PC moderno ou de um console de videogame.

Utilizando-se as técnicas de geometria de fluxo contínuo descritas anteriormente, torna-se prático alimentar continuamente a geometria nos servidores de app/jogo 1521-1525 de tal modo que possam animar rostos foto-reais continuamente, permitindo a criação de videogames com rostos que sejam quase indistinguíveis de rostos de ação ao vivo.

Integração de conteúdo linear com recursos alternativos

Filmes de cinema, programação de televisão e material de áudio (coletivamente, "conteúdo linear") estão amplamente disponíveis para usuários domésticos e corporativos em muitas formas. O conteúdo linear pode ser adquirido em mídia física, como mídia de CD, DVD e Blu-ray. Também pode ser registrado por DVRs a partir de radiodifusão de TV a cabo e satélite. Ademais, está disponível como conteúdo pay-per-view (PPV) através de satélite e TV a cabo e como vídeo sob demanda (VOD) em TV a cabo.

Crescentemente, o conteúdo linear está disponível através da

Internet, tanto como conteúdo transferido por download quanto como conteúdo de fluxo contínuo. Atualmente, não há de fato um local para ir a fim de experimentar todos os recursos associados à mídia linear. Por exemplo, DVDs e outra mídia óptica de vídeo têm tipicamente recursos interativos não disponíveis em qualquer lugar, como comentários do diretor, clipes de "making of", etc. Os sites de música online abrangem informações sobre arte e canções geralmente não disponíveis em CDs, mas nem todos os CDs estão disponíveis online. E os sites da web associados à programação de televisão têm frequentemente recursos, blogs e, algumas vezes, comentários extra dos atores ou pessoal de criação.

Adicionalmente, com muitos filmes de cinema ou eventos esportivos, existem frequentemente vídeo games que são liberados (no caso de filmes de cinema) frequentemente junto com a mídia linear ou (no caso de esportes) podem estar estritamente atrelados a eventos do mundo real (por exemplo, o comércio de jogadores).

O serviço de hospedagem 210 é bem adequado para a distribuição de conteúdo linear na ligação de formas distintas de conteúdo relacionado. Certamente, a distribuição de filmes de cinema não é mais desafiadora que a distribuição de vídeo games altamente interativos, e o serviço de hospedagem 210 é capaz de distribuir conteúdo linear para uma ampla faixa de dispositivos, no lar ou no escritório, ou para dispositivos móveis. A figura 23 mostra uma página de interface de usuário exemplificativa para serviço de hospedagem 210 que mostra uma seleção de conteúdo linear.

Mas, diferentemente da maioria dos sistemas de distribuição de conteúdo linear, o serviço de hospedagem 210 também é capaz de distribuir componentes interativos relacionados (por exemplo, os menus e recursos em DVDs, as sobreposições interativas em HD-DVDs, e animação em Adobe Flash (conforme explicado abaixo) em sites da web). Dessa forma, as limitações do dispositivo do cliente 415 não mais introduzem limitações com para as quais os recursos estão disponíveis.

Adicionalmente, o sistema de hospedagem 210 é capaz de unir o conteúdo linear com conteúdo de vídeo game dinamicamente, em tempo

real. Por exemplo, se um usuário estiver assistindo uma partida de Quadribol em um filme do Harry Potter, e decide que quer tentar jogar Quadribol, ele pode apenas clicar em um botão e o filme irá pausar e imediatamente ele será transportado para o segmento de Quadribol de um vídeo game do Harry Potter. Após jogar a partida de Quadribol, um outro clique de um botão, e o filme irá reiniciar instantaneamente.

Com tecnologia de produção e gráficos fotorreais, onde o vídeo capturado fotograficamente é indistinguível a partir de personagens de ação ao vivo, quando um usuário realiza uma transição de um jogo de Quadribol em um filme de ação ao vivo para um jogo de Quadribol em um vídeo game em um serviço de hospedagem conforme descrito no presente documento, as duas cenas são virtualmente indistinguíveis. Isto fornece opções criativas totalmente novas para diretores tanto de conteúdo linear quanto de conteúdo interativo (por exemplo, vídeo game) conforme as linhas entre as duas palavras se tornam indistinguíveis.

Através da utilização da arquitetura de serviço de hospedagem mostrada na figura 14, o controle da câmera virtual em um filme 3D pode ser oferecido ao expectador. Por exemplo, em uma cena que ocorre dentro de um vagão de trem, seria possível permitir que o expectador controle a câmera virtual e observe em torno do vagão, enquanto o roteiro progride. Considera-se que todos os objetos 3D ("ativos") no vagão estejam disponíveis bem como um nível adequado de potência de computação capaz de produzir as cenas em tempo real bem como o filme original.

E até mesmo para entretenimento não gerado por computador, existem muito recursos interativos excitantes que podem ser oferecidos. Por exemplo, o filme de cinema de 2005 "Pride and Prejudice" teve muitas cenas em mansões inglesas antigas ornadas. Para certas cenas da mansão, o usuário pode pausar o vídeo e, então, controlar câmera para fazer um tour pela mansão, ou talvez a área circundante. Para implantar isto, uma câmera poderia ser carregada através da mansão com uma lente olho de peixe posto que deixa um rastro de sua posição, bem como o QuickTime VR da Apple, Inc. da técnica anterior é implementado. Os vários quadros seriam,

então, transformados, então, as imagens não são distorcidas e, então, armazenados em matriz RAID 1511-1512 junto com o filme, e reproduzidos novamente quando o usuário escolhe participar de um tour virtual.

Com eventos esportivos, um evento de esportes ao vivo, tal como um jogo de basquetebol, pode ser transmitido através do serviço de hospedagem 210 para usuários assistir, como se fosse para TV regular. Após os usuários terem assistidos um jogo particular, um vídeo game do jogo (eventualmente com jogadores de basquetebol parecidos com os jogadores reais) poderia aparecer com os jogadores iniciando na mesma posição, e os usuários (talvez cada um tomando controle de um jogador) poderiam refazer a partida para ver se eles poderiam fazer melhor que os jogadores.

O serviço de hospedagem 210 descrito no presente documento é extremamente bem adequado para suportar este mundo futurístico, pois é capaz de suportar potência computacional e recursos de armazenamento em massa que são impraticáveis para a instalação em um lar ou na maioria das instalações de um escritório, e também seus recursos computacionais estão sempre atualizados, com o último hardware de computação disponível, enquanto que em uma instalação doméstica, sempre haverá lares com PCs e vídeo games de geração mais antiga. Ademais, no serviço de hospedagem 210, toda esta complexidade computacional é escondida do usuário, mesmo que este possa estar usando sistemas muito sofisticados, do ponto de vista do usuário, é simples como trocar os canais da televisão. Adicionalmente, o usuário seria capaz de acessar toda a potência computacional e as experiências da potência computacional seriam trazidas de qualquer cliente 415.

Jogos para Múltiplos Jogadores

Desde que um jogo seja um jogo para múltiplos jogadores, então, este será capaz de se comunicar tanto com servidores de jogo/aplicativo 1521-1525 através de rede de roteamento de entrada 1502 quanto com uma ponte de rede com a Internet (não mostrada) com máquinas de jogo ou servidores que não estão operando no serviço de hospedagem 210. Quando se joga jogos para múltiplos jogadores com computadores na Internet geral,

então, os servidores de jogo/aplicativo 1521-1525 terão o benefício de acesso extremamente rápido à Internet (em comparação a se o jogo estava operando em um servidor no lar), mas eles seriam limitados pelas capacidades dos outros computadores que executam o jogo em conexões mais lentas, e também potencialmente limitados pelo fato de que os servidores do jogo na Internet foram designados para acomodar o menor denominador comum, o que seria para computadores domésticos em conexões de Internet do consumidor relativamente lentas.

Mas quando um jogo para múltiplos jogadores é jogado totalmente dentro de um centro servidor de serviço de hospedagem 210, então, um mundo de diferença é alcançável. Cada servidor de jogo/aplicativo 1521-1525 que hospeda um jogo para um usuário será interconectado com outros servidores de jogo/aplicativo 1521-1525 bem como quaisquer servidores que estão hospedando o controle central para o jogo para múltiplos jogadores com velocidade extremamente alta, conectividade de latência extremamente baixa e muitos arranjos de armazenamento muito rápidos. Por exemplo, se Gigabit Ethernet for usado para a rede de roteamento de entrada 1502, então, os servidores de jogo/aplicativo 1521-1525 estarão se comunicando uns com os outros e se comunicando com quaisquer servidores que hospedam o controle central para o jogo para múltiplos jogadores em velocidade de gigabit/segundo com potencialmente apenas 1ms de latência ou perda. Adicionalmente, os arranjos RAID 1511-1512 serão capazes de responder muito rapidamente e, então, transferir dados em velocidades de gigabit/segundo. Como exemplo, se um usuário personalizar um personagem em termos de aparência e equipamentos de tal modo que o personagem tenha uma grande quantidade de geometria e comportamentos que são exclusivos para o personagem, com sistemas da técnica anterior limitados ao cliente de jogo que opera no lar em um PC ou console de jogo, se aquele personagem estiver sendo visualizado por outro usuário, o usuário teria que esperar pela conclusão de uma transferência por download lenta e vagarosa de modo que todos os dados de geometria e comportamento carreguem em seu computador. Dentro do serviço de hospedagem 210, a mesma transferência por

download poderia ser por Gigabit Ethernet, servido a partir de um arranjo RAID 1511-1512 em velocidade de gigabit/segundo. Mesmo se o usuário doméstico teve uma conexão com a Internet de 8Mbps (que é extremamente rápida para os padrões atuais), a Gigabit Ethernet é 100 vezes mais rápida. Então, o que levaria um minuto pela conexão de Internet rápida, levaria menos de um segundo pela Gigabit Ethernet.

Agrupamentos e Torneios de Melhor Jogador

O serviço de hospedagem 210 é extremamente bem adequado para torneios. Devido ao fato de que nenhum jogo está sendo executado em um cliente local, não há oportunidade para usuários conversaram (por exemplo, conforme seria o caso em um torneio da técnica anterior através da modificação da cópia do jogo que está sendo executado em seu PC local para proporcioná-los uma vantagem injusta). Ademais, devido à capacidade do roteamento de saída 1540 em multidifundir os fluxos de UDP, o serviço de hospedagem 210 é capaz de radiodifundir os principais torneios para milhares ou mais pessoas na audiência de uma só vez.

Na realidade, quando existem certos fluxos de vídeo que são muito populares em que milhares de usuários estão recebendo o mesmo fluxo (por exemplo, mostrando visualizações de um torneio principal), pode ser mais eficaz enviar o fluxo de vídeo para uma Rede de Distribuição de Conteúdo (CDN) tal como Akamai ou Limelight para distribuição de massa para muitos dispositivos do cliente 415.

Um nível similar de eficácia pode ser obtido quando uma CDN é usada para mostrar uma página de Game Finder de melhores agrupamentos de jogador.

Para torneios principais, um locutor famoso ao vivo pode ser usado para fornecer comentário durante certas partidas. Embora um grande número de usuários esteja assistindo um torneio principal e um número relativamente pequeno esteja jogando no torneio. O áudio do locutor famoso pode ser roteado para os servidores de jogo/aplicativo 1521-1525 que hospedam os usuários que jogam no torneio e que hospedam quaisquer cópias do modo espectador do jogo no torneio, e o áudio pode ser depurado

em cima do áudio do jogo. O vídeo de um locutor famoso pode ser sobreposto nos jogos, talvez, somente em visualizações de espectador.

Aceleração De Carregamento De Página Da Web

A World Wide Web e seu protocolo de transporte primário, Protocolo de Transferência de Hipertexto (HTTP), foram concebidos e definidos em uma era em que somente empresas tinham conexões com a Internet de alta velocidade, e os consumidores que estiveram online estavam usando modems de discagem ou ISDN. Naquele tempo, o "padrão de ouro" para uma conexão rápida era uma linha T1 que fornecia 1,5Mbps de taxa de dados simetricamente (isto é, com taxa de dados igual em ambas as direções).

Atualmente, a situação é completamente diferente. A velocidade de conexão doméstica média através de conexões por cable modem ou DSL em grande parte do mundo desenvolvido tem taxa de dados a jusante muito mais alta que uma linha T1. De fato, em algumas partes do mundo, a fibra até a calçada (FTTC) está trazendo taxas de dados tão altas quanto 50 a 100Mbps para o lar.

Desafortunadamente, o HTTP não foi arquitetado (nem implantado) para levar vantagem de modo eficaz sobre estes aprimoramentos de velocidade dramáticos. Um site da web é uma coleção de arquivos em um servidor remoto. Em termos muito simples, o HTTP solicita o primeiro arquivo, espera que o arquivo seja transferido por download e, então, solicita o segundo arquivo, espera que o arquivo seja transferido por download, etc. Na realidade, o HTTP permite que mais de uma "conexão aberta", isto é, mais de um arquivo seja solicitado por vez, mas, devido aos padrões consentidos (e um desejo de impedir que os servidores da web sejam sobrecarregados) somente muito poucas conexões abertas são permitidas. Além disso, por causa da maneira na qual as páginas da web são construídas, os navegadores frequentemente não estão atentos às múltiplas páginas simultâneas que poderiam estar disponíveis para transferência por download imediatamente (isto é, somente após a passagem de uma página que se torna evidente que um novo arquivo, como uma imagem, precisa ser transferido por download). Dessa forma, os arquivos em site da web são essen-

cialmente carregados um por um. Ademais, por causa do protocolo de solicitação e resposta usado por HTTP, há aproximadamente (acessando servidores da web típicos nos EUA) uma latência de 100ms associada a cada arquivo que é carregado.

Com conexões de velocidade relativamente baixa, isto não causa muito problema, devido ao fato de o tempo de transferência por download para os próprios arquivos domina o tempo de espera para as páginas da web. Mas, conforme as velocidades de conexão crescem, especialmente com páginas da web complexas, problemas começam a surgir.

No exemplo mostrado na figura 24, um site da web comercial típico é mostrado (este site da web particular era de uma marca de calçados atléticos principal). O site da web tem 54 arquivos. Os arquivos incluem arquivos HTML, CSS, JPEG, PHP, JavaScript e Flash, e incluem conteúdo de vídeo. Um total de 1,5MBytes precisa ser carregado antes que a página esteja no ar (isto é, o usuário pode clicar nisto e começar a usar). Existe inúmeras razões para o grande número de arquivos. Uma delas é uma página da web complexa e sofisticada e, a outra, é uma página da web que é montada dinamicamente com base nas informações sobre o usuário que acessa a página (por exemplo, qual país de origem do usuário, qual língua, se o usuário fez compras antes, etc.), e dependente de todos estes fatores, diferentes arquivos são transferidos por download. Ainda, é uma página da web comercial muito típica.

A figura 24 mostra a quantidade de tempo que decorre antes que a página da web esteja no ar conforme a velocidade de conexão cresce. Com uma velocidade de conexão de 1,5 Mbps 2401, com o uso de um servidor da web convencional com um navegador da web convencional, isto leva 13,5 segundos até que a página da web esteja no ar. Com uma velocidade de conexão de 12 Mbps 2402, o tempo de carga é reduzido para 6,5 segundos, ou cerca de duas vezes. Mas com uma velocidade de conexão de 96 Mbps 2403, o tempo de carga é somente reduzido em cerca de 5,5 segundos. A razão pela qual é porque em tal velocidade de transferência por download alta, o tempo para transferir por download os próprios arquivos é

mínimo, mas a latência por arquivo, aproximadamente 100 ms cada, ainda permanece, resultando em 54 arquivos * 100 ms = 5,4 segundos de latência. Dessa forma, não importa quão rápida a conexão para o lar é, este site da web levará sempre pelo menos 5,4 segundos até que esteja no ar. Um outro fator é o enfileiramento do lado do servidor; cada solicitação de HTTP é adicionada na parte posterior da fila, então, em um servidor ocupado isto terá um impacto significativo porque para cada item pequeno a ser obtido a partir do servidor da web, as solicitações de HTTP precisam esperar por sua vez.

Uma maneira de resolver estas questões consiste em descartar ou redefinir HTTP. Ou, talvez para conseguir que o proprietário do site da web consolide melhor seus arquivos em um único single arquivo (por exemplo, em formato de Adobe Flash). Mas, como uma questão prática, esta empresa, bem como muitas outras, tem uma grande transação de investimento em sua arquitetura do site da web. Adicionalmente, embora alguns lares tenham conexões de 12 a 100 Mbps, a maioria dos lares ainda tem velocidades inferiores, e HTTP funciona bem em baixa velocidade.

Uma alternativa é hospedar navegadores da web em servidores de jogo/aplicativo 1521-1525, e hospedar os arquivos para os servidores da web nos arranjos RAID 1511-1512 (ou potencialmente em RAM ou em armazenamento local nos servidores de jogo/aplicativo 1521-1525 que hospedam os navegadores da web. Por causa da interconexão muito rápida através do roteamento de entrada 1502 (ou armazenamento local), em vez de ter 100 ms de latência por arquivo com o uso de HTTP, haverá menos latência por arquivo com o uso de HTTP. Então, em vez de o usuário em sua casa acessar a página da web através de HTTP, o usuário pode acessar a página da web através do cliente 415. Então, mesmo com uma conexão de 1,5 Mbps (por causa disto a página da web não requer muita largura de vídeo para seu vídeo), a página da web estará no ar em menos de 1 segundo por linha 2400. Essencialmente, não haverá latência antes de o navegador da web que opera em um servidor de aplicativo/jogo 1521-1525 estar exibindo uma página no ar, e não haverá latência detectável antes de o

cliente 415 exibir a saída de vídeo do navegador da web. Conforme o usuário movimenta o mouse em torno e/ou digita na página da web, as informações de entrada de usuário serão enviadas para o navegador da web que opera no servidor de aplicativo/jogo 1521-1525, e o navegador da web irá responder consequentemente.

Uma desvantagem desta abordagem é se o compressor estiver constantemente transmitindo dados de vídeo, então, a largura de vídeo é usada, mesmo se a página da web se tornar estática. Isto pode ser remediado através da configuração do compressor para transmitir somente dados quando (e se) a página da web alterar e, então, somente transmitir dados para as partes da página que se alteram. Embora existam algumas páginas da web com banners piscando, etc. que estão constantemente em alteração, tais páginas da web tendem a ser perturbadoras, e usualmente as páginas da web são estáticas salvo se houver uma razão para algo estar em movimento (por exemplo, um clipe de vídeo). Para tais páginas da web, é mais provável o caso em que menos dados serão transmitidos com o uso do serviço de hospedagem 210 que um servidor da web convencional, devido ao fato de que somente as imagens exibidas reais serão transmitidas, nenhum código executável por cliente pequeno, e nenhum objeto grande que nunca pode ser visualizado, tais como imagens giratórias.

Dessa forma, com o uso do serviço de hospedagem 210 para hospedar páginas da web de legado, os tempos de carregamento da página da web podem ser reduzidos para o point em que abrir uma página da web é como alterar canais em uma televisão: a página da web está no ar eficaz e instantaneamente.

Facilitação de Depuração de Jogos e Aplicativos

Conforme mencionado anteriormente, os vídeo games e aplicativos com gráficos em tempo real são aplicativos muito complexos e tipicamente quando são liberados no campo contêm erros. Embora os desenvolvedores de software obtenham retroalimentação de usuários sobre erros, e possam ter alguns meios para superar o estado da maquina após crashes, é muito difícil identificar exatamente o que causou a pane em um jogo ou

aplicativo em tempo real ou a execução inapropriada.

Quando um jogo ou aplicativo opera no serviço de hospedagem 210, a saída de vídeo/áudio do jogo ou aplicativo é constantemente registrada em uma memória temporária de atraso 1515. Adicionalmente, um processo de monitoração opera em cada servidor de aplicativo/jogo 1521-1525 que relata regularmente ao sistema de controle de serviço de hospedagem 401 que o servidor de aplicativo/jogo 1521-1525 está operando suavemente. Se o processo de monitoração falhar no relatório, então, o sistema de controle do servidor 401 irá tentar se comunicar com o servidor de aplicativo/jogo 1521-1525, e se bem sucedido, irá coletar qual estado de máquina está disponível. Quaisquer que sejam as informações disponíveis, junto com o vídeo/áudio registrado pela memória temporária de atraso 1515 serão enviadas para o desenvolvedor do software.

Dessa forma, quando o desenvolvedor do software de jogo ou aplicativo obter o aviso de uma pane do serviço de hospedagem 210, ele obtém um registro quadro a quadro de o que levou à pane. Estas informações podem ser imensamente valiosas no rastreamento de erros e no conserto dos mesmos.

Note também que quando um servidor de aplicativo/jogo 1521-1525 entra em pane, o servidor é reiniciado no ponto reiniciável mais recente, e uma mensagem é fornecida para o usuário com desculpas pela dificuldade técnica.

Compartilhamento de Recurso e Economias de Custo

O sistema mostrado nas figuras 4a e 4b fornece uma variedade de benefícios tanto para usuários finais quanto para desenvolvedores de jogo e aplicativo. Por exemplo, tipicamente, sistemas de cliente doméstico e corporativo (por exemplo, PCs ou consoles de jogo) estão somente em uso por uma pequena porcentagem de horas em uma semana. De acordo com uma publicação de 5 de outubro de 2006 de Nielsen Entertainment "Active Gamer Benchmark Study" (http://www.prnewswire.com/cgi-bin/stories.pl?ACCT=104&STORY=/www/story/ 10-05-2006/0004446115&EDATE=) os jogadores ativos gastam em média 14

horas por semana jogando em consoles de vídeo game e cerca de 17 horas por semana em portáteis. O relatório também relata que para toda atividade de jogo (incluindo console, portátil e jogo em PC), os jogadores ativos gastam em média 13 horas por semana. Levando em consideração a figura superior do tempo de jogo em console de vídeo game, existe 24*7=168 horas em uma semana, o que implica em dizer que um lar do jogador ativo, um console de vídeo game está em uso somente 17/168=10% das horas de uma semana. Ou, 90% do tempo, o console de vídeo game está inativo. Dado o alto custo de consoles de vídeo game, e o fato de que os fabricantes subsidiam tais dispositivos, isto é um uso muito ineficiente de um recurso dispendioso. Os PCs em empresas são também tipicamente usados somente em uma fração das horas da semana, especialmente, PCs do tipo desktop não portáteis frequentemente requeridos para aplicativos de alta tecnologia tal como Autodesk Maya. Embora algumas empresas operem em todas as horas e em feriados, e alguns PCs (por exemplo, portáteis trazidos para casa para a realização de trabalhos à noite) são usados em todas as horas e feriados, a maioria das atividades empresariais tende a estar em torno de 9 h às 17 h, em uma determinada zona de tempo empresarial, de segunda-feira à sexta-feira, menos feriados e pausas (tal como almoço), e uma vez que a maioria dos usos do PC ocorre enquanto o usuário está ativamente engajado com o PC, a utilização do PC do tipo desktop tende a seguir estas horas de operação. Se for considerado que os PCs são utilizados constantemente de 9 h a 17 h, 5 dias por semana, isto implicaria no fato de que os PCs são utilizados 40/168=24% das horas da semana. Os PCs do tipo desktop de desempenho alto são investimentos muito dispendiosos para empresas, e isto reflete um nível de utilização muito baixo. As escolas que estão ensinando em computadores do tipo desktop podem usar computadores para uma fração ainda menor da semana, e embora isto varie dependendo das horas e ensino, a maioria das aulas ocorre durante as horas do dia de segunda-feira a sexta-feira. Então, em geral, os PCs e os consoles de vídeo game são utilizados somente em uma pequena fração das horas da semana.

Notavelmente, devido ao fato de que muitas pessoas estão trabalhando em empresas ou em escola durante as horas do dia de segunda-feira a sexta-feira sem contar feriados, estas pessoas geralmente não estão jogando vídeo games durante estas horas e, então, quando elas jogam vídeo games, isto ocorre geralmente durante outras horas, tais como noites, finais de semana e feriados.

Dada a configuração do serviço de hospedagem mostrado na figura 4a, os padrões de uso descritos nos dois parágrafos acima resultam em utilização de recursos muito eficiente. Obviamente, há um limite no número de usuários que podem usufruir do serviço de hospedagem 210 em um determinado instante, particularmente, se os usuários estão requerendo capacidade de resposta em tempo real para aplicativos complexos como vídeo games 3D sofisticados. Mas, diferentemente de um console de vídeo game em uma casa ou um PC usado por uma empresa, que tipicamente estão inativos a maior parte do tempo, os servidores 402 podem ser reutilizados por diferentes usuários em diferentes instantes. Por exemplo, um servidor de desempenho alto 402 com CPUs duplos de desempenho alto e GPUs duplos e uma grande quantidade de RAM pode ser utilizado por empresas e escolas de 9 h às 17 h em dias úteis, mas ser utilizado por jogadores que jogam um vídeo game sofisticado nas noites, fins de semana e em feriados. Similarmente, os aplicativos de desempenho baixo podem ser utilizados por empresas e escolas em um servidor de desempenho baixo 402 com um CPU Celeron, sem GPU (ou um GPU muito sem recurso) e RAM limitada durante as horas de trabalho e um jogo de baixo desempenho pode utilizar um servidor de desempenho baixo 402 durante as horas de folga.

Adicionalmente, com a disposição de serviço de hospedagem descrita no presente documento, os recursos são compartilhados eficientemente dentre milhares, se não milhões, de usuários. Em geral, os serviços online têm somente uma pequena porcentagem de sua base de usuário total com o uso do serviço em um determinado instante. Se for considerado as estatísticas de uso de vídeo game de Nielsen mencionadas anteriormente, é fácil observar o porquê. Se os jogadores ativos jogam jogos de consoles

somente 17 horas de uma semana, e se for considerado que o tempo de uso pico para jogo é durante as horas típicas de folga das noites (17 h às 0 h, 7*5 dias=35 horas/semana) e final de semana (8 h às 0 h, 16*2=32 horas/semana), então, existem 35+32=65 horas pico por semana para 17 horas de jogo. A carga de usuário pico exata no sistema é difícil de estimar por muitas razões: alguns usuários irão jogar durante tempos fora de pico, pode haver certos instantes no dia em que existem picos de usuários agrupados, os tempos de pico podem ser afetados pelo tipo de jogo jogado (por exemplo, jogos para crianças serão jogados provavelmente mais cedo na noite), etc. Mas, posto que o número médio de horas exibido por um jogador é muito menor que o número de horas do dia quando um jogador está provavelmente jogando um jogo, somente uma fração do número de usuários do serviço de hospedagem 210 estará usando isto em um determinado instante. A título de análise, devemos considerar que a carga de pico é 12,5%. Dessa forma, somente 12,5% dos recursos de computação, compressão e largura de vídeo são usados em um determinado instante, resultando somente em 12,5% do custo de hardware para suportar um determinado usuário para jogar um jogo de determinado nível de desempenho devido à reutilização de recursos.

Além disso, tendo em vista que alguns jogos e aplicativos requerem mais potência computacional que outros, os recursos podem ser alocados dinamicamente com base no jogo a ser jogado ou dos aplicativos executados por usuários. Então, um usuário que seleciona um jogo ou aplicativo de baixo desempenho será alocado em um servidor de baixo desempenho (menos dispendioso) 402, e um usuário que seleciona um jogo ou aplicativo de alto desempenho será alocado em um servidor de alto desempenho (mais dispendioso) 402. Na realidade, um determinado jogo ou aplicativo pode ter seções de desempenho inferior e desempenho superior do jogo ou aplicativos, e o usuário pode ser trocado de um servidor 402 para outro servidor 402 entre seções do jogo ou aplicativo para manter o operando no servidor de custo inferior 402 que satisfaz as necessidades do jogo ou do aplicativo. Observa-se que os arranjos RAID 405, que serão bem

mais rápidos que um único disco, estão disponíveis para todos os servidores de desempenho baixo 402, que terão o benefício das taxas de transferência de disco mais rápidas. Então, custo médio por servidor 402 por todos os jogos a serem jogados ou aplicativos a serem usados é muito menor que o custo do servidor mais dispendioso 402 que executa os jogos ou aplicativo de desempenho mais altos, ainda até mesmo os servidores de desempenho baixo 402, irão derivar benefícios de desempenho de disco dos arranjos RAID 405.

Adicionalmente, um servidor 402 no serviço de hospedagem 210 pode ser nada mais que uma placa mãe de PC sem um disco ou interfaces periféricas além de uma interface de rede e, no instante, podem ser integrado em um único chip com somente uma interface de rede rápida para SAN 403. Ademais, Os arranjos RAID 405 provavelmente serão compartilhados dentre muito mais usuários que discos, então, o custo do disk por usuário ativo será muito menor que uma unidade de disco. Todo este equipamento irá provavelmente residir em uma prateleira em um ambiente de servidor ambientalmente controlado. Se um servidor 402 falhar, isto pode ser prontamente reparado ou substituído no serviço de hospedagem 210. Adversamente, um PC ou console de jogo no lar ou escritório precisa ser um aparelho autônomo e resistente que precisa ser capaz de resistir a desgaste razoável contra choque ou queda, exigindo um alojamento, ter pelo menos uma unidade de disco, resistir condições ambientais adversas (por exemplo, ser expandido em um gabinete AV superaquecido com outra instalação), requer uma garantia de serviço, ser embalado e transportado, e vendido por um varejista que irá coletar um lucro de varejo. Adicionalmente, um PC ou console de jogo precisa ser configurado para satisfazer o desempenho pico da maioria dos jogos ou aplicativos antecipados computacionalmente intensos a serem usados no mesmo instante, apesar de jogos ou aplicativos com desempenho inferior (ou seções de jogos ou aplicativos) podem ser jogados a maior parte do tempo. Ademais, se o PC ou console falhar, é um processo dispendioso e demorado (que impacta adversamente no fabricante, usuário e desenvolvedor do software) repará-lo.

Dessa forma, posto que o sistema mostrado na figura 4a fornece uma experiência ao usuário comparável a de um recurso computacional local, para um usuário no lar, escritório ou escola para experimentar um determinado nível de capacidade computacional, é muito menos dispendioso fornecer esta capacidade computacional através da arquitetura mostrada na figura 4a.

Eliminação da Necessidade de Atualização

Adicionalmente, os usuários não precisam mais se preocupar sobre atualizar PCs e/ou consoles para jogar novos jogos ou manipular novos aplicativos de desempenho superior. Qualquer jogo ou aplicativos no serviço de hospedagem 210, independentemente de qual tipo de servidor 402 é requerido para aquele jogo ou aplicativos, está disponível para o usuário, e todos os jogos e aplicativos operam quase instantaneamente (por exemplo, carregamento rápido dos arranjos RAID 405 ou armazenamento local em servidores 402) e apropriadamente com as últimas atualizações e consertos de erro (isto é, desenvolvedores do software serão capazes de escolher uma configuração de servidor ideal para o(s) servidor(es) 402 que opera(m) um determinado jogo ou aplicativo e, então, configurar o(s) servidor(es) 402 com unidades ideais e, então, ao longo do tempo, os desenvolvedores serão capazes de fornecer atualizações, consertos de erro, etc. para todas as cópias do jogo ou aplicativo no serviço de hospedagem 210 de uma só vez). Na realidade, após o usuário iniciar o uso do serviço de hospedagem 210, é provável que o usuário ache que aqueles jogos e aplicativos continuam a fornecer uma melhor experiência (por exemplo, através de atualizações e/ou consertos de erro) e pode ser o caso em que um usuário descobre um ano depois que um novo jogo ou aplicativo se tornou disponível no serviço 210 que está utilizando a tecnologia computacional (por exemplo, uma GPU de desempenho superior) que não havia um ano antes, então, seria impossível que o usuário comprasse a tecnologia um ano antes para que jogasse o jogo ou executasse aplicativos um ano depois. Tendo em vista que o recurso computacional que está executando o jogo ou o aplicativo é invisível para o usuário (isto é, a partir da perspectiva do

usuário, o usuário está simplesmente selecionando um jogo ou aplicativo que começa a operar quase instantaneamente (como se o usuário tivesse trocado os canais em uma televisão), o hardware do usuário terá sido "atualizado" sem o usuário mesmo estando ciente da atualização.

Eliminação da Necessidade por Cópias de Segurança

Um outro problema principal para usuários em empresas, escolas e lares são as cópias de segurança. As informações armazenadas em um PC local ou console de vídeo game (por exemplo, no caso de um console, as conquistas e a classificação do jogo do usuário) podem ser perdidas se um disco falhar, ou se houver um apagamento inadvertido. Existem muitos aplicativos disponíveis que fornecem cópias de segurança manuais ou automáticas para PCs, e o estado do console de jogo pode ser transferido por upload para um servidor online para cópia de segurança, mas cópias de segurança locais são tipicamente copiadas para um outro disco local (ou outro dispositivo de armazenamento não volátil) que precisa ser armazenado em algum local seguro e organizado, e as cópias de segurança para serviços online são frequentemente limitadas, por causa da velocidade a montante lenta disponível através de conexões de Internet típicas de baixo custo. Com o serviço de hospedagem 210 da figura 4a, os dados que são armazenados em arranjos RAID 405 podem ser configurados com o uso das técnicas de configuração RAID da técnica anterior bem conhecida por aqueles elementos versados na técnica de tal modo que se um disco falhar, nenhum dado será perdido, e um técnico no centro do centro de servidor que aloja o disco falho será avisado e, então, irá substituir o disco, que, então, será automaticamente atualizado de modo que o arranjo RAID seja novamente tolerante á falha. Adicionalmente, uma vez que todas as unidades de disco estão próximas umas das outras e com redes locais rápidas entre elas através do SAN 403, não é difícil em um centro de servidor fazer com que todos os sistemas de disco tenham cópias de segurança em uma base regular para armazenamento secundário, que pode ser armazenado no centro de servidor ou relocado externamente. Do ponto de vista dos usuários de serviço de hospedagem 210, seus dados estão simplesmente seguros a

todo instante, e eles nunca precisam se preocupar com cópias de segurança.

Acesso a Demonstrações

Os usuários frequentemente desejam experimentar jogos ou aplicativos antes de comprá-los. Conforme descrito anteriormente, existem meios da técnica anterior através dos quais se demonstra jogos e aplicativos (a forma verbal de "demonstração" significa experimentar uma versão de demonstração, que também é chamada de "demonstração", mas como um substantivo), mas cada um deles sofre de limitações e/ou inconveniências. Com o uso do serviço de hospedagem 210, é fácil e conveniente para os usuários experimentar demonstrações. Na realidade, tudo que o usuário é selecionar a demonstração através de uma interface de usuário (tal como uma descrita abaixo) e experimentar a demo. A demonstração irá carregar quase instantaneamente sobre um servidor 402 apropriado para a demonstração, e irá operar justamente como qualquer outro jogo ou aplicativo. Se a demonstração requerer um servidor de desempenho muito alto 402, ou um servidor de desempenho baixo 402, e independentemente do tipo de cliente de lar ou escritório 415 o usuário está usando, a partir do ponto de vista do usuário, a demonstração irá somente funcionar. O divulgador do software da demonstração do jogo ou do aplicativo será capaz de controlar exatamente qual demonstração o usuário é permitido a experimentar e por quanto tempo e, obviamente, a demonstração pode incluir elementos de interface de usuário que oferecem ao usuário uma oportunidade de obter acesso a uma versão completa do jogo ou aplicativo demonstrado.

Uma vez que as demonstrações são propensas a serem oferecidas abaixo do custo ou livre de impostos, alguns usuários podem tentar usar demonstrações repetidas (particularmente, demonstrações de jogo, que podem ser divertidas para jogar repetidamente). O serviço de hospedagem 210 pode empregar várias técnicas para limitar o uso de demonstração para um determinado usuário. A abordagem mais fácil consiste em estabelecer um ID de usuário para cada usuário e limitar o número de vezes que um determinado ID de usuário é permitido a jogar uma demonstração. Um

usuário, entretanto, pode definir múltiplos IDs de usuário, especialmente, se forem gratuitos. Uma técnica para abordar este problema é limitar o número de vezes que um determinado cliente 415 é permitido a jogar uma demonstração. Se o cliente for dispositivo autônomo, então, o dispositivo terá um número de série, e o serviço de hospedagem 210 pode limitar o número de vezes que uma demonstração pode ser acessada por um cliente com aquele número de série. Se o cliente 415 estiver operando como software em um PC ou outro dispositivo, então, um número de série pode ser atribuído pelo serviço de hospedagem 210 e armazenado no PC e usado para limitar o uso de demonstração, mas tendo em vista que os PCs podem ser reprogramados pelos usuários, e o número de série apagado ou alterado, uma outra opção é que o serviço de hospedagem 210 mantenha um registro do endereço de Controle de Acesso de Mídia (MAC) do adaptador de rede do PC (e/ou outros identificadores específicos de máquina tais como números de série de discos rígidos, etc.) e limitar o uso de demonstração para isto. Posto que os endereços MAC dos adaptadores de rede podem ser alterados, entretanto, este não é um método à prova de falha. Uma outra abordagem consiste em limitar o número de vezes em que uma demonstração pode ser reproduzida para um determinado endereço de IP. Embora os endereços de IP possam ser periodicamente reatribuídos por provedores de cable modem e DSL, isto não acontece na prática muito frequentemente, e se puder ser determinado (por exemplo, através do contato com a ISP) que o IP está em um bloco de endereços de IP para acessos por DSL ou cable modem residenciais, então, um número pequeno de usos de demonstração pode tipicamente ser estabelecido para um determinado lar. Ademais, podem haver múltiplos dispositivos em um lar por trás de um roteador NAT que compartilha o mesmo endereço de IP, mas tipicamente em uma instalação residencial, há um número limitado de tais dispositivos. Se o endereço de IP estiver em um bloco que serve empresas, então, um número maior de demonstrações pode ser estabelecido para uma empresa. Mas, no fim, uma combinação de todas as abordagens anteriormente mencionadas é a melhor maneira de limitar o número de demonstrações em PCs. Embora

possa não haver uma maneira à prova de falha para que um determinado usuário tecnicamente adepto possa ser limitado no número de demonstrações reproduzidas repetidamente, criando um grande número de barreiras para criar um empecilho suficiente de tal modo que não seja válido para a maioria dos usuários de PC usuários abusar do sistema de demonstração, e, em vez disso, eles irão usar as demonstrações da forma pretendida: para experimentar novos jogos e aplicativos.

Benefícios para Escolas, Empresas e Outras Instituições

Benefícios significativos se acumulam particularmente para empresas, escolas e outras instituições que utilizam o sistema mostrado na figura 4a. As empresas e escolas têm custos substanciais associados à instalação, manutenção e atualização de PCs, particularmente, quando vêm de PCs para operar aplicativos de alto desempenho, tal como Maya. Conforme determinado anteriormente, os PCs são geralmente utilizados por somente uma fração de horas da semana, e, como no lar, o custo do PC com um determinado nível de capacidade de desempenho é muito maior em um ambiente de escritório ou escola que em um ambiente de centro de servidor.

No caso de empresas ou escolas maiores (por exemplo, grandes universidades), pode ser prático para os departamentos de TI de tais entidades definir centros do servidor e manter computadores que são remotamente acessados através conexões em LAN. Existem inúmeras soluções para acesso remoto de computadores por uma LAN ou através de conexão de largura de vídeo alta privada entre escritórios. Por exemplo, com o Windows Terminal Server da Microsoft, ou através de aplicativos computacionais de rede virtual como VNC, da RealVNC, Ltd., ou através de meios de cliente pequenos da Sun Microsystems, os usuários podem obter acesso remoto para PCs ou servidores, com uma faixa de qualidade em tempo de resposta de gráficos e experiência de usuário. Adicionalmente, tais centros de servidores autogerenciados são tipicamente dedicados para uma única empresa ou escola e como tal, são incapazes de levar vantagem da sobreposição de uso que é possível quando aplicativos distintos (por

exemplo, aplicativos de entretenimento e negócios) utilizam os mesmos recursos computacionais em diferentes momentos da semana. Então, muitas empresas e escolas perdem a escala, recursos ou perícia para configurar um centro de servidor propriamente que tenha uma conexão de rede com velocidade de LAN para cada usuário. Na realidade, uma grande percentual de escolas e empresas tem as mesmas conexões de Internet (por exemplo, DSL, cable modems) que os lares.

Ainda, tais organizações podem ter a necessidade de computação de desempenho muito alto, em uma base regular ou em uma base periódica. Por exemplo, uma pequena empresa de arquitetura pode ter somente um número pequeno de arquitetos, com necessidades computacionais relativamente modestas quando se faz trabalho de projeto, mas pode requerer computação 3D com desempenho muito alto periodicamente (por exemplo, quando se cria uma apresentação flutuante em 3D de um novo projeto arquitetônico para um cliente). O sistema mostrado na figura 4a é extremamente bem adequado para tais organizações. As organizações precisam de nada mais que o mesmo tipo de conexão de rede que é oferecido para lares (por exemplo, DSL, cable modems) e é tipicamente muito barato. Eles podem utilizar PCs baratos como o cliente 415 ou dispensar todos os PCs e utilizar dispositivos dedicados baratos que implantam simplesmente a lógica de sinal de controle 413 e descompressão de vídeo de baixa latência 412. Estes recursos são particularmente atrativos para escolas que podem ter problemas com furto de PCs ou dano aos componentes delicados no interior dos PCs.

Tal disposição resolve inúmeros problemas para tais organizações (e muitas destas vantagens também são compartilhadas por usuários domésticos de computação para fins gerais). Primeiramente, o custo operacional (que definitivamente precisa ser superado de alguma forma pelos usuários a fim de ter um negócio viável) pode ser muito menor, devido ao fato de que (a) os recursos computacionais são compartilhados com outros aplicativos que têm diferentes tempos de uso pico durante a semana, (b) as organizações podem obter acesso a (e incorrer o custo de) recursos

computacionais de alto desempenho somente quando necessário, (c) as organizações não precisam ter que fornecer recursos para cópia de segurança ou, de outro modo, manutenção de recursos computacionais de alto desempenho.

Eliminação de Pirataria

Além disso, jogos, aplicativos, filmes interativos, etc., não podem ser mais pirateados como hoje em dia. Devido ao fato de que cada jogo é armazenado e executado no serviço de hospedagem 210, os usuários não são dotados de acesso ao código de programa subjacente, então, não há nada para piratear. Mesmo se um usuário estiver copiando o código fonte, o usuário não seria capaz de executar o código em um console de jogo ou computador domestico padrão. Isto abre mercado em locais do mundo tal como China, onde o vídeo game padrão não é disponível. A revenda de jogos usados também não é possível, pois não existem cópias de jogos distribuídos para usuários.

Para desenvolvedores de jogos, existem menos descontinuidades de mercado como é o caso hoje em dia quando novas gerações de consoles de jogo ou PCs são introduzidas no mercado. O serviço de hospedagem 210 pode ser gradualmente atualizado com a tecnologia computacional mais avançada ao longo do tempo como alteração de requisitos de jogo, em contraste à situação atual em que uma geração completamente nova de tecnologia de console ou PC força os usuários e os desenvolvedores a atualizarem e o desenvolvedor do jogo é dependente da distribuição periódica da plataforma de hardware para o usuário (por exemplo, no caso do PlayStation 3, sua introdução foi atrasada em mais de um ano, e os desenvolvedores tiveram que esperar até estivesse disponível e números significativos de unidades foram compradas).

Transmissão de Vídeo Interativa

As descrições acima fornecem uma ampla faixa de aplicativos viabilizados pelo conceito subjacente inovador de transmissão de baixa latência baseada em Internet geral de vídeo interativo (que implicitamente inclui áudio junto com o vídeo, conforme usado no presente documento). Os

sistemas da técnica anterior que forneceram transmissão de vídeo através da Internet têm somente aplicativos viabilizados que podem ser implantados interações de alta latência. Por exemplo, os controles de reprodução básicos para vídeo linear (por exemplo, pausa, retrocesso, avanço rápido) funcionam adequadamente com alta latência, e é possível selecionar dentre as alimentações de vídeo linear. Ademais, conforme determinado anteriormente, a natureza de alguns vídeo games os permite que sejam executados com alta latência. Porém, a alta latência (ou baixa razão de compressão) das abordagens da técnica anterior para transmissão de vídeo tem limitado gravemente os aplicativos potenciais de transmissão de vídeo ou estreitado suas disposições em ambientes de rede especializados e, mesmo em tais ambientes, as técnicas anteriores introduzem sobrecargas substanciais nas redes. A tecnologia descrita no presente documento abre a porta para a ampla faixa de aplicativos possíveis com transmissão de vídeo de baixa latência de vídeo interativo através da Internet, particularmente, aqueles viabilizados através de conexões de Internet de grau de consumidor.

Na realidade, com os dispositivos do cliente tão pequenos quanto o cliente 465 da figura 4c suficientes para fornecer uma experiência de usuário acentuada com uma quantidade arbitrária efetivamente de potência computacional, quantidade arbitraria de armazenamento rápido e rede extremamente rápida dentre servidores potentes, é viável uma nova era de computação. Adicionalmente, devido ao fato de que os requisitos de largura de vídeo não crescem conforme a potência computacional do sistema cresce (isto é, devido ao fato de os requisitos de largura de vídeo estarem somente atrelados à resolução, qualidade e taxa de quadro de exibição), uma vez que a conectividade de Internet de banda larga é onipresente (por exemplo, através de cobertura sem fio de baixa latência expandida), confiável e de largura de vídeo suficientemente alta para satisfazer as necessidades dos dispositivos de exibição 422 de todos os usuários, a questão será se os clientes grandes (tais como PCs ou telefone móvel que executam Windows, Linux, OSX, etc.) ou ainda os clientes pequenos (tal como Adobe Flash ou Java) são necessários para aplicativos corporativos e de consumidor típicos.

O advento de vídeo interativo de fluxo contínuo resulta na reformulação de suposições acerca da estrutura de arquiteturas da computação. Um exemplo disso é a modalidade de centro de servidor de serviço de hospedagem 210 mostrada na figura 15. O caminho de vídeo para armazenamento temporário de atraso e/ou vídeo de grupo 1550 é um laço de retorno em que a saída de vídeo interativo de fluxo contínuo de multicast dos servidores de aplicativo/jogos 1521 a 1525 é fornecida como retorno nos servidores de aplicativos/jogos 1521 a 1525 ou em tempo real através de um caminho 1552 ou após um atraso selecionável através do caminho 1551. Isso permite uma ampla variedade de aplicações práticas (por exemplo, como aquelas ilustradas nas figuras 16, 17 e 20) quer seriam ou impossíveis ou irrealizáveis pelo servidor ou arquiteturas computacionais da técnica anterior. Porém, como uma característica arquitetônica mais geral, o que o laço de retorno 1550 fornece é a recursão no nível de vídeo interativo de fluxo contínuo, já que o vídeo pode ser enlaçado novamente indefinidamente conforme o aplicativo solicita do mesmo. Isso permite uma ampla variedade de possibilidades de aplicação não disponíveis anteriormente.

Outra característica arquitetônica chave é que os fluxos de vídeo são fluxos de UDP unidirecionais. Isso permite efetivamente um grau arbitrário de multicast de vídeo interativo de fluxo contínuo (em contraste, fluxos de duas vias, como fluxos TCP/IP, criariam cada vez mais obstruções de log de tráfego nas redes a partir das comunicações de ida e volta, conforme o número de usuários aumentava). Multicasting é uma capacidade importante no centro de servidor porque permite que o sistema seja responsivo às necessidades crescentes dos usuários da Internet (e, de fato, da população mundial) de comunicar em uma base de um para muitos ou mesmo de muitos para muitos. Novamente, os exemplos discutidos aqui, como na figura 16, que ilustra o uso de ambos a recursão de vídeo interativo de fluxo contínuo e o multicast, ilustram apenas a ponta de um grande iceberg de possibilidades.

PEERING DE NÃO TRÂNSITO

Em uma modalidade, o serviço de hospedagem 210 tem uma ou

mais conexões de peering a um ou mais Provedores de Serviço de Internet (ISP) que também fornecem serviço de Internet a usuários, e, dessa forma, o serviço de hospedagem 210 também pode ser capaz de comunicar com o usuário através de uma rota de não-trânsito que fica na rede daquele ISP. Por exemplo, se o serviço de hospedagem 210 Interface de WAN 441 conectado diretamente à rede de Comcast Cable Communications, Inc., e as premissas de usuário 211 foram supridas com serviço de banda larga com um modem a cabo Comcast, uma rota entre o serviço de hospedagem 210 e o cliente 415 pode ser estabelecida totalmente na rede de Comcast. As vantagens potenciais disso seria incluir um custo mais baixo para as comunicações (já que os custos de trânsito de IP entre duas ou mais redes ISP podem ser evitadas), uma conexão potencialmente mais confiável (no caso, havia congestão ou outras interrupções de trânsito entre redes ISP), e latência inferior (no caso, havia congestão, rotas ineficientes ou outros atrasos entre redes de ISP).

Nessa modalidade, quando o cliente 415 inicialmente entra em contato com o serviço de hospedagem 210 no início de uma sessão, o serviço de hospedagem 210 recebe o endereço de IP das premissas de usuário 211. Então, usa tabelas de endereço de IP disponíveis, por exemplo, a partir de ARIN (Registro Americano para Números da Internet), para verificar se o endereço de IP é aquele alocado a um ISP particular conectado ao serviço de hospedagem 210 que pode levar às premissas do usuário 211 sem o trânsito de IP até outro ISP. Por exemplo, se o endereço de IP estava entre 76.21.0.0 e 76.21.127.255, então o endereço de IP é designado à Comcast Cable Communications, Inc. Nesse exemplo, se o serviço de hospedagem 210 mantiver as conexões aos ISPs de Comcast, AT&T e Cox, então seleciona Comcast como o ISP mais provável a fornecer um caminho ótimo ao usuário particular.

COMPRESSÃO DE VÍDEO COM O USO DE RETORNO

Em uma modalidade, o retorno é fornecido a partir do dispositivo de cliente para o serviço de hospedagem indicar ladrilho bem sucedido (ou malsucedido) e/ou entrega de quadro. As informações de retorno fornecidas

a partir do cliente são, então, usadas para ajustar as operações de compressão de vídeo no serviço de hospedagem.

Por exemplo, as **Figures 25a a b** ilustram uma modalidade da invenção na qual um canal de retorno 2501 é estabelecido entre o dispositivo do cliente 205 e o serviço de hospedagem 210. O canal de retorno 2501 é usado pelo dispositivo de cliente 205 para enviar reconhecimentos empacotados de ladrilhos/quadros recebidos com sucesso e/ou indicações de ladrilhos/quadros recebidos sem sucesso.

Em uma modalidade, depois de receber com sucesso cada ladrilho/quadro, o cliente transmite uma mensagem de reconhecimento ao serviço de hospedagem 210. Nessa modalidade, o serviço de hospedagem 210 detecta uma perda de pacote se não receber um reconhecimento após um período de tempo especificado e/ou se receber um reconhecimento de que o dispositivo de cliente 205 recebeu um ladrilho/quadro subsequente ao invés do que foi enviado. Alternativamente, ou adicionalmente, o dispositivo de cliente 205 pode detectar a perda de pacote e transmitir uma indicação de perda de pacote ao serviço de hospedagem 210 junto com uma indicação dos ladrilhos/quadros afetados pela perda de pacote. Nessa modalidade, o reconhecimento contínuo de ladrilhos/quadros entregues com sucesso não é exigido.

Independentemente da maneira como uma perda de pacotes é detectada, na modalidade ilustrada nas **figuras 25a a 25b**, após a geração de um conjunto inicial de I-ladrilhos para uma imagem (não mostrada na **figura 25a**), o encodificador gera subsequentemente apenas P-ladrilhos até que uma perda de pacotes seja detectada. Observe que na **figura 25a**, cada quadro, como o 2510, é ilustrado na forma de 4 ladrilhos verticais. O quadro pode ser ladrilhado em uma configuração diferente, como 2x2, 2x4, 4x4 etc., ou o quadro pode ser totalmente encodificado sem ladrilhos (isto é, como 1 ladrilho grande). Os exemplos precedentes de configurações de ladrilhagem de quadro são fornecidos para o propósito de ilustração desta modalidade da invenção. Os princípios fundamentais da invenção não são limitados a qualquer configuração de ladrilhagem de quadro particular.

A transmissão de apenas P-ladrilhos reduz os requisitos de largura de banda do canal por todas as razões apresentadas acima (isto é, P-ladrilhos são geralmente menores que I-ladrilhos). Quando uma perda de pacotes é detectada por meio do canal de retorno 2501, novos I-ladrilhos são gerados pelo encodificador 2500, conforme ilustrado na **figura 25b,** para reinicializar o estado do decodificador 2502 no dispositivo do cliente 205. Conforme ilustrado, em uma modalidade, os I-ladrilhos são distribuídos ao longo de múltiplos quadros encodificados para limitar a largura de banda consumida por cada quadro encodificado individual. Por exemplo, na **figura 25**, em que cada quadro inclui 4 ladrilhos, um único I-ladrilho é transmitido em uma posição diferente dentro de 4 quadros encodificados sucessivos.

O encodificador 2500 pode combinar as técnicas descritas em relação a esta modalidade com outras técnicas de encodificação descritas na presente invenção. Por exemplo, além de gerar I-ladrilhos em resposta a uma perda de pacotes detectada, o encodificador 2500 pode gerar I-ladrilhos em outras circunstâncias em que os I-ladrilhos podem ser benéficos para reproduzir de maneira apropriada a sequência de imagens (como em resposta a transições de cena repentinas).

A **figura 26a** ilustra outra modalidade da invenção que se baseia em um canal de retorno 2601 entre o dispositivo do cliente 205 e o serviço de hospedagem 210. Em vez de gerar novos I-ladrilhos/quadros em resposta a uma perda de pacotes detectada, o encodificador 2600 desta modalidade ajusta as dependências dos P-ladrilhos/quadros. Como um assunto inicial, deve ser observado que os detalhes específicos apresentados neste exemplo não são necessários para o cumprimento dos princípios fundamentais da invenção. Por exemplo, embora este exemplo seja descrito com o uso de P-ladrilhos/quadros, os princípios fundamentais da invenção não são limitados a nenhum formato de encodificação particular.

Na **figura 26a**, o encodificador 2600 encodifica uma pluralidade de ladrilhos/quadros descomprimidos 2605 em uma pluralidade de P-ladrilhos/quadros 2606 e transmite os P-ladrilhos/quadros por meio de um canal de comunicação (por exemplo, a Internet) para um dispositivo do

cliente 205. Um decodificador 2602 no dispositivo do cliente 205 decodifica os P-ladrilhos/quadros 2606 para gerar uma pluralidade de ladrilhos/quadros descomprimidos 2607. O(s) estado(s) anterior(es) 2611 do encodificador 2600 é/são armazenado(s) em um dispositivo de memória 2610 no serviço de hospedagem 210 e o(s) estado(s) anterior(es) 2621 do decodificador 2602 é/são armazenado(s) em um dispositivo de memória 2620 no dispositivo do cliente 205. O "estado" de um decodificador é um termo bem conhecido na técnica de sistemas de codificação de vídeo, como MPEG-2 e MPEG-4. Em uma modalidade, o(s) "estado(s)" anterior(es) armazenado(s) nas memórias compreende(m) os dados combinados de P-ladrilhos/quadros anteriores. As memórias 2611 e 2621 podem ser integradas no encodificador 2600 e no decodificador 2602, respectivamente, em vez de serem separadas do encodificador 2600 e do decodificador 2602, conforme mostrado na **figura 26a**. Além disso, vários tipos de memórias podem ser usados, incluindo, a título de exemplo e não de limitação, memória de acesso aleatório.

Em uma modalidade, quando não ocorre perda de pacotes, o encodificador 2600 encodifica cada P-ladrilho/quadro para que seja dependente do P-ladrilho/quadro anterior. Dessa maneira, conforme indicado pela notação usada na **figura 26a**, o P-ladrilho/quadro 4 é dependente do P-ladrilho/quadro 3 (identificado com o uso da notação 4_3); o P-ladrilho/quadro 5 é dependente do P-ladrilho/quadro 4 (identificado com o uso da notação 5_4); e o P-ladrilho/quadro 6 é dependente do P-ladrilho/quadro 5 (identificado com o uso da notação 6_5). Neste exemplo, o P-ladrilho/quadro 4_3 foi perdido durante a transmissão entre o encodificador 2600 e o decodificador 2602. A perda pode ser comunicada ao encodificador 2600 de várias maneiras, incluindo, sem limitação, aquelas descritas acima. Por exemplo, cada vez que o decodificador 2606 recebe e/ou decodifica com êxito um ladrilho/quadro, essas informações podem ser comunicadas pelo decodificador 2602 ao encodificador 2600. Se o encodificador 2600 não receber uma indicação de que um ladrilho/quadro particular foi recebido e/ou decodificado após um período de tempo, o encodificador 2600 irá admitir que o ladrilho/quadro não foi recebido com êxito. Alternativamente, ou além disso, o decodificador

2602 pode notificar o encodificador 2600 quando um ladrilho/quadro particular não for recebido com êxito.

Em uma modalidade, independente de como o ladrilho/quadro perdido for detectado, uma vez que este o é, o encodificador 2600 encodificada o próximo ladrilho/quadro que usa o último ladrilho/quadro conhecido como tendo sido recebido com sucesso pelo decodificador 2602. No exemplo mostrado na **figura 26a**, os ladrilhos/quadros 5 e 6 não são considerados como tendo sido "recebidos com sucesso" porque estes não podem ser apropriadamente decodificados pelo decodificador 2602 devido a perda de ladrilho/quadro 4 (isto é, a decodificação de ladrilho/quadro 5 depende do ladrilho/quadro 4 e a decodificação de ladrilho/quadro 6 depende do ladrilho/quadro 5). Portanto, no exemplo mostrado na **figura 26a**, o encodificador 2600 encodifica o ladrilho/quadro 7 para ser dependente do ladrilho/quadro 3 (o último ladrilho/quadro recebido com sucesso) ao invés de ladrilho/quadro 6 que o decodificador 2602 não pode ser decodificar apropriadamente. Embora não ilustrado na **figura 26a**, o ladrilho/quadro 8 irá ser subsequentemente encodificado para ser dependente do ladrilho/quadro 7 e o ladrilho/quadro 9 irá ser encodificado para ser dependente do ladrilho/quadro 8, assumindo que nenhuma perda adicional de pacote seja detectada.

Conforme mencionado acima, tanto o encodificador 2600 quanto o decodificador 2602 mantém os estados passados de encodificador e decodificador, 2611 e 2621, dentro de memórias 2610 e 2620, respectivamente. Portanto, quando encodificando o ladrilho/quadro 7, o encodificador 2600 recupera o estado anterior do encodificador associado com o ladrilho/quadro 3 da memória 2610. De maneira similar, a memória 2620 associada com decodificador 2602 armazena pelo menos o último conhecido estado bom de decodificador (o estado associado com P-telha/quadro 3 no exemplo). Consequentemente, o decodificador 2602 recupera a informação de estado passado associada com o ladrilho/quadro 3 de modo que o ladrilho/quadro 7 possa ser decodificado.

Como um resultado das técnicas descritas acima, o vídeo inte-

rativo, de baixa latência e em tempo real pode ser encodificado e transmitido usando uma largura de banda relativamente pequena porque nenhum I-ladrilho/quadro é requerido (exceto para inicializar o decodificador e o encodificador no início da transmissão). Ademais, enquanto a imagem de vídeo produzida pelo decodificador pode incluir temporariamente distorções não desejadas que resultam do ladrilho/quadro perdido 4 e os ladrilhos/quadros 5 e 6 (que não podem ser apropriadamente decodificados devido a perda de ladrilho/quadro 4), essa distorção será visível por uma duração muita curta. Ademais, se os ladrilhos forem usados (ao invés de quadros de vídeo completo), a distorção irá ser limitada a uma região em particular da imagem de vídeo renderizada.

Um método de acordo com uma modalidade da invenção é ilustrado na **figura 26b**. Em 2650, um ladrilho/quadro é gerado com base em um ladrilho/quadro gerado previamente. Em 2651, um ladrilho/quadro perdido é detectado. Em uma modalidade, o ladrilho/quadro perdido é detectado com base na informação comunicada do encodificador para o decodificador, como descrito acima. Em 2652, o próximo ladrilho/quadro é gerado com base em um ladrilho/quadro que é conhecido como tendo sido recebido com sucesso e/ou decodificado no decodificador. Em uma modalidade, o encodificador gera o próximo ladrilho/quadro carregando o estado associado com o ladrilho/quadro da memória recebido com sucesso e/ou decodificado. De maneira similar, quando o decodificador recebe o novo ladrilho/quadro, este decodifica o ladrilho/quadro carregando o estado associado com o ladrilho/quadro da memória recebido com sucesso e/ou decodificado.

Em uma modalidade o próximo ladrilho/quadro é gerado com base no último ladrilho/quadro recebido com sucesso e/ou decodificado no encodificador. Em outra modalidade, o próximo ladrilho/quadro gerado é um I-ladrilho/quadro. Em ainda outra modalidade, a escolha de quando gerar o próximo ladrilho/quadro com base em um ladrilho/quadro previamente recebido com sucesso ou como um I quadro é com base em quantos ladrilhos/quadros foram perdidos e/ou a latência do canal. Em uma situação em que um número relativamente pequeno (por exemplo, 1 ou 2) de ladri-

lhos/quadros são perdidos e a latência de ida e volta é relativamente baixa (por exemplo 1 ou 2 quadro por vez), então este pode ser ótima para gerar um P-ladrilho/quadro desde que a diferença entre o último ladrilho/quadro recebido com sucesso e o gerado recentemente pode ser relativamente pequeno. Se diversos ladrilhos/quadros forem perdidos ou a latência de ida e volta for alta, então está pode ser ótima para gerar um I-ladrilho/quadro desde que a diferença entre o último ladrilho/quadro recebido com sucesso e o gerado recentemente pode ser grande. Em uma modalidade, um limite de perda ladrilho/quadro e/ou um valor de limite de latência é ajustado para determinar se deve transmitir um I-ladrilho/quadro ou um P-ladrilho/quadro. Se o número de ladrilhos/quadros perdidos for abaixo do limite de perda de ladrilho/quadro e/ou se a latência de ida e volta for abaixo da do valor limite de latência, então um novo I-ladrilho/quadro é gerado; de outra forma, um novo P-ladrilho/quadro é gerado.

Em uma modalidade, o encodificador sempre tenta gerar um P-ladrilho/quadro relativo ao último ladrilho/quadro recebido com sucesso, e se no processo de encodificação o encodificador determinar que o P-ladrilho/quadro irá ser provavelmente maior que um I-ladrilho/quadro (por exemplo se este comprimiu 1/8 do ladrilho/quadro e o tamanho comprimido for maior que 1/8 do tamanho do I-ladrilho/quadro médio comprimido anteriormente), então o encodificador abandonará a compressão do P-ladrilho/quadro e irá comprimir um I-ladrilho/quadro.

Se os pacotes perdidos ocorrem raramente, os sistemas descrita acima com o uso de retroalimentação para relatar uma ladrilho/quadro solto resulta tipicamente em uma perturbação muito ligeira no fluxo de vídeo para o usuário devido ao fato de que um ladrilho/quadro que foi perturbado por um pacote perdido é substituído aproximadamente no tempo de um ciclo entre o dispositivo do cliente 205 e o serviço de hospedagem 210 presumindo que o codificador 2600 compreende o ladrilho/quadro em um curto período de tempo. E, devido ao fato de que o novo ladrilho/quadro, que é comprimido, é baseado m um último quadro no fluxo de vídeo descomprimido, o fluxo de vídeo não fica para trás do fluxo de vídeo não comprimido.

Porém, sem um pacote que contém o novo ladrilho/quadro também for perdido, então isto resulta em um atraso de pelo menos dois ciclos para solicitar e enviar novamente outro novo ladrilho/quadro, que, em situações práticas, resultará em perturbação notável para o fluxo de vídeo. Como consequência, é muito importante que o ladrilho/quadro recém-codificado enviado após o ladrilho/quadro ignorado ser enviado com sucesso do serviço de hospedagem 210 para o dispositivo do cliente 205.

Em uma modalidade, técnicas de codificação de correção de erros de repasse (FEC), como aquelas descritas anteriormente e ilustradas nas figuras 11a, 11b, 11c e 11d, são usadas para mitigar a probabilidade de perda do ladrilho/quadro recém-codificado. Se a codificação de FEC já estiver sendo usada na transmissão de ladrilhos/quadros, então um código de FEC mais forte é usado para o ladrilho/quadro recém-codificado.

Uma causa potencial de pacotes ignorados é uma perda repentina na largura de banda de canal, por exemplo, se algum ouro usuário da conexão de banda larga nas premissas de usuário 211 inicia com o uso de uma grande quantidade de largura de banda. Se um ladrilho/quadro recém-gerado também for perdido devidos aos pacotes ignorados (mesmo se for usado FEC), então, em uma modalidade quando o serviço de hospedagem 210 for notificado pelo cliente 415 que um segundo ladrilho/quadro recentemente codificado foi ignorado, o compressor de vídeo 404 reduz a taxa de dados quando codifica um ladrilho/quadro recém-codificado subsequente. Diferentes modalidades reduzem a taxa de dados com o uso de diferentes técnicas. Por exemplos, em uma modalidade, esta redução da taxa de dados é realizada ao baixar a qualidade do ladrilho/quadro codificado através do aumento da razão de compressão. Em outra modalidade, a taxa de dados é reduzida ao baixar a taxa de quadro do vídeo (por exemplo, de 60fps para 30fps) e, consequentemente, baixar a taxa de transmissão de dados. Em uma modalidade, ambas as técnicas para reduzir a taxa de dados são usadas (por exemplo, reduzindo a taxa de quadro e aumentando a razão de compressão). Se essa taxa inferior de transmissão de dados obter sucesso na mitigação dos pacotes ignorados, então, de acordo com a detecção da

taxa de dados de canal and métodos de ajuste anteriormente descrita, o serviço de hospedagem 210 continuará a codificar a uma taxa de dados menor e, então, ajustar gradualmente a taxa de dados par cima ou para baixo conforme o canal permitir. A recepção contínua de dados de retroalimentação relacionados aos pacotes ignorados e/ou à latência permite que o serviço de hospedagem 210 ajuste dinamicamente a taxa de dados baseado nas condições atuais do canal.

GERENCIAMENTO DE ESTADO EM UM SISTEMA DE JOGO ONLINE

Uma modalidade da invenção emprega técnicas para armazenar de forma eficaz e portar o estado atual de um jogo ativo entre servidores. Embora as modalidades descritas no presente documento estejam relacionadas ao jogo online, os princípios subjacentes da invenção podem ser usados para vários outros tipos de aplicações (por exemplo, aplicativos de modelo, processadores de palavras, software de comunicação como correio eletrônico ou mensagem instantânea etc). A **figura 27a** ilustra uma arquitetura de sistema exemplificador para a implantação desta modalidade e a **figura 27b** ilustra um método exemplificador. Embora o método e a arquitetura de sistema sejam descritos concorrentemente, o método ilustrado na **figura 27b** não se limita a nenhuma arquitetura de sistema particular.

Em 2751 da **figura 27b**, um usuário inicia um novo jogo online game em um serviço de hospedagem 210a de um dispositivo do cliente 205. Em resposta, em 2752, uma imagem "limpa" do jogo 2702a é carregada do armazenamento (por exemplo, um disco rígido, ou conectado diretamente a um servidor que executa o jogo, ou conectado a um servidor através de uma rede) à memória (por exemplo, RAM) no serviço de hospedagem 210a. A imagem "limpa" compreende o código de programa de tempo de execução e dados para o jogo antes da iniciação de qualquer jogo (por exemplo, como quando o jogo é executado pela primeira vez). O usuário joga então o jogo em 2753, fazendo com que a imagem "limpa" mude para uma imagem não-limpa (por exemplo, um jogo em execução representado por "Estado A" na **figura 27a**). Em 2754, o jogo é interrompido ou finalizado, ou pelo usuário ou pelo serviço de hospedagem 210a. Em 2755, a lógica de gerenciamento de

estado 2700a no serviço de hospedagem 210a determina as diferenças entre a imagem "limpa" do jogo e o estado de jogo atual ("Estado A"). Várias técnicas conhecidas podem ser usadas para calcular a diferença entre duas imagens binárias incluindo, por exemplo, aquelas usadas no serviço "diff" conhecido disponível no sistema operacional Unix. Certamente, os princípios subjacentes da invenção não se limitam a nenhuma técnica particular para o cálculo da diferença.

Independente de como as diferenças são calculadas, uma vez que as mesmas tenham sido feitas, os dados diferenciais são armazenados localmente dentro de um dispositivo de armazenamento 2705a e/ou transmitidos para um serviço de hospedeiro diferente 210b. Se transmitidos para um serviço de hospedeiro diferente 210b, os dados diferenciais podem ser armazenados em um dispositivo de armazenamento (não mostrado) no novo serviço de hospedeiro 210b. Em ambos os casos, os dados diferenciais são associados com a conta do usuário no serviço de hospedeiros de modo que os mesmos possam ser identificados na próxima vez em que o usuário realiza o login no serviço de hospedeiros e inicia o jogo. Em uma modalidade, ao invés de serem transmitidos imediatamente, os dados diferenciais não são transmitidos para um novo serviço de hospedeiro até a próxima vez em que o usuário tenta jogar o jogo (e um serviço de hospedeiro diferente é identificado como a melhor escolha para hospedar o jogo).

Retornando ao método mostrado na **figura 27b**, em 2757, o usuário reinicia o jogo a partir de um dispositivo cliente, que pode ser o mesmo dispositivo cliente 205 a partir do qual o usuário inicialmente jogou o jogo ou um dispositivo cliente diferente (não mostrado). Em resposta, em 2758, lógica de gerenciamento de estado 2700b no serviço de hospedeiro 210b recupera a imagem "limpa" do jogo a partir de um dispositivo de armazenamento e dos dados diferenciais. Em 2759, a lógica de gerenciamento de estado 2700b combina a imagem limpa e dados de diferença para reconstruir o estado em que o jogo estava no serviço de hospedeiro original 210a ("Estado A"). Diversas técnicas conhecidas podem ser usadas para recriar o estado de uma imagem binária usando os dados diferenciais incluindo, por

exemplo, aqueles usados do bem conhecido função "remendo" disponível no sistema operacional Unix. As técnicas de cálculo de diferenças usadas em programas de cópia de segurança bem conhecidos como PC Backup também podem ser usados. Os princípios fundamentais da invenção não se limitam a qualquer técnica em particular ao usar dados de diferença para recriar a imagem binária.

Além disso, em 2760, dados dependentes de plataforma 2710 são incorporados na imagem de jogo final 2701b. Os dados dependentes de plataforma 2710 podem incluir quaisquer dados que sejam exclusivos para a plataforma do servidor de destino. A título de exemplo, e não de limitação, os dados dependentes de plataforma 2710 podem incluir o endereço de Controle de Acesso ao Meio (MAC) da nova plataforma, o endereço TCP/IP, a hora do dia, números seriais de hardware (por exemplo, para o disco rígido e CPU), endereços de servidor de rede (por exemplo, servidores DHCP/Wins), e número(s) serial(is) / código(s) de ativação de software (incluindo número(s) serial(is) / código(s) de ativação do Sistema Operacional).

Outros dados dependentes de plataforma relacionados ao cliente/usuário podem incluir (mas não se limitam a) os seguintes:

1. A resolução de tela do usuário. Quando o usuário retoma o jogo, o usuário pode estar usando um dispositivo diferente com uma resolução diferente.

2. A configuração de controle do usuário. Quando o jogo é retomado, o usuário pode ter trocado de um controle de jogo para um teclado/mouse.

3. Direitos do usuário, como se uma taxa de desconto expirou (por exemplo, se o usuário estava jogando o jogo durante um período promocional e está agora jogando durante um período normal com custos mais altos) ou se o usuário ou dispositivo tem algumas restrições de idade (por exemplo, os pais do usuário podem ter mudado as configurações para uma criança, de modo que a criança não seja permitida a ver material adulto, ou se o dispositivo onde o jogo é jogado (por exemplo, um computador em uma biblioteca pública) tem certas restrições sobre se o material adulto pode

ser exibido).

4. A classificação do usuário. O usuário pode ter sido permitido a jogar um jogo de múltiplos jogadores em uma certa liga, mas devido ao fato de alguns outros usuários terem excedido a classificação do usuário, o usuário pode ter sido rebaixado para uma liga inferior.

Os exemplos precedentes de dados dependentes de plataforma 2710 são fornecidos para fins de ilustração desta modalidade da invenção. Os princípios fundamentais da invenção não se limitam a qualquer conjunto particular de dados dependentes de plataforma.

A **figura 28** ilustra graficamente como a lógica de gerenciamento de estado 2700a no primeiro serviço de hospedeiro extrai dados de diferença 2800 a partir do jogo sendo executado 2701a. A lógica de gerenciamento de estado 2700b no segundo serviço de hospedeiro então combina a imagem limpa 2702b com os dados diferenciais 2800 e dados dependentes de plataforma 2710 para regenerar o estado do jogo sendo executado 2701b. Conforme mostrado em geral na **figura 28**, o tamanho dos dados diferenciais é significativamente menor do que o tamanho de toda a imagem do jogo 2701a e, consequentemente, uma quantidade significativa de espaço de armazenamento e a largura de banda é conservada através do armazenamento/transmissão de somente dados de diferença. Embora não mostrado na **figura 28**, os dados dependentes de plataforma 2700 podem sobrescrever alguns dos dados diferenciais quando os mesmos são incorporados na imagem de jogo final 2701b.

Embora uma implementação de videogame online seja descrita acima, os princípios fundamentais da invenção não se limitam a videogames. Por exemplo, as técnicas de gerenciamento de estado precedentes podem ser implementadas dentro do contexto de qualquer tipo de aplicativo hospedado online.

TÉCNICAS PARA MANTER UM DECODIFICADOR DE CLIENTE

Em uma modalidade da invenção, o serviço de hospedeiro 210 transmite um novo decodificador para o dispositivo de cliente 205 cada vez que o usuário solicita uma conexão com o serviço de hospedeiro 210. Con-

sequentemente, nessa modalidade, o decodificador usado pelo dispositivo de cliente é sempre atualizado e unicamente adaptado ao hardware/software implantado no dispositivo de cliente.

Conforme ilustrado na **figura 29**, nessa modalidade, o aplicativo que é permanentemente instalado no dispositivo de cliente 205 não inclui um decodificador. Especialmente, esse é um aplicativo de transferência por transferência por download de cliente 2903 que gerencia a transferência por download e instalação de um decodificador temporário 2900 cada vez que o dispositivo de cliente 205 se conecta ao serviço de hospedeiro 210. O aplicativo de transferência por download 2903 pode ser implantado em hardware, software, firmware ou em qualquer combinação dos mesmos. Em resposta a uma solicitação do usuário para uma nova sessão online, o aplicativo de transferência por download 2903 transmite informações relacionadas ao dispositivo de cliente 205 sobre uma rede (por exemplo, a Internet). As informações podem incluir dados de identificação que identificam o dispositivo de cliente e/ou a configuração de hardware/software do dispositivo de cliente (por exemplo, processador, sistema de operação, etc).

Com base nessas informações, um aplicativo de transferência por download 2901 no serviço de hospedeiro 210 seleciona um decodificador temporário apropriado 2900 para ser usado no dispositivo de cliente 205. O aplicativo de transferência por download 2901 no serviço de hospedeiro, então, transmite o decodificador temporário 2900 e o aplicativo de transferência por download 2903 no dispositivo de cliente, verifica e/ou instala o decodificador no dispositivo de cliente 205. O encodificador 2902, então, encodifica o conteúdo de áudio/vídeo usando qualquer uma das técnicas descritas no presente documento e transmite o conteúdo 2910 para o decodificador 2900. Uma vez que o novo decodificador 2900 é instalado, esse decodifica o conteúdo para a sessão online atual (isto é, usando uma ou mais das técnicas de descompressão de áudio/vídeo descritas no presente documento). Em uma modalidade, quando a sessão é finalizada, o decodificador 2900 é removido (por exemplo, desinstalado) do dispositivo de cliente 205.

Em uma modalidade, o aplicativo de transferência por download 2903 caracteriza o canal como o decodificador temporário 2900 que está sendo transferido por download, realizando avaliações do canal como a taxa de dados executável no canal (por exemplo, determinando quanto tempo leva para que dados sejam transferidos por download), a taxa de perda do pacote no canal e a latência do canal. O aplicativo de transferência por download 2903 gera dados de caracterização do canal que descrevem as avaliações do canal. Esses dados de caracterização do canal são, então, transmitidos do dispositivo de cliente 205 para o aplicativo de transferência por download 2901 do serviço, que usa os dados de caracterização do canal para determinar como utilizar o canal da melhor forma para transmitir meios para o dispositivo de cliente 205.

O dispositivo de cliente 205 tipicamente enviará de volta mensagens ao serviço de hospedeiro 205 durante a transferência por download do decodificador temporário 2900. Essas mensagens podem incluir mensagens de reconhecimento que indicam se os pacotes foram recebidos com ou sem erros. Em adição, as mensagens fornecem retorno para o aplicativo de transferência por download 2901 como para a taxa de dados (calculada com base na taxa nas quais pacotes são recebidos), a taxa de erros do pacote (com base na porcentagem de pacotes relatados recebidos com erros) e a latência de ida e volta do canal (com base na quantidade de tempo que leva antes de o aplicativo de transferência por download 2901 receber retorno sobre um dado pacote que foi transmitido).

A titulo de exemplo, se a taxa de dados for determinada como sendo de 2 Mbps, então, o aplicativo de transferência por download pode escolher uma resolução de janela de vídeo menor para o encodificador 2902 (por exemplo, 640x480 a 60 fps) do que se a taxa de dados fosse determinada como sendo de 5 Mbps (por exemplo, 1.280x720 a 60 fps). Diferentes correções antecipadas de erro (FEC) ou de estruturas de pacote podem ser escolhidas, dependendo da taxa de perda do pacote.

Se a perda de pacote for muito baixa, então, o áudio e vídeo comprimidos podem se transmitidos sem qualquer correção de erro. Se a

perda de pacote for mediana, então, o áudio e vídeo comprimidos podem ser transmitidos com técnicas que codificam correções de erro (por exemplo, como aquelas previamente descritas e ilustradas nas figuras 11a, 11b, 11c e 11d). Se a perda de pacote for muito alta, pode ser determinado que um fluxo audiovisual de qualidade adequada não pode ser transmitido e o dispositivo de cliente 205 ou pode notificar ao usuário que o serviço de hospedeiro não esta disponível através do canal de comunicação (isto é, "link"), ou pode tentar estabelecer uma rota diferente para o serviço de hospedeiro que tem uma perda de pacote inferior (conforme descrito abaixo).

Se a latência for baixa, então, o áudio e vídeo comprimidos podem ser transmitidos com abaixa latência e uma sessão pode ser estabelecida. Se a latência for muito alta (por exemplo, maior do que 80 ms) então, para jogos que necessitam de baixa latência, o dispositivo de cliente 205 pode notificar ao usuário que o serviço de hospedeiro não está disponível através do link, que um link esta disponível, mas o tempo de resposta para a entrada de usuário será lento ou "atrasado," ou que o usuário pode tentar estabelecer uma rota de diferença para o serviço de hospedeiro que tem uma latência inferior (conforme descrito abaixo).

O Dispositivo de cliente 205 pode tentar ser conectado ao Serviço do hospedeiro 210 através de outra rota através da rede (por exemplo, a Internet) para ver se defeitos são reduzidos (por exemplo, a perda de pacote é menor, a latência é menor, ou mesmo se a taxa de dados for maior). Por exemplo, o serviço do hospedeiro 210 pode ser conectado à Internet de múltiplos locais geograficamente (por exemplo, um centro hospedeiro em Los Angeles e um em Denver), e talvez haja uma grande perda de pacote devido à congestão em Los Angeles, mas não há congestão em Denver. Além disso, o serviço do hospedeiro 210 pode se conectar à Internet através de múltiplos provedores de serviço de internet (por exemplo, AT&T e Comcast).

Por causa da congestão ou outros problemas entre o dispositivo de cliente 205 e um dos provedores de serviço (por exemplo, AT&T), perda de pacote e/ou alta latência e/ou taxa de dados constrita podem resultar.

Entretanto, se o dispositivo de cliente 205 se conecta ao serviço do hospedeiro 210 através de outro provedor de serviço (por exemplo, Comcast), pode ser capaz de conectar sem problemas de congestão e/ou baixa perda de pacote e/ou baixa latência e/ou taxa de dados mais alta. Deste modo, se o dispositivo de cliente 205 experimentar uma perda de pacote acima de um limite especificado (por exemplo, um número específico de pacotes ignorados através de uma duração especificada), latência acima de um limite especificado e/ou uma taxa de dados abaixo de um limite especificado enquanto acontece o download do decodificador temporário 2900, em uma modalidade, este tenta reconectar ao serviço do hospedeiro 210 através de uma rota alternativa (tipicamente através de conectar a um endereço IP diferente ou um nome de domínio diferente) para determinar se uma conexão melhor pode ser obtida.

Se a conexão ainda estiver experimentando defeitos inaceitáveis após as opções de conexões alternativas serem esgotadas, pode ser porque a conexão local de dispositivo de cliente 205 à Internet está sofrendo falhas, ou porque fica muito longe do serviço do hospedeiro 210 para alcançar uma latência adequada. Em tal caso o dispositivo de cliente 205 pode notificar o usuário que o serviço do hospedeiro não está disponível através do link ou que é apenas disponível com defeitos, e/ou apenas certos tipos de jogos/aplicações baixa-latência são disponíveis.

Já que esta avaliação e melhora potencial das características de link entre o serviço do hospedeiro 210 e o dispositivo de cliente 205 ocorre enquanto o decodificador temporário está sendo transferido por download, este reduz a quantidade de tempo que o dispositivo de cliente 205 precisaria para gastar separadamente transferindo por download o decodificador temporário 2900 e avaliando as características de link. Apesar disto, em outra modalidade, a avaliação e melhora potencial das características de link são desempenhadas pelo dispositivo de cliente 205 separadamente da transferência por download do decodificador temporário 2900 (por exemplo, através do uso de dados de teste fictício em vez do código de programa de decodificador). Há inúmeras razões porque esta pode ser uma implantação

preferencial. Por exemplo, em algumas modalidades, o dispositivo de cliente 205 é implantado parcial ou inteiramente no hardware. Deste modo, para estas modalidades, não há um decodificador de software para este necessário para o download.·.

COMPRESSÃO QUE USA TAMANHOS DE LADRILHOS BASEADOS EM PADRÕES

Como mencionado acima, quando uma compressão baseada em ladrilho é usada, os princípios de base da invenção não são limitados a nenhuma orientação, formato ou tamanho de ladrilho particular. Por exemplo, em um sistema de compressão baseado em DCT como MPEG-2 e MPEG-4, os ladrilhos pode ter o tamanho de macroblocos (componentes usados em compressão de vídeo que representa tipicamente um bloco de 16 por 16 pixels). Esta modalidade fornece um nível muito bom de granulosidade para trabalhar com azulejos.

Além disso, independente do tamanho do ladrilho, diversos tipos de padrões de padrões de ladrilho podem ser usados. Por exemplo, a **figura 30** ilustra uma modalidade em que múltiplos I-ladrilhos são usados em cada quadro R 3001-3004. Um padrão de rotação é usado em que as I-telhas são dispersadas através de cada quadro R, de modo que um quadro inteiro I é gerado a cada quatro quadros R. A dispersão de I- ladrilhos desta maneira irá reduzir os efeitos de uma perda de pacote (limitando a perda de uma pequena região de exibição).

Os ladrilhos podem também ser conformados em uma estrutura nativa integral do algoritmo de compressão de base. Por exemplo, se o algoritmo de compressão H.264 for usado, em uma modalidade, os ladrilhos são determinados para serem do tamanho de "fatias"H.264. Isto permite que as técnicas descritas no presente documento sejam facilmente integradas no contexto de diversos algoritmos de compressão de padrões diferentes como H.264 e MPEG-4. Uma vez que o tamanho do ladrilho é determinado a uma estrutura de compressão nativa, as mesmas técnicas como aqueles descritas acima podem ser implantadas.

TÉCNICAS PARA RETROCESSO DE FLUXO E OPERAÇÕES DE REPRODUÇÃO

Como anteriormente descrito em conexão com a **figura 15**, o

fluxo de vídeo/áudio não comprimido1529 gerado por um servidor de aplicativo/jogo 1521-1525 pode ser comprimido por compressão compartilhada de hardware 1530 em múltiplas resoluções resultando simultaneamente em múltiplos fluxos de vídeo/áudio comprimidos 1539. Por exemplo, um fluxo de vídeo/áudio gerado por servidor de aplicativo/jogo 1521 pode ser comprimido em 1280x720x60fps pela compressão compartilhada de hardware 1530 e transmitido para um usuário por roteamento de saída 1540 como tráfego de Internet de saída 1599. Este mesmo fluxo de vídeo/áudio pode ser simultaneamente subescalonado ao tamanho de miniatura (por exemplo, 200x113) pela compressão compartilhada de hardware 1530 via caminho 1552 (ou através do buffer de atraso 1515) ao servidor de aplicativo/jogo 1522 para ser exibido como uma miniatura 1600 de uma coleção de miniaturas na **figura 16**. Quando a miniatura 1600 é ampliada através de tamanho intermediário 1700 na **figura 17** ao tamanho 1800 (1280x720x60fps) na **figura 18**, então em vez de descomprimir o fluxo de miniatura, servidor de aplicativo/jogo 1522 pode descomprimir uma cópia do fluxo de 1280x720x60fps sendo enviado ao usuário de servidor de aplicativo/jogo 1521, e escalonar o vídeo de maior resolução à medida que é ampliado a partir de tamanho de miniatura para o tamanho1280x720. Esta abordagem apresenta a vantagem de reutilizar o fluxo comprimido 1280x720 duas vezes. Mas há várias desvantagens: (a) o fluxo de vídeo comprimido enviado ao usuário pode variar em qualidade de imagem se o rendimento de dados da conexão de Internet do usuário varia resultando em uma qualidade imagem variante vista pelo usuário "espectador" de servidor de aplicativo/jogo 1522, mesmo se tal conexão de Internet do usuário não variar, (b) servidor de aplicativo/jogo 1522 terá que usar recursos de processamento para descomprimir toda a imagem 1280x720 e então escalonar tal imagem (e da mesma forma aplicar um filtro de reamostragem) para exibir tamanhos muito menores (por exemplo, 640x360) durante a ampliação, (c) se quadros são ignorado devido à largura de banda limitada de conexão de Internet e/ou pacotes perdidos/corrompidos, e o usuário espectador "retrocede" e "pausa" o vídeo gravado no buffer de atraso 1515, o usuário espectador descobrirá

que os quadros ignorados estão faltando no buffer de atraso (isto será particularmente aparente se o usuário "avançar" quadro a quadro), e (d) se o usuário espectador retrocede para encontrar um quadro particular no vídeo gravado no buffer de atraso, então o servidor de aplicativo/jogo 1522 terá que encontrar um I quadro ou I ladrilhos antes do quadro procurado no fluxo de vídeo gravado no buffer de atraso, e então descomprimir todos os P quadros/ladrilhos até que o quadro desejado seja alcançado. Estas mesmas limitações não seriam aplicáveis para usuários "espectadores" do fluxo de vídeo/áudio ao vivo, mas usuários (incluindo o usuário que gerou o fluxo de vídeo/áudio) assistindo a uma cópia arquivada (por exemplo, "Brag Clip") do fluxo de vídeo/áudio.

Uma modalidade alternativa da invenção aborda estes problemas ao comprimir o fluxo de vídeo em mais de um tamanho e/ou estrutura. Um fluxo (o fluxo "Ao Vivo") é comprimido da melhor forma para fluir até o usuário final, como descrito aqui, baseado nas características da conexão de rede (por exemplo, largura de banda de dados, confiabilidade de pacote) e as capacidades do cliente local do usuário (por exemplo, capacidade de descompressão, resolução de exibição). Outros fluxos (referidos aqui as fluxos "HQ") são comprimidos em alta qualidade, em uma ou mais resoluções, e em uma estrutura receptiva à reprodução de vídeo, e tais fluxos HQ são roteados e armazenados dentro do centro servidor 210. Por exemplo, em uma modalidade, os fluxos HQ comprimidos são armazenados em um arranjo de disco RAID 1515 e são utilizados para fornecer funções como pausar, retroceder, e outras funções de reprodução (por exemplo, "Brag Clips" que podem ser distribuídos para outros usuários para visualização).

Como ilustrado na **figura 31a**, uma modalidade da invenção compreende um encodificador 3100 capaz de comprimir um fluxo de vídeo em pelo menos dois formatos: um que periodicamente inclui I-Ladrilhos ou I-Quadros 3110 e um que não inclui I-Ladrilhos ou I-Quadros 3111, a menos que seja necessário devido um distúrbio do fluxo ou porque um I-Ladrilho ou I-Quadro é determinado para ser menor do que um I-Ladrilho ou I-Quadro (como descrito acima). Por exemplo, o fluxo "Ao Vivo" 3111 transmitido ao

usuário enquanto joga um videogame pode ser comprimido ao utilizar apenas P-Quadros (a menos que I-Ladrilhos ou I-Quadros sejam necessários ou menores como descrito acima). Em adição, o encodificador 3100 desta modalidade comprime ao mesmo tempo fluxo de vídeo Ao Vivo 3111 em um segundo formato que, em uma modalidade, inclui periodicamente I-Ladrilhos ou I-Quadros (ou tipo similar de formato de imagem).

Enquanto as modalidades descritas acima empregam I-Ladrilhos, I-Quadros, P-Ladrilhos e P-Quadros, os princípios subjacentes da invenção não estão limitados a qualquer algoritmo de compressão em particular. Por exemplo, qualquer tipo de formato de imagem no qual quadros são dependentes de quadros anteriores ou subsequentes pode ser utilizado no lugar de P-Ladrilhos ou P-Quadros. De forma similar, qualquer tipo de formato de imagem que não é dependente de quadros anteriores ou subsequentes pode ser substituído em favor dos I-Ladrilhos ou I-Quadros descritos acima.

Conforme mencionado acima, o Fluxo HQ 3110 inclui I-Quadros periódicos (por exemplo, em uma modalidade, a cada 12 quadros aproximadamente). Isto é significante devido ao fato de que se o usuário quiser rebobinar rapidamente o fluxo de vídeo armazenado para um ponto particular, Ladrilhos ou I-Quadros são necessários. Com um fluxo comprimido de somente P-Quadro (isto é, sem o primeiro quadro da sequência sendo um I-Quadro), seria necessário que o decodificador voltasse ao primeiro quadro da sequência (que pode estar a horas de distância) e descomprime P-quadros até o ponto ao qual o usuário quer rebobinar. Com um I-Quadro a cada 12 quadros armazenados no Fluxo HQ 3110, o usuário pode decidir rebobinar até um ponto particular spot e o I-Quadro precedente mais próximo do fluxo HQ não está a mais que 12 quadros antes do quadro desejado. Mesmo se a taxa de decodificação máxima do decodificador for em tempo real (por exemplo, 1/60° de um segundo para um fluxo de 60 quadro/seg), então 12 (quadros)/60 (quadros/seg) = 1/5 segundo afastado de um I-Quadro. E, em muitos casos, os decodificadores podem operar muito mais rápido que em tempo real, por exemplo, em 2x o tempo real, um decodificador poderia

decodificar 12 quadros em 6 quadros, que é apenas 1/10° de um segundo atraso para um "rebobinar". Não é necessário dizer que mesmo um decodificador rápido (por exemplo, 10x o tempo real) poderia ter um atraso inaceitável se o I-Quadro precedente mais próximo fosse um número grande de quadros anteriores a um ponto de rebobinagem (por exemplo, poderia tomar 1 hora/10=6 minutos para realizar a "rebobinagem"). Em outra modalidade, I-Ladrilhos periódicos são usados, e neste caso quando o usuário procura rebobinar o decodificador encontrará o I-Ladrilho precedente mais próximo antes do ponto de rebobinagem, e então começar a decodificação daquele ladrilho a partir daquele ponto até que todos os ladrilhos sejam decodificados até o ponto de rebobinagem. Embora I-Ladrilhos ou I-Quadros periódicos resultem em compressão de menos eficiência que eliminar I-Quadros completamente, o serviço hospedeiro 210 tipicamente tem largura de banda localmente disponível mais que suficiente e capacidade de armazenamento para gerenciar o fluxo HQ.

Em outra modalidade, o decodificador 3100 encodifica o fluxo HQ com I-Ladrilho ou I-Quadros periódicos, seguidos pelos P-Ladrilhos ou P-Quadro, conforme previamente descrito, mas também precedidos pelos B-Ladrilhos ou B-Quadros. Os B-Quadros, conforme descritos previamente, são quadros que precedem um I-Quadro e são baseados em diferenças de quadro do I-Quadro que trabalha para trás no tempo. Os B-Ladrilhos são a contraparte do ladrilho, que precede um I-Ladrilho e com base em diferenças de quadro que trabalha para trás a partir do I-Ladrilho. Nesta modalidade, se o ponto de rebobinagem desejado for um B-Quadro (ou contiver B-Ladrilhos), então o decodificador encontrará o I-Quadro ou I-Ladrilho *seguinte* mais próximo e decodificará para trás no tempo até o ponto de rebobinagem desejado ser decodificado, e então como a reprodução de vídeo prossegue a partir daquele pondo para frente, o decodificador decodificará B-Quadros, I-Quadros e P-Quadro (ou suas contrapartes de ladrilho) em quadros sucessivos que vão para frente. Uma vantagem de empregar B-Quadros ou B-Ladrilhos em adição a tipos I e P é que, normalmente uma qualidade mais alta em uma dada razão de compressão pode ser atingida.

Já em outra modalidade, o encodificador 3100 encodifica o fluxo HQ como todos os I-Quadros. Uma vantagem desta abordagem é que o ponto de rebobinagem é um I-Quadro, e como resultado, nenhum outro quadro precisa ser decodificado a fim de alcançar o ponto de rebobinagem. Uma desvantagem é a taxa de dados comprimidos será muito alta comparada a encodificação de fluxo I, P ou I, P, B.

Outras ações de reprodução de fluxo de vídeo (por exemplo, rebobinagem rápida ou lenta, avanço rápido ou lento, etc.), tipicamente são muito mais praticamente realizadas com I-Quadros ou I-Ladrilhos periódicos (sozinhos ou combinados com contrapartes P e/ou B), já que em cada caso o fluxo é reproduzido em um diferente quadro a fim de avançar quadro por quadro no tempo, e como resultado, o decodificador precisa encontrar e decodificar um quadro particular, frequentemente arbitrário, na sequência. Por exemplo, no caso de avanço muito rápido (por exemplo, velocidade de 100x), cada quadro sucessivo exibido é de 100 quadros após o quadro anterior. Mesmo com um decodificador que roda a 10x o tempo real e decodifica 10 quadros em tempo de 1 quadro, ainda seria 10x mais lendo para atingir avanço rápido de 100x. Entretanto, com I-Quadros ou I-Ladrilhos periódicos conforme descritos acima, o decodificador é capaz de procurar o I-Quadro ou I-Ladrilhos aplicáveis mais próximos ao quadro que precisa ser exibido em seguida e semente decodifica os quadros ou ladrilhos interveniente ao ponto do quadro alvo.

Em outra modalidade as I-Quadros são codificadas no fluxo HQ em uma periodicidade consistente (por exemplo, sempre a cada 8 quadros) e os multiplicadores de velocidade colocados à disposição do usuário para o avanço rápido e retrocesso que são mais rápidos que a taxa de I-Quadro são múltiplos exatos da periodicidade de I-Quadro. Por exemplo, se a periodicidade de I-Quadro é 8 quadros, então as velocidades de avanço rápido e retrocesso disponibilizadas ao usuário podem ser 1X, 2X, 3X, 4X, 8X, 16X, 64X e 128X e 256X. Para as velocidades mais rápidas que a periodicidade de I-Quadro, o decodificador irá pular para frente até I-Quadro mais próximo que é número de quadros à frente na velocidade (por exemplo, se o quadro

exibido atualmente for 3 quadros antes de um I-Quadro, então em 128X, o decodificador irá pular para um quadro 128+3 quadros à frente) e, então, para cada quadro sucessivo o decodificador irá pular o número exato de quadros conforme a velocidade escolhida (por exemplo, na velocidade escolhida de 128X, o decodificador pularia 128 quadros) que estabeleceria exatamente em um I-Quadro cada vez. Dessa forma, dado que todas as velocidades mais rápidas que a periodicidade de I-Quadro são múltiplos exatos da periodicidade de I-Quadro, o decodificador nuca precisará decodificar quaisquer quadros precedentes ou seguintes para buscar o quadro desejado e terá somente que decodificar um I-Quadro por quadro exibido. Para as velocidades mais lentas que a periodicidade de I-Quadro (por exemplo, 1X, 2X, 3X, 4X), ou para velocidades mais rápidas que não são múltiplas da periodicidade de I-Quadro, para cada quadro exibido, o decodificador busca sejam quais forem os quadros, exigem pelo menos quadros recentemente decodificados adicionais para exibir o quadro desejado, seja um I-Quadro não decodificado ou um quadro já decodificado ainda disponível na forma decodificada (em RAM ou outro armazenamento rápido) e então, decodificam quadros intervenientes, conforme necessário, até o quadro desejador ser decodificado e exibido. Por exemplo, em 4X avanço rápido, em uma sequência codificada I,P com 8X periodicidade de I-Quadro, se o quadro atual for um P-quadro que é 1 quadro que segue um I-quadro, estão o quadro desejado a ser exibido está 4 quadros adiante, que seria o 5° P-Quadro que segue o I-quadro precedente. Se o quadro exibido atualmente (que acabara de ser decodificado) for usado como um ponto de partida, o decodificador precisará decodificar mais 4 P-quadros para exibir o quadro desejado. Se o I-Quadro precedente for usado, o decodificador precisará decodificar 6 quadros (o I-Quadro e os 5 P-Quadros seguintes) a fim de exibir o quadro desejado. (Caramente, neste caso, é vantajoso usar o quadro exibido atualmente para minimizar os quadros adicionais a decodificar). Então, o próximo quadro a ser decodificado, 4 quadros a frente, seria o 1° Quadro que segue um I-Quadro. Neste caso, se o quadro decodificado atualmente foi usado como um ponto de partida, o decodificador precisaria

decodificar mais 4 quadros (2 P-Quadros, um I-Quadro e um P-Quadro). Mas, se o próximo I-Quadro foi usado no lugar, o decodificador precisaria somente decodificar o I-Quadro e o P-Quadro seguinte. (Claramente, neste caso, é vantajoso usar o próximo I-quadro como um ponto de partida para minimizar os quadros adicionais a decodificar). Dessa forma, neste exemplo, o decodificador alternaria entre usar o quadro decodificado atualmente como um ponto de partida e usar um I-Quadro subsequente como um ponto de partida. Como um princípio geral, independente do modo de reprodução de fluxo de vídeo HQ (avanço rápido, retrocesso ou etapa) e velocidade, o decodificador começaria com qualquer quadro, seja um I-Quadro ou um quadro decodificado anteriormente, exige pelo menos o número de quadros decodificados recentemente para exibir o quadro desejado para cada quadro sucessivo exibido para esse modo de reprodução e velocidade.

Conforme ilustrado na **figura 31b**, uma modalidade do serviço de hospedeiro 210 inclui a lógica de repetição de fluxo 3112 para administrar requerimentos do usuário para repetir o fluxo HQ 3110. A lógica de repetição de fluxo 3112 recebe os requerimentos do cliente que contêm comandos de reprodução de vídeo (por exemplo, pausa, retrocesso, reprodução de um ponto especificado, etc), interpreta os comandos e decodifica o fluxo HQ 3110 do ponto especificado (ou começando com ou um I-Quadro ou um quadro decodificado anteriormente, conforme apropriado e, então, procedendo para frente ou para trás ao ponto especificado). Em uma modalidade, um fluxo HQ decodificado é fornecido para um encodificador 3100 (potencialmente o encodificador próprio 3100, se capaz de codificar mais que um fluxo ao mesmo tempo ou um encodificador separado 3100) de modo que possa ser recomprimido (usando as técnicas descritas na presente invenção) e transmitido ao dispositivo de cliente 205. O decodificador 3102 no dispositivo de cliente, então, decodifica e passa o fluxo conforme descrito acima.

Em uma modalidade, a lógica de repetição de fluxo 3112 não decodifica o fluxo HQ e, então, faz com que o encodificador 3100 recodifique o fluxo. Particularmente, ele simplesmente transmite o fluxo HQ 3110

diretamente para o dispositivo de cliente 205 a partir do ponto especificado. O decodificador 3102 no dispositivo de cliente 205, então, decodifica o fluxo HQ. Como as funções de reprodução descritas na presente invenção não têm tipicamente os mesmos requerimentos de baixa latência que reproduzir um videogame em tempo real (por exemplo, se o jogador está simplesmente revendo o jogo anterior e não jogando ativamente), a latência adicionada tipicamente inerente no fluxo HQ usualmente de qualidade mais alta pode resultar em uma experiência de usuário final aceitável (por exemplo, com latência mais alta, mas vídeo de qualidade mais alta).

A título de exemplo, e não limitação, se o usuário está jogando videogame, o encodificador 3100 está fornecendo um fluxo Ao vivo de essencialmente todos os P-quadros otimizados para a conxão do usuário e cliente local (por exemplo, aproximadamente 1,4 Mbps em uma resolução de 640 x 360). Ao mesmo tempo, o encodificador 3100 também comprime o fluxo de vídeo como um fluxo HQ 3110 dentro do serviço de hospedeiro 310 e armazena o fluxo HQ em um arranjo de Decodificador de Vídeo Digital (RAID) local em, por exemplo, 1280 x 720 a 10 Mbps com I-quadros cada 12 quadros. Se o usuário aperta um botão "Pause", o jogo então será pausado no último quadro decodificado do cliente e a tela congelará. Depois, se o usuário aperta um botão "Rewind", a lógica de repetição de fluxo 3112 irá ler o fluxo HQ 3110 a partir do RAID do DVR a começar do I-quadro mais próximo ou quadro já decodificado disponível, como descrito acima. A lógica de repetição de fluxo 3112 descomprimirá os quadros P ou B que intervêm, como necessário, sequenciar novamente os quadros como necessário de forma que a sequência de reprodução é inverso na velocidade de rebobinar desejada, e então redimensiona (com o uso de técnicas de escalonamento de imagem da técnica anterior bem conhecidas na técnica) o decodificado desejado que deveria ser exibido de 1280 x 720 a 640 x 360, e o encodificador de fluxo Ao vivo 3100 comprimirão novamente o fluxo sequenciado novamente na resolução 640 x360 e transmitir o mesmo ao usuário. Se o usuário pausar novamente, e depois etapas individuais através do vídeo para assistir uma sequência de perto, o fluxo HQ 3110 sobre o RAID do DVR

terá cada quadro disponível para realizar etapas individuais (ainda que o fluxo Ao vivo original possa ter quadros ignorados por qualquer uma dentre as muitas rasões descritas no presente documento). Além disso, a qualidade da reprodução do vídeo será bem alta em cada ponto no fluxo HQ, enquanto que podem existir pontos no fluxo Ao vivo onde, por exemplo, a largura de banda foi comprometida, o que resulta em uma redução temporária na qualidade da imagem comprimida. Embora a qualidade da imagem comprometida por um breve período de tempo, ou em uma imagem em movimento, possa ser aceitável para o usuário, se o usuário para em um quadro em particular (ou lentamente em etapas individuais) e studa os quadros de perto, a qualidade comprometida pode não ser aceitável. É também fornecida ao usuário a habilidade de avanço rápido, ou pular para um ponto em particular, mediante especificação de um ponto dentro do fluxo HQ (por exemplo, 2 minutos antes). Todas estas operações seriam impraticáveis em sua generalidade plena e em alta qualidade com um fluxo de vídeo Ao vivo que foi P-quadro-somente ou raramente (ou imprevisivelmente) teve I-Quadros.

Em uma modalidade, é fornecido ao usuário uma janela de vídeo (não mostrada) tal como uma janela de vídeo Apple QuickTime ou Adobe Flash com um "purificador" (ou seja, um controle deslizante esquerda-direita) que permite que o usuário faça uma varredura para frente e para trás através do fluxo de vídeo, até onde o fluxo HQ armazenou o vídeo. Embora pareça ao usuário como se ele ou ela estivesse "purificando" através do fluxo Ao vivo, na realidade, ele ou ela purifica o fluxo armazenado HQ 3110, o qual é depois redimensionado e recomprimido como um fluxo Ao vivo. Em adição, como mencionado previamente, se o fluxo HQ por qualquer outra pessoa ao mesmo tempo, ou pelo usuário em um momento diferente, pode ser assistido uma resolução maior (ou menor) do que a resolução do fluxo Ao vivo enquanto o fluxo HQ encodificado simultaneamente, e a qualidade tão alta quanto a qualidade do fluxo Ao vivo do *espectador*, potencialmente até a qualidade do fluxo HQ.

Assim, mediante a encodificação simultânea tanto do fluxo Ao vivo (como descrito no presente documento de uma maneira apropriada por

suas exigências de baixa latência, largura de banda e tolerância a erros de pacote) e um fluxo HQ com suas exigências de alta qualidade, ação de reprodução de fluxo, é fornecido, desse modo, ao usuário uma configuração desejada para ambos os cenários. E, na realidade, é efetivamente transparente ao usuário que existem dois fluxos diferentes encodificados de modo diferente. Da perspectiva do usuário, a experiência é altamente responsiva com baixa latência, apesar de executar sobre uma conexão de Internet com uma largura de banda relativamente baixa e altamente variável, apesar de a funcionalidade do Gravador de Vídeo Digital (DVR) é qualidade muito alta, com ações flexíveis e velocidades flexíveis.

Como um resultado das técnicas descritas acima, o usuário recebe os benefícios tanto de fluxo de vídeo Ao vivo e HQ durante jogo online ou outra interação online, sem sofrer de nenhuma das limitações ou de um fluxo Ao vivo ou de um fluxo HQ.

A figura 31c ilustra uma modalidade de uma arquitetura de sistema para realizer as operações acima. Como ilustrado, nesta modalidade, o encodificador 3100 encodifica uma série de fluxos "Ao vivo" 3121L, 3122L, e 3125L e uma série correspondente de fluxos "HQ" 3121H1-H3, 3122H1-H3, e 3125H1-H3, respectivamente. Cada fluxo HQ H1 é encodificado em resolução completa, enquanto cada encodificador H2, e H3 escalona ao fluxo de vídeo para um tamanho menor antes da encodificação. Por exemplo, se o fluxo de vídeo fosse de resolução 1280x720, H1 encodificaria em resolução 1280x720, enquanto H2 poderia escalonar a 640x360 e encodificar naquela resolução e H3 poderia escalonar a 320x180 e encodificar naquela resolução. Qualquer número de escalonador/encodificadores H*n* simultâneos poderiam ser utilizados, que fornecem endificações de HQ simultâneas múltiplas em uma variedade de resoluções.

Cada um dos fluxos de vídeo Ao vivo opera em resposta a sinais de retorno de canais 3161, 3162 e 3165 recebidos através de uma conexão de internet de entrada 3101, conforme descrito acima (consultar, por exemplo, a discussão dos sinais de retorno 2501 e 2601 nas figuras 25 e 26). Os fluxos de vídeo Ao vivo são transmitidos pela Internet (ou outra rede) através

de uma lógica de roteamento de saída 3140. Os compressores ao vivo 3121L a 3125L incluem lógica para adaptar os fluxos de vídeo comprimidos (incluindo quadros escalonados, ignorados etc.) com base em retorno de canal.

Os fluxos de HQ são roteados por lógica de roteamento de entrada 3141 e 1502 para armazenamentos temporários de atraso interno (por exemplo, série RAID 3115) ou outros dispositivos de armazenamento de dados através do caminho de sinal 3151 e/ou são fornecidos como retorno através do caminho de sinal 3152 em servidores de aplicativo/jogo e no codificador 3100 para processamento adicional. Conforme descrito acima, os fluxos de HQ 3121H*n* a 3125H*n* são transmitidos para usuários finais por solicitação (consultar, por exemplo, s figura 31b e o texto associado).

Em uma modalidade, o encodificador 3100 é implantado com a lógica de compressão compartilhada de hardware 1530, mostrado na figura 15. Em outra modalidade, algum ou todos os encodificadores e escalonadores são subsistemas individuais. Os princípios subjacentes da invenção não estão limitados a qualquer compartilhamento particular de recursos de escalonamento ou compressão e/ou configuração de hardware/software.

Uma vantagem da configuração da figura 31c é que os Servidores de Aplicativo/Jogo 3121 a 3125 que exigem janelas de vídeo de tamanho completo menores não precisarão processar e descomprimir uma janela de tamanho completo. Além disso, os Servidores de Aplicativo/Jogo 3121 a 3125 que exigem tamanhos de janelas intermediários podem receber um fluxo comprimido que é próximo do tamanho de janela desejado e, então, escalonar para cima ou para baixo para o tamanho de janela desejado. Além disso, se múltiplos Servidores de Aplicativo/Jogo 3121 a 3125 solicitarem o mesmo tamanho de fluxo de vídeo a partir de outro Servidor de Aplicativo/Jogo 3121 a 3125, o Roteador de Entrada 3141 pode implantar técnicas de multicast de IP, como as conhecidas na técnica, e difundir o fluxo solicitado para múltiplos Servidores de Aplicativo/Jogo 3121 a 3125 de uma vez, sem solicitar um fluxo independente para cada Servidor de Aplicativo/Jogo que faça uma solicitação. Se algum servidor de Aplicativo/Jogo que

recebe uma difusão alterar o tamanho de uma janela de vídeo, ele pode comutar para a difusão de um tamanho de vídeo diferente. Assim, um número arbitrariamente grande de usuários pode assistir a um fluxo de vídeo de Servidor de Aplicativo/Jogo, cada um com a flexibilidade de escalonar suas janelas de vídeo e sempre ter o benefício de um fluxo de vídeo escalonado próximo do tamanho de janela desejada.

Uma desvantagem com a abordagem mostrada na figura 31c é que em muitas implantações práticas do Serviço de Hospedeiro 210, nunca há um momento em que todos os fluxos de HQ comprimidos, muito menos todos os tamanhos de todos os fluxos de HQ comprimidos, são assistidos de uma vez. Quando o encodificador 3100 é implantado como um recurso compartilhado (por exemplo, um escalonador/compressor, cada um implantado em software ou hardware), este desperdício é mitigado. Mas podem haver questões práticas em conectar um grande número de fluxos descomprimidos para um recurso compartilhado comum, devido à largura de banda envolvida. Por exemplo, cada fluxo 1080p60 é quase 3 Gbps, que é muito além até mesmo de Gigabit Ethernet. As modalidades alternativas a seguir confrontam esta questão.

A figura 31d mostra uma modalidade alternativa do Serviço de Hospedeiro 210 em que cada Servidor de Aplicativo/Jogo 3121 a 3125 tem dois compressores alocados para ele: (1) um compressor de fluxo Ao vivo 3121L a 3125L, que adapta o fluxo de vídeo comprimido com base no Retorno de Canal 3161 a 3165, e (2) um compressor de fluxo de HQ que envia um fluxo de HQ de resolução completa, conforme descrito acima. Notavelmente, o compressor Ao vivo é dinâmico e adaptável, utilizando comunicações de duas vias com o cliente 205, enquanto o fluxo de HQ é não adaptável e de uma via. Outra diferença entre o fluxo é que a qualidade do fluxo Ao vivo pode variar drasticamente, dependendo das condições do canal e da natureza do material de vídeo. Alguns quadros podem ser de baixa qualidade, e podem haver quadros ignorados. Além disso, o fluxo Ao vivo pode ser quase inteiramente de P-quadros ou P-ladrilhos, com I-quadros ou I-ladrilhos aparecendo com pouca frequência. O fluxo de HQ

tipicamente será uma taxa de dados muito maior do que o Fluxo Ao vivo, e fornecerá alta qualidade consistente, sem ignorar qualquer quadro.O fluxo de HQ pode ser todo de I-quadros, ou pode ter I-quadros ou I-ladrilhos frequentes e/ou regulares. O fluxo de HQ pode também incluir B-quadros ou B-ladrilhos.

Em uma modalidade, o escalonamento e a recompressão de vídeo Compartilhado 3142 (detalhado abaixo) seleciona apenas determinados fluxos de vídeo de HQ 3121H1 a 3125H1 para serem escalonados e recomprimidos em uma ou mais resoluções diferentes, antes de enviar para o Roteamento de Entrada 3141 para rotear conforme descrito anteriormente. Cada um dos outros fluxos de vídeo de HQ são atravessados em seu tamanho completo para o Roteamento de Entrada 3141 para rotear, conforme descrito anteriormente, ou não são atravessados de qualquer modo. Em uma modalidade, a decisão em que fluxos de HQ são escalonados e recomprimidos e/ou em que fluxos de HQ são atravessados de qualquer modo é determinada com base em se há um Servidor de Aplicativo/Jogo 3121 a 3125 que está solicitando fluxo de HQ particular na resolução particular (ou uma resolução próximo da resolução escalonada ou completa). Através deste meio, os únicos fluxos de HQ que são escalonados e recomprimidos (ou potencialmente atravessados de qualquer modo) são fluxos de HQ que são realmente necessários. Em muitas aplicações do Servidor de Hospedeiro 210, isto resulta em uma redução drástica de recursos de escalonamento e compressão. Além disso, dado que todo fluxo de HQ é ao menos comprimido em sua resolução completa pelos compressores 3121H1 a 3125H1, a largura de banda necessária para ser roteado para e dentro de escalonamento e recompressão de vídeo Compartilhado 3142 é reduzida drasticamente como se fosse aceito vídeo descomprimido. Por exemplo, um fluxo 1080p60 descomprimido de 3GBps poderia ser comprimido para 10Mbps e ainda retém qualidade muito alta. Assim, com conectividade Gigabit Ethernet, em vez de ser incapaz de portar até mesmo um fluxo de vídeo de 3Gbps descomprimido, seria possível portar dezenas de fluxos de vídeo de 10Mbps, com redução pouco aparente de qualidade.

figura 31f mostra detalhes do escalonamento e recompressão do Vídeo compartilhado 3142, junto com um grande número de compressores de vídeo de HQ, HQ 3121H1-3131H1. O roteamento interno 3192 através de solicitações por fluxos de vídeo particulares escalonados para tamanhos particulares dos servidores de Aplicativo/Jogo 3121-3125 seleciona geralmente um subconjunto de fluxos de HQ comprimidos por compressores de vídeo de HQ, HQ 3121H1-3131H1. Um fluxo dentro desse subconjunto selecionado de fluxos é roteado tanto através de um Descompressor 3161-3164 se o fluxo solicitado for escalonado, ou roteado no Caminho de vídeo não escalonado 3196 se o fluxo solicitado estiver em resolução completa. Os fluxos a serem escalonados são descomprimidos a vídeos descomprimidos pelos Descompressores 3161-3164, depois, cada um é escalonado ao tamanho solicitado pelos Escalonadores 3171-3174, depois, cada vídeo comprimido pelos Compressores 3181-3184. Observe que se um fluxo de HQ particular é solicitado em mais de uma resolução, depois o Roteamento Interno 3192 faz o multicast do fluxo (usando a tecnologia de multicasting de IP que é bastante conhecida pelos versados na técnica) para um ou mais Descompressores 3161-3164 e (se um tamanho solicitado se completa resolução) ao Roteamento de Saída 3193. Todos os fluxos solicitados, tanto escalonados (a partir dos Compressores 3181-3184) ou não (a partir do Roteamento Interno 3192), são então enviados ao Roteamento de Saída 3193. O Roteamento 3193 então envia cada fluxo solicitado ao servidor de Aplicativo/Jogo 3121-3125 que fez a solicitação. Em uma modalidade, se mais de um servidor de Aplicativo/Jogo solicitar o mesmo fluxo na mesma resolução, então o Roteamento Externo 3193 faz o multicast do fluxo para todos os servidores de Aplicativo/Jogo 3121-3125 que estão fazendo a solicitação.

Na modalidade preferencial presente do escalonamento e compressão de Vídeo compartilhado 3142, o roteamento é implantado pelo uso de comutadores Gigabit Ethernet, e a descompressão, escalonamento, e compressão é implantada por dispositivos semicondutores especializados distintos que implantam cada função. A mesma funcionalidade poderia ser

implantada com um maior nível de integração no hardware ou por processadores muito rápidos.

figura 31e mostra outra modalidade de Serviço de Hospedeiro 210, em que a função de Armazenamento temporário de atraso 3115, descrito previamente, é implantada em um armazenamento temporário de atraso de Vídeo compartilhado, subsistema de escalonamento e descompressão 3143. Os detalhes do subsistema 3143 são mostrados na **figura 31g**. A operação do subsistema 3143 é similar a do subsistema 3142 mostrada na **figura 31f**, exceto que 3191 seleciona primeiro quais fluxos de vídeo de HQ serão roteados, através de solicitações dos servidores de Aplicativos/Jogos 3121-3125, e depois, os fluxos de HQ que são solicitados a serem atrasados são roteados através do Armazenamento temporário de atraso 3194, implantado como arranjo de RAID na modalidade preferencial presente (mas poderia ser implantada em qualquer meio de armazenamento de largura de banda e capacidade), e fluxos que não são solicitados para serem atrasados são roteados através do Caminho de vídeo não atrasado 3195. A saída do Armazenamento temporário de atraso 3194 e Vídeo não atrasado 3195 são então roteadas pelo Roteamento Interno 3192 com base no fato de fluxos solicitados serem escalonados ou não. Fluxos escalonados são roteados através dos Descompressores 3161-3164, Escalonadores 3171-3174 e Compressores 3181-3184 ao Roteamento de saída 3193, e o Vídeo não escalonado 3196 é também enviado ao Roteamento de saída 3193, e depois o Roteamento de saída 3193 envia o vídeo em modo unicast ou multicast aos servidores de Aplicativo/Jogo da mesma maneira que descrito anteriormente no subsistema 3142 da **figura 31f**.

Outra modalidade de armazenamento temporário de vídeo, subsistema de escalonamento e descompressão é mostrado na **figura 31h**. Nessa modalidade um Armazenamento temporário de atraso individual de HQ 3121D-HQ 3131D é fornecido para cada fluxo de HQ. Por causa do rápido declínio de custo de RAM e Flash ROM, os quais podem ser usados para atrasar um fluxo de vídeo comprimido, isso pode acabar sendo menos dispendioso e /ou mais flexível de que ter um Armazenamento Temporário

de atraso 3194. Ou, em ainda outar modalidade, um único Armazenamento temporário de atraso 3197 (mostrado em linha pontilhada) pode fornecer atraso para todos os fluxos de HQ individualmente em recurso coletivo de alto desempenho (por exemplo, disco, Flash ou RAM muito rápidos). Em qualquer caso, cada Armazenamento temporário de atraso de HQ 3121D-3131D é capaz de atrasar variavelmente um fluxo da fonte de vídeo de HQ, ou passar o fluxo não atrasado. Em outra modalidade, cada armazenamento temporário de atraso é capaz de fornecer fluxos múltiplos com diferentes quantidades de atraso. Todos os atrasos ou não atrasos são solicitados por Serviços de Aplicativo/Jogo 3121-3125. Em todos esses casos fluxos de vídeo atrasados e não atrasados 3198 são enviados ao Roteamento Interno 3192, e procedem através do resto do subsistema 3143 como descrito previamente em relação às **figuras 31g**.

Nas modalidades precedentes relativas as várias **figuras 31*n*** observa-se que o Fluxo ao vivo utiliza uma conexão de duas vias a é adaptado para um usuário particular, com mínima latência. Os fluxos de HQ utilizam conexões de uma via e são tanto unicast quanto multicast. Observe que enquanto a função multicast é ilustrada nessas figuras como uma única unidade, como poderia ser implantada em um comutador Gigabit Ethernet, em um sistema de larga escala, a função multicast seria implantada através de uma árvore de comutadores múltiplos. De fato, no caso de um fluxo de vídeo de um jogador de vídeo game de alta classificação, pode ser que o fluxo de HQ do jogador seja assistido por milhões de usuários simultaneamente. Nesse caso, provavelmente haveria um grande número de comutadores individuais em estágios sucessivos difundindo o fluxo de HQ que sofreu multicast.

Para ambos os propósitos de diagnóstico, e para fornecer realimentação para o usuário (por exemplo, informar ao usuário sobre a popularidade de seu desempenho de jogo), em uma modalidade, o serviço de hospedagem 210 rastrearia a quantidade de espectadores simultâneos que há de cada fluxo de vídeo de Servidor de Aplicação/Jogo 3121 a 3125. Isso pode ser realizado mantendo-se uma contagem de execução do

número de solicitações ativas por Aplicação/Servidores de jogo para um fluxo de vídeo particular. Então, um jogador que tem 100.000 espectadores simultâneos saberá que seu jogo é muito popular, e isso criará um incentivo para que jogadores tenham um melhor desempenho e atraiam espectadores. Quando há um grande número de espectadores de fluxos de vídeo (por exemplo, de uma partida de videogame de um campeonato), pode-se desejar que os comentaristas falem durante a partida de videogame de tal modo que alguns ou todos os usuários assistindo ao multicast possam ouvir seus comentários.

As Aplicações e Jogos sendo executados em Aplicação/Servidores de jogo serão fornecidos com uma Interface de Programa de Aplicação (API) em que a Aplicação e/ou Jogo possa submeter solicitações para fluxos de vídeo particulares com características particulares (por exemplo, resolução e quantidade de atraso). Ademais, esses APIs, submetidos a um ambiente de operação sendo executado em Aplicação/Servidor de jogo, ou a um Sistema de Controle de Serviço de Hospedagem 401 da **figura 4a** pode rejeitar tais solicitações por uma variedade de motivos. Por exemplo, o fluxo de vídeo solicitado pode ter certas restrições de direitos de licença (por exemplo, de modo que possa ser visualizado somente por um único espectador, e não difundido para outros), pode haver restrições de assinatura (por exemplo, o espectador pode ter que pagar pelo direito de visualizar o fluxo), pode haver restrições de idade (por exemplo, o espectador pode ter que ter 18 anos para visualizar o fluxo), pode haver restrições de privacidade (por exemplo, a pessoa que usa a Aplicação ou que está jogando o jogo pode limitar a visualização a somente um número ou classe selecionada de espectadores (por exemplo, seus "amigos"), ou pode não permitir a visualização completamente), e pode haver restrições que exigem que o material seja atrasado (por exemplo, se o usuário está jogando um jogo de camuflagem em que sua posição pode ser revelada). Há inúmeras outras restrições que limitariam a visualização do fluxo. Em qualquer um desses casos, a solicitação por Aplicação/Servidor de jogo seria rejeitada com um motivo para a rejeição e, em uma modalidade, com alternativas através das quais a

solicitação seria aceita (por exemplo, estabelecer a taxa que seve ser paga para uma assinatura).

Os fluxos de vídeo de HQ que estão armazenados em Armazenamentos Temporários de Atraso, em qualquer uma das modalidades precedentes, podem ser exportados para outros destinos fora do Serviço de Hospedagem 210. Por exemplo, um fluxo de vídeo particularmente interessante pode ser solicitado por uma Aplicação/Servidor de jogo (tipicamente, através da solicitação de um usuário) para ser exportado para o YouTube. Em tal caso, o fluxo de vídeo seria transmitido através da Internet em um formato consentido pelo YouTube, junto com informações descritivas adequadas (por exemplo, o nome do usuário jogador, o jogo, o tempo, a pontuação, etc.). Isso poderia ser implantado através do multicast, em um fluxo separado, do áudio de comentários para todos os Servidores de Jogo/Aplicação 3121 a 3125 solicitando tais comentários. Os Servidores de Jogo/Aplicação fundiriam o áudio do comentário, com o uso de técnicas de mixagem de áudio bem conhecidas pelos versados na técnica, ao fluxo de áudio enviado às premissas de usuário 211. Poderia haver múltiplos comentaristas (por exemplo, com diferentes pontos de vista, ou em diferentes idiomas), e os usuários poderiam selecionar entre esses.

De modo similar, os fluxos de áudio separados poderiam ser mixados em ou funcionar como substituição para trilha de áudio de fluxos de vídeo particulares (ou fluxos individuais) no Serviço de Hospedagem 210, mixando ou substituindo o áudio do fluxo contínuo de vídeo em tempo real ou de um Armazenamento Temporário de Atraso. Tal áudio pode ser um comentário ou narração, ou poderia fornecer vozes para personagens no fluxo de vídeo. Isso permitira que Machinima (animações de geração por usuário de fluxos de vídeo de videogame) fosse prontamente criado por usuários.

Os fluxos de vídeo descritos ao longo desse documento são mostrados como capturados a partir da saída de vídeo de Servidores de Aplicação/Jogo, e, então, submetido a fluxo e/ou atraso e reutilizado ou distribuído em uma variedade de maneiras. Os mesmos Armazenamentos

Temporários de Atraso podem ser usados para manter o material de vídeo proveniente das fontes diferentes do Servidor de Aplicação/Jogo e fornecem o mesmo grau de flexibilidade para reprodução e distribuição, com restrições adequadas. Tais fontes incluem alimentações ao vivo a partir de estações de televisão (através do ar, ou não através do ar, como CNN, e outros liberados mediante pagamento, como HBO, ou gratuitos). Tais fontes também incluem filmes ou programas de televisão, filmes caseiros, propagandas e, ainda, alimentações de teleconferência de vídeo ao vivo pré-gravados. As alimentações ao vivo seriam gerenciadas como a saída ao vivo de um Servidor de Jogo/Aplicação. Os material pré-gravado seria gerenciado como a saída de um Armazenamento Temporário de Atraso.

Em uma modalidade, os diversos módulos funcionais ilustrados aqui e as etapas associadas podem ser realizadas por componentes de hardware específicos que contêm lógica fisicamente conectada para realizar as etapas, como um circuito integrado específico de aplicação ("ASIC") ou através de qualquer combinação de componentes de computador programados e componentes de hardware personalizados.

Em uma modalidade, os módulos podem ser implantados em um processador de sinal digital programável ("DSP"), como uma arquitetura TMS320x da Texas Instruments (por exemplo, um TMS320C6000, TMS320C5000, etc). Diversos DSPs diferentes podem ser usados enquanto ainda estão em conformidade com esses princípios fundamentais.

As modalidades podem incluir diversas etapas, conforma estabelecido acima. Essas etapas podem ser incorporadas em instruções executáveis por máquina que fazem com que um processador de propósito geral ou de propósito especial realize determinadas etapas. Diversos elementos que não são relevantes para esses princípios fundamentais, como uma memória de computador, disco rígido, dispositivos de entrada, foram deixados de fora de algumas ou de todas as figuras para evitar o obscurecimento dos aspectos pertinentes.

Os elementos do assunto apresentado também podem ser fornecidos como um meio legível por máquina para armazenar as instruções

executáveis por máquina. O meio legível por máquina pode incluir, mas sem limitação, memória rápida, discos ópticos, CD-ROMs, DVD ROMs, RAMs, EPROMs, EEPROMs, cartões magnéticos ou ópticos, meio de propagação ou outro tipo de meio legível por máquina adequado para armazenar instruções eletrônicas. Por exemplo, a presente invenção pode ser transferida por download como um programa de computador que pode ser transferido a partir de um computador remoto (por exemplo, um servidor) para um computador que fez a solicitação (por exemplo, um cliente) por meio de sinais de dados incorporados em uma onda portadora ou outro meio de propagação através de um link de comunicação (por exemplo, um modem ou conexão de rede).

Deve-se compreender, ainda, que elementos do assunto apresentado também podem ser fornecidos como um produto de programa de computador que pode incluir um meio legível por máquina que tem instruções instaladas no mesmo que pode ser usado para programar um computador (por exemplo, um processador ou outro dispositivo eletrônico) para realizar uma sequência de operações. Alternativamente, as operações podem ser realizadas através de uma combinação de hardware e software. O meio legível por máquina pode incluir, mas sem limitação, disquetes, discos ópticos, CD-ROMs e discos magnético-ópticos, ROMs, RAMs, EPROMs, EEPROMs, cartões magnéticos ou ópticos, meio de propagação ou outro tipo de mídia/meio legível por máquina adequado para armazenar instruções eletrônicas. Por exemplo, elementos do assunto descrito a serem transferidos por download como um produto de programa de computador, em que o programa pode ser transferido a partir de um computador remoto ou dispositivo eletrônico para um progresso por solicitação através de sinais de dados incorporados em uma onda portadora ou outro meio de propagação por meio de um link de comunicação (por exemplo, um modem ou conexão de rede).

Além disso, embora o assunto apresentado tenha sido descrito em conjunto com modalidades específicas, diversas modificações e alterações são bem conhecidas no escopo da presente descrição. Consequen-

temente, o relatório descritivo e os desenhos devem ser considerados como ilustrativos ao invés de serem considerados em um sentido restritivo.

REIVINDICAÇÕES

1. Sistema implementado para computador para jogos online compreendendo:

um servidor de vídeo game ou aplicativo recebendo entradas de usuário relacionadas a um vídeo game online ou aplicativo transmitidos a partir de um cliente operado por um usuário, e executando responsivamente código do programa do vídeo game para renderizar uma sequência de imagens de vídeo resultante a partir da execução do vídeo game ou aplicativo;

um primeiro codificador de fluxo para comprimir a sequência de imagens de vídeo e gerar um fluxo de vídeo ao vivo durante uma sessão de jogo ao vivo com o usuário do dispositivo do cliente, o primeiro codificador de fluxo recebendo sinais de retorno do canal do dispositivo do cliente e responsivamente adaptando a compressão da sequência de imagens de vídeo baseada nos sinais de retorno de canal, o primeiro codificador de fluxo transmitindo continuamente o fluxo de vídeo ao vivo ao dispositivo do cliente durante a sessão de jogo ao vivo com o usuário, o dispositivo do cliente decodificando e renderizando o fluxo de vídeo ao vivo para visualização pelo usuário;

em que as operações de receber entradas de usuário transmitidas a partir de um dispositivo do cliente, executar o vídeo game ou aplicativo no servidor de vídeo game ou aplicativo, comprimir a sequência de imagens de vídeo e transmitir continuamente o fluxo de vídeo ao vivo pelo primeiro codificador de fluxo, e decodificar e renderizar o fluxo de vídeo ao vivo para visualização pelo usuário são realizadas de modo que o usuário tenha a percepção de que o vídeo game ou aplicativo selecionados estão respondendo instantaneamente à entrada do usuário recebida a partir do cliente;

um segundo codificador de fluxo para comprimir a sequência de imagens de vídeo em uma qualidade de vídeo específica e/ou taxa de compressão não relacionada ao sinal de retorno do canal durante a sessão de jogo ao vivo com o usuário, assim gerando qualidade alta de fluxo de vídeo (HQ), o fluxo de vídeo HQ tendo qualidade de vídeo relativamente maior e/ou taxa de compressão menor que o fluxo de vídeo ao vivo; e

um dispositivo de armazenamento para armazenar o fluxo de vídeo HQ para a subsequente reprodução ao usuário do dispositivo de cliente e a outros usuários por requisição.

2. Sistema, de acordo com a reivindicação 1, em que os sinais de retorno de canal indicam se os pacotes foram recebidos com sucesso e/ou descodificados pelo dispositivo do cliente, em que o primeiro codificador de fluxo adapta dinamicamente sua compressão sob a determinação de que um ou mais pacotes não foram recebidos e/ou não foram descodificados com sucesso.

3. Sistema, de acordo com a reivindicação 2, em que adaptar dinamicamente a compressão compreende gerar um novo I-quadro ou I-bloco.

4. Sistema, de acordo com a reivindicação 2, em que adaptar dinamicamente a compressão compreende a geração de um novo quadro/bloco comprimido para ser dependente de um quadro/bloco conhecido por ter sido recebido e/ou decodificado com sucesso no dispositivo do cliente.

5. Sistema, de acordo com a reivindicação 4, em que o novo quadro/bloco é um P-quadro/bloco e o último quadro/bloco conhecida por ter sido recebido e/ou descodificado com sucesso no dispositivo do cliente é um P-quadro/bloco.

6. Sistema, de acordo com a reivindicação 3, em que o codificador de fluxo ao vivo comprime a sequência de imagens de vídeo usando P-quadros/blocos até que os sinais de retorno de canal indiquem que um ou mais pacotes não foram recebidos e/ou descodificados com sucesso.

7. Sistema, de acordo com a reivindicação 1, em que o primeiro codificador de fluxo adapta a compressão da sequência de imagens de vídeo para reduzir a latência comunicada nos sinais de retorno de canal enquanto a sequência comprimida de imagens de vídeo é transmitida ao dispositivo do cliente.

8. Sistema, de acordo com a reivindicação 1, em que a qualidade de vídeo especificada compreende uma resolução específica em uma taxa de quadro específica.

9. Sistema, de acordo com a reivindicação 1, em que o dispositivo de armazenamento compreende um buffer de delay para armazenar temporariamente o fluxo de vídeo HQ antes da transmissão para um ou mais usuarios.

10. Sistema, de acordo com a reivindicação 1, compreendendo ainda:

lógica de escala e de recompressão para realizar escala e recompressar o fluxo de vídeo HQ para uma nova resolução especificada por um usuário.

11. Sistema, de acordo com a reivindicação 1, compreendendo ainda:

uma pluralidade de codificadores de fluxo adicionais para comprimir a sequência de imagens de vídeo em uma pluralidade correspondente de níveis de qualidade de vídeo especificada/ou taxas de compressão não relacionadas ao sinal de retorno de canal durante a sessão de jogo ao vivo com o usuário, gerando, desse modo, uma pluralidade de fluxos de vídeo de níveis de qualidade e/ou taxas de compressão diferentes.

12. Sistema, de acordo com a reivindicação 11, compreendendo ainda:

lógica de roteamento para selecionar um ou mais dos fluxos HQ e a pluralidade de fluxos adicionais; e

escala e/ou lógica de recompressão escala e/ou recompressão do fluxo HQ selecionado e/ou pluralidade de fluxos adicionais em uma resolução e/ou tamanho, respectivamente.

13. Sistema, de acordo com a reivindicação 1, compreendendo ainda:

lógica de transmissão múltipla para a transmissão múltipla do fluxo HQ para clientes de um ou mais usuários requerentes e/ou para um ou mais servidores de jogo sob pedido.

14. Sistema, de acordo com a reivindicação 1, compreendendo ainda:

a lógica de roteamento para rotear o fluxo HQ diretamente para

outros servidores de jogo e/ou clientes de usuários requerentes sob pedido e/ou para ler os fluxos HQ do dispositivo de armazenamento após um atraso e para rotear os fluxos HQ atrasados para outros servidores de jogo e/ou clientes de usuários requerentes sob pedido.

15. Sistema, de acordo com a reivindicação 1, compreendendo ainda:

lógica de mistura/mixagem para misturar/mixar áudio com o fluxo de vídeo HQ; e

lógica do roteamento para rotear o áudio misturado/mixado e fluxo de vídeo HQ para clientes dos usuários sob pedido.

16. Método implementado para computador para jogos online compreendendo as etapas de:

receber entradas de usuário a partir de um cliente relacionadas a um vídeo game online ou aplicativo e executar responsivamente o código de programa do vídeo game para renderizar uma sequência de imagens de vídeo resultantes a partir da execução do vídeo game ou aplicativo;

comprimir a sequência de imagens de vídeo e gerar um fluxo de vídeo ao vivo durante uma sessão de jogo ao vivo com um usuário de um dispositivo do cliente;

receber sinais de resposta de canal do dispositivo do cliente e responsivamente adaptar a compressão da sequência de imagens de vídeo baseadas nos sinais de resposta do canal,

continuamente transmitindo o fluxo de vídeo ao vivo para o dispositivo do cliente durante a sessão de jogo ao vivo com o usuário, o dispositivo do cliente decodificando e renderizando o fluxo de vídeo ao vivo para visualização pelo usuário;

em que as operações de receber entradas de usuário transmitidas a partir de um dispositivo do cliente, executar o vídeo game ou aplicativo no servidor de vídeo game ou aplicativo, comprimir a sequência de imagens de vídeo e transmitir continuamente o fluxo de vídeo ao vivo pelo primeiro codificador de fluxo, e decodificar e renderizar o fluxo de vídeo ao vivo para visualização pelo usuário são realizadas de modo que o usuário tenha a per-

cepção de que o vídeo game ou aplicativo selecionados estão respondendo instantaneamente à entrada do usuário recebida a partir do cliente;

comprimir a sequência de imagens de vídeo em uma qualidade de vídeo especificada/ou taxa de compressão não relacionada ao sinal de resposta do canal durante a sessão de jogo ao vivo com o usuário, gerando desse modo, um fluxo de vídeo da alta qualidade (HQ), o fluxo de vídeo HQ que tem uma qualidade de vídeo relativamente mais elevada e/ou menor taxa de compressão do que o fluxo de vídeo ao vivo; e

armazenar o fluxo de vídeo HQ em um dispositivo de armazenamento para a reprodução subsequente para o usuário do dispositivo do cliente e para outros usuários sob pedido.

17. Método, de acordo com a reivindicação 16, em que os sinais de retorno de canal indicam se os pacotes foram recebidos e/ou descodificados com sucesso pelo dispositivo do cliente, o método compreendendo ainda adaptar dinamicamente a compressão sob a determinação de que um ou mais pacotes não foram recebidos e/ou descodificados com sucesso.

18. Método, de acordo com a reivindicação 17, em que adaptar dinamicamente a compressão compreende a geração de um I-quadro ou de um I-bloco.

19. Método, de acordo com a reivindicação 17, em que adaptar dinamicamente a compressão compreende a geração de um novo quadro/bloco comprimido para ser dependente de um último quadro/bloco conhecido por ter sido recebido e/ou descodificado com sucesso no dispositivo do cliente.

20. Método, de acordo com a reivindicação 19, em que o novo quadro/bloco é um P-quadro/bloco e o último quadro/bloco conhecido por ter sido recebido e/ou descodificado com sucesso no dispositivo do cliente é um P-quadro/bloco.

21. Método, de acordo com a reivindicação 18, em que comprimir a sequência de imagens de vídeo e gerar um fluxo de vídeo ao vivo compreende ainda a compressão da sequência de imagens de vídeo usando P-quadros/blocos até que os sinais de retorno de canal indiquem que um ou

mais pacotes não foram recebidos e/ou não foram descodificados com sucesso.

22. Método, de acordo com a reivindicação 16, em que adaptar a compressão modificando ainda mais a compressão da sequência de imagens de vídeo para reduzir a latência comunicada nos sinais de retorno de canal enquanto a sequência comprimida de imagens de vídeo é transmitida ao dispositivo do cliente.

23. Método, de acordo com a reivindicação 16, em que a qualidade vídeo especificada compreende uma resolução específica em uma taxa de quadros específica.

24. Método, de acordo com a reivindicação 16, em que o dispositivo de armazenamento compreende um buffer de atraso para armazenar temporariamente o fluxo de vídeo HQ antes da transmissão para um ou mais usuários.

25. Método, de acordo com a reivindicação 16, compreendendo ainda escalar e recomprimir o fluxo de vídeo HQ em uma nova resolução especificada por um usuário.

26. Método, de acordo com a reivindicação 16, compreendendo ainda comprimir a sequência de imagens de vídeo em uma pluralidade de níveis de qualidade de vídeo especificada e/ou taxa de compressão não relacionadas ao sinal de retorno de canal durante a sessão de jogo ao vivo com o usuário, gerando desse modo uma pluralidade dos fluxos vídeo de níveis de qualidade e/ou de taxas de compressão diferentes.

27. Método, de acordo com a reivindicação 26, compreendendo ainda:

selecionar um ou mais dos fluxos HQ e a pluralidade de fluxos adicionais; e

escalar e/ou recomprimir o fluxo HQ e/ou a pluralidade de fluxos adicionais selecionados em uma resolução e/ou tamanho especificado, respectivamente.

28. Método, de acordo com a reivindicação 16, compreendendo ainda:

multitransmitir o fluxo HQ para os clientes de um ou mais usuários requerentes e/ou para um ou mais servidores de jogo sob solicitação.

29. Método, de acordo com a reivindicação 16, compreendendo ainda:

rotear os fluxos HQ diretamente para outros servidores do jogo e/ou clientes de usuários requerentes sob pedido e/ou para ler os fluxos HQ do dispositivo de armazenamento após um atraso e rotear os fluxos HQ atrasados para outros servidores do jogo e/ou clientes de usuários requerentes sob pedido.

30. Método, de acordo com a reivindicação 16, compreendendo ainda:

misturar/mixar áudio com o fluxo de vídeo HQ; e

rotear o áudio misturado/mixado e o fluxo de vídeo HQ para os clientes dos usuários sob pedido.

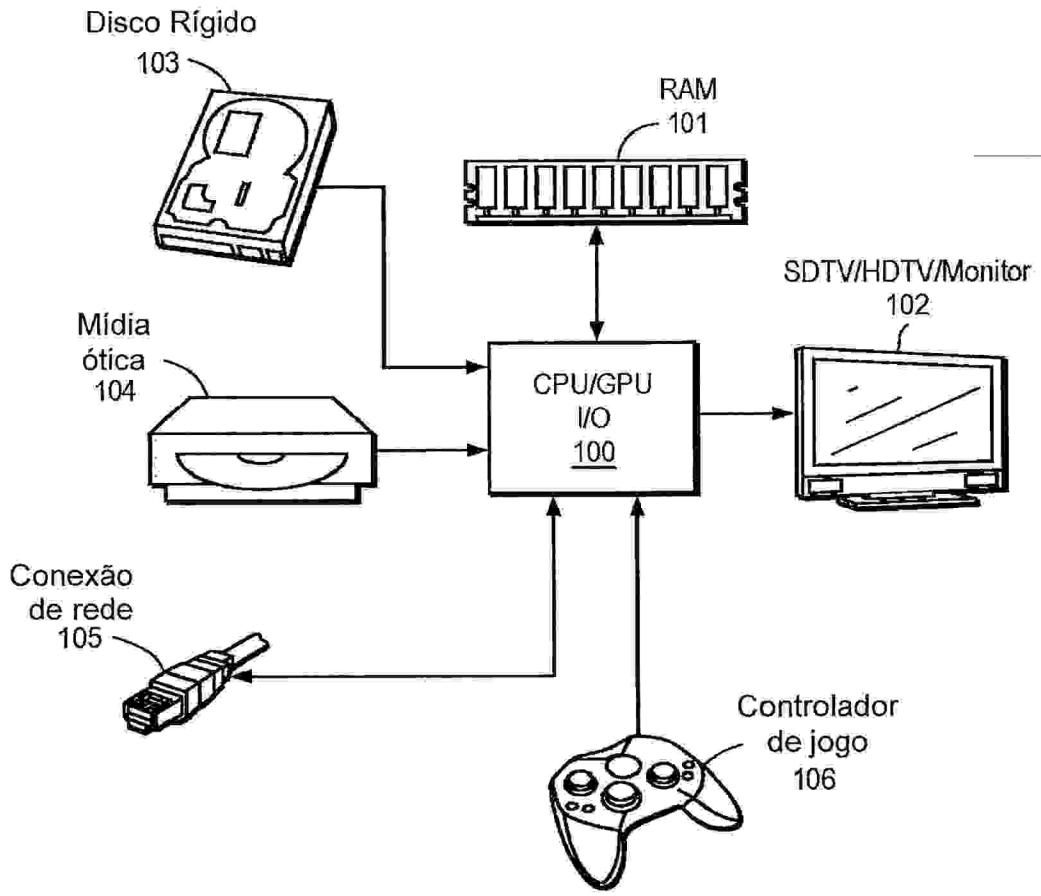


FIG. 1
(Técnica anterior)

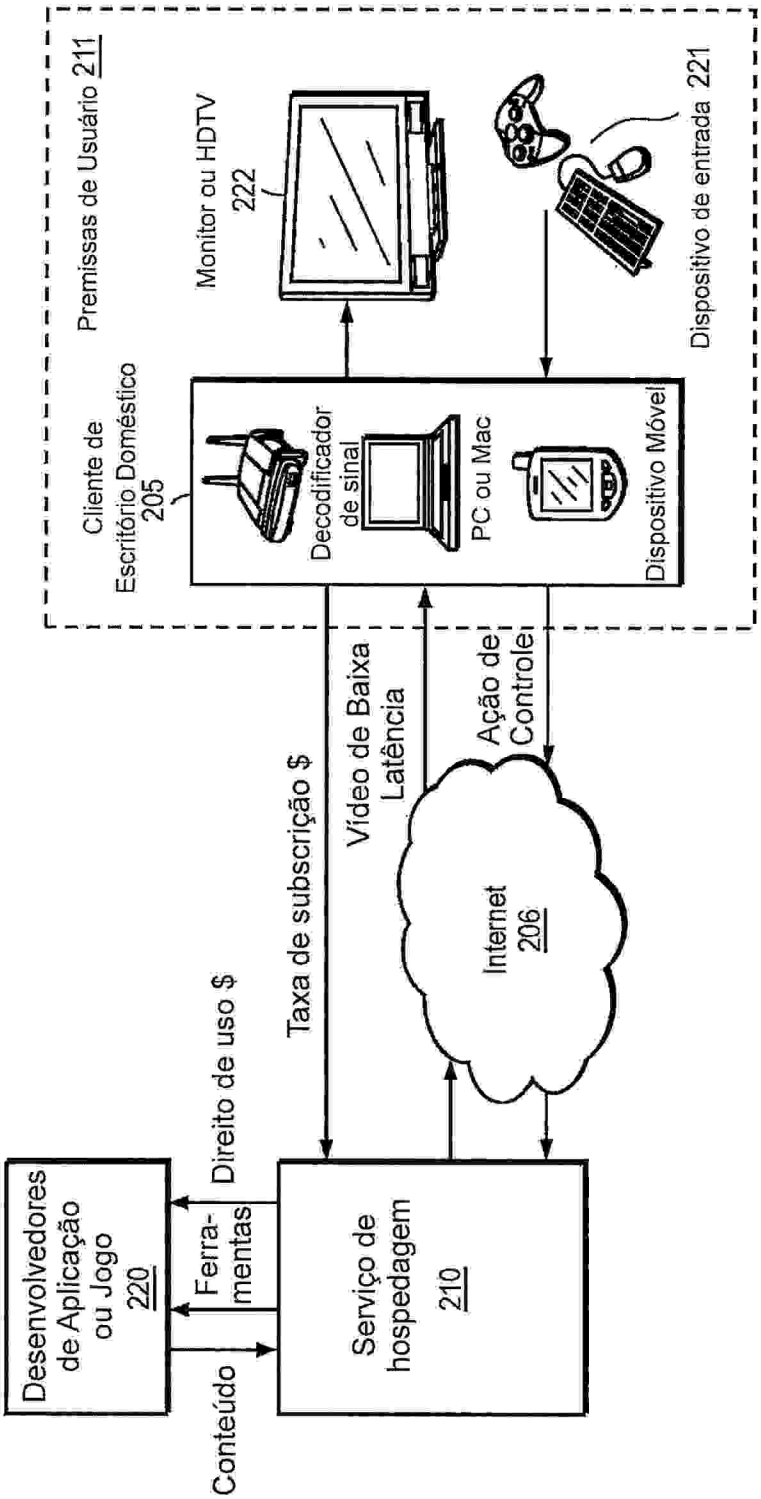

Premissas de Usuário 211
Cliente de Escritório Doméstico 205
Decodificador de sinal
PC ou Mac
Dispositivo Móvel
Monitor ou HDTV 222
Dispositivo de entrada 221
Taxa de subscrição $
Vídeo de Baixa Latência
Ação de Controle
Internet 206
Direito de uso $
Ferra-mentas
Conteúdo
Desenvolvedores de Aplicação ou Jogo 220
Serviço de hospedagem 210

FIG. 2a

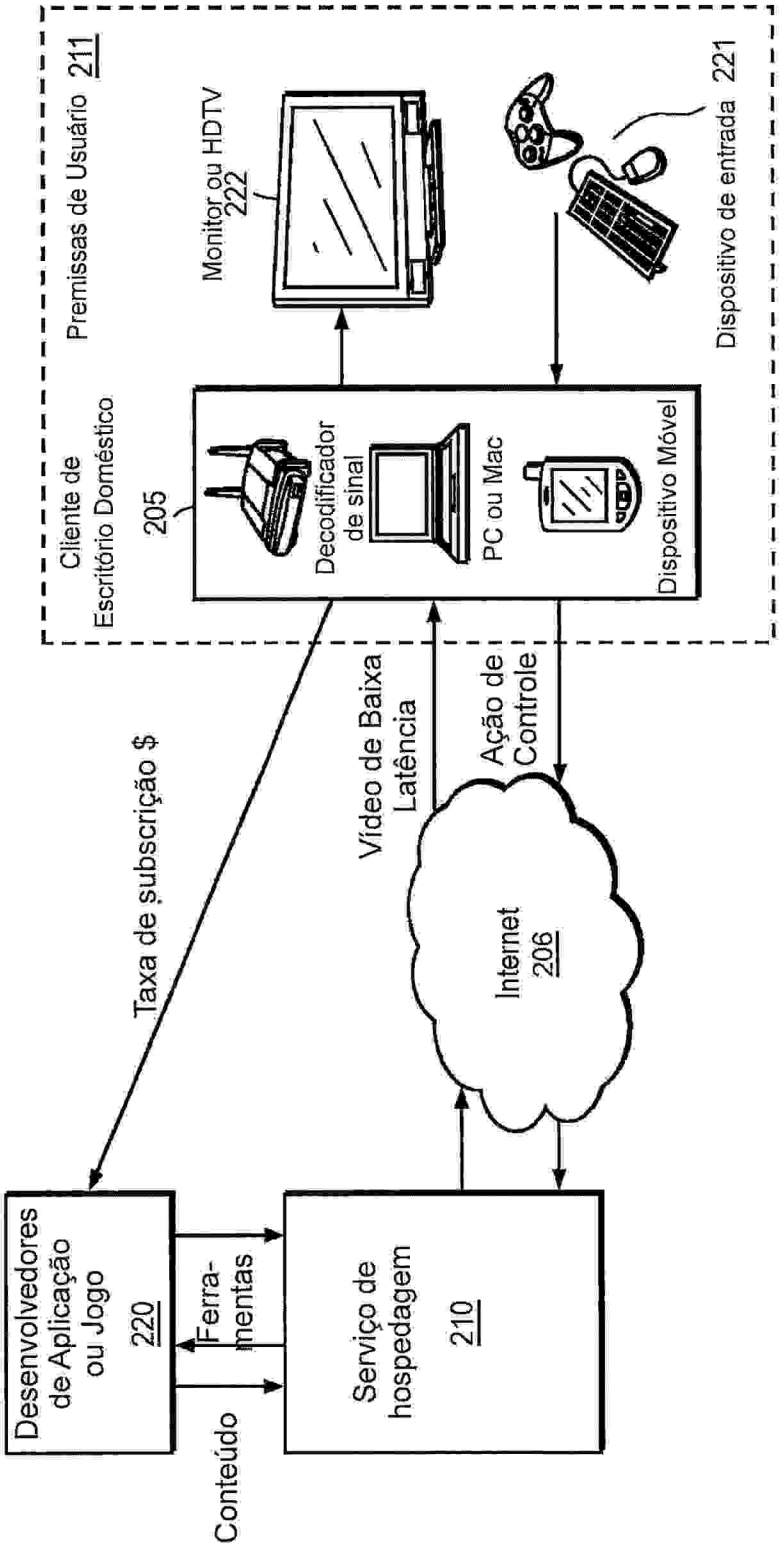

Premissas de Usuário 211
Cliente de Escritório Doméstico
205
Monitor ou HDTV 222
Decodificador de sinal
PC ou Mac
Dispositivo Móvel
Dispositivo de entrada 221
Taxa de subscrição $
Vídeo de Baixa Latência
Ação de Controle
Internet 206
Desenvolvedores de Aplicação ou Jogo 220
Ferra-mentas
Serviço de hospedagem 210
Conteúdo

FIG. 2b

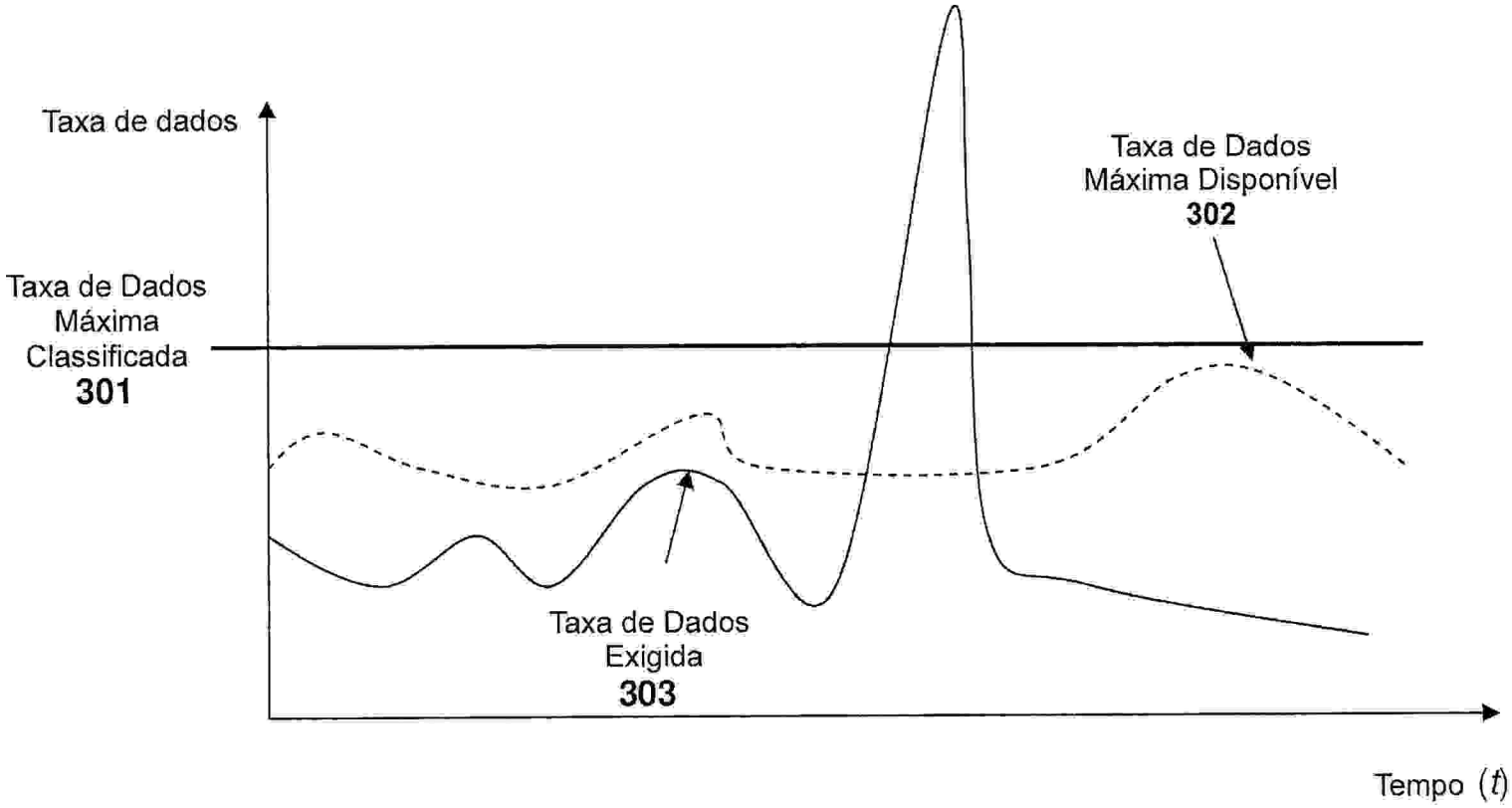

Taxa de dados
Taxa de Dados
Máxima
Classificada
301
Taxa de Dados
Máxima Disponível
302
Taxa de Dados
Exigida
303
Tempo (t)

Fig. 3

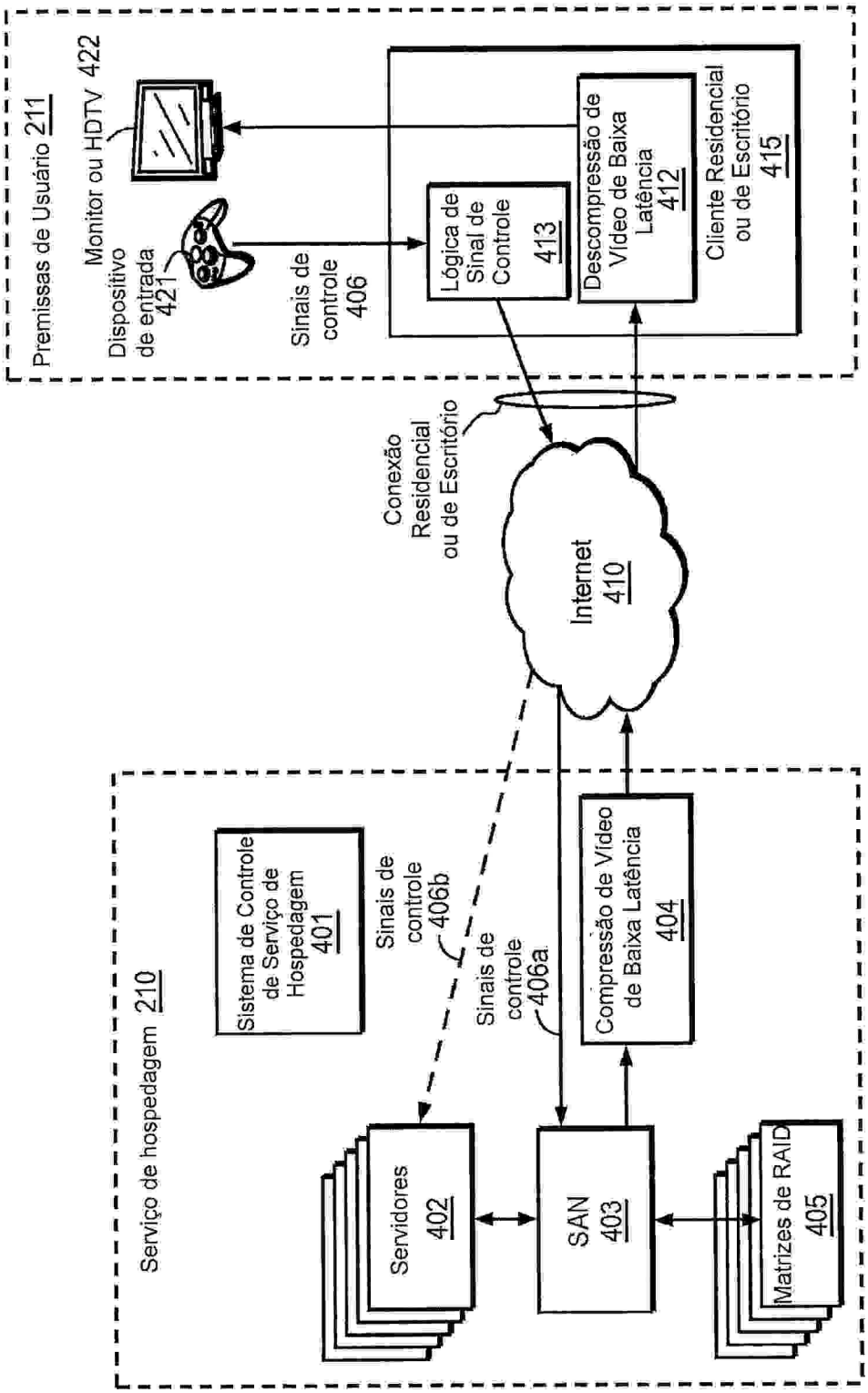

Premissas de Usuário 211
Monitor ou HDTV 422
Dispositivo de entrada 421
Sinais de controle 406
Lógica de Sinal de Controle 413
Descompressão de Vídeo de Baixa Latência 412
Cliente Residencial ou de Escritório 415
Conexão Residencial ou de Escritório
Internet 410
Serviço de hospedagem 210
Sistema de Controle de Serviço de Hospedagem 401
Sinais de controle 406b
Sinais de controle 406a
Compressão de Vídeo de Baixa Latência 404
Servidores 402
SAN 403
Matrizes de RAID 405

FIG. 4a

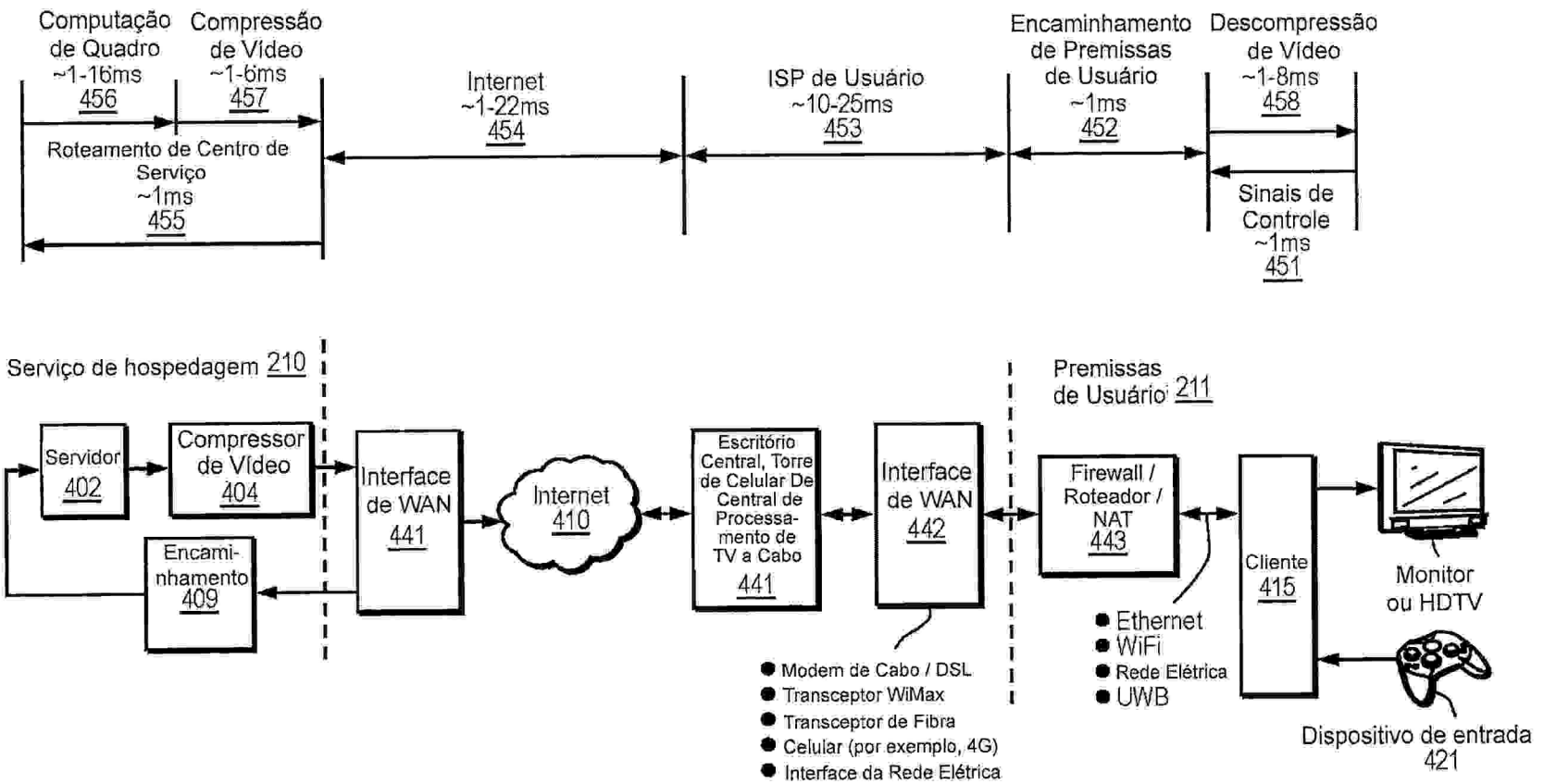

Computação de Quadro ~1-16ms 456
Compressão de Vídeo ~1-6ms 457
Roteamento de Centro de Serviço ~1ms 455
Internet ~1-22ms 454
ISP de Usuário ~10-25ms 453
Encaminhamento de Premissas de Usuário ~1ms 452
Descompressão de Vídeo ~1-8ms 458
Sinais de Controle ~1ms 451
Serviço de hospedagem 210
Premissas de Usuário 211
Servidor 402
Compressor de Vídeo 404
Encami-nhamento 409
Interface de WAN 441
Internet 410
Escritório Central, Torre de Celular De Central de Processa-mento de TV a Cabo 441
Interface de WAN 442
Modem de Cabo / DSL
Transceptor WiMax
Transceptor de Fibra
Celular (por exemplo, 4G)
Interface da Rede Elétrica
Firewall / Roteador / NAT 443
Ethernet
WiFi
Rede Elétrica
UWB
Cliente 415
Monitor ou HDTV
Dispositivo de entrada 421

FIG. 4b

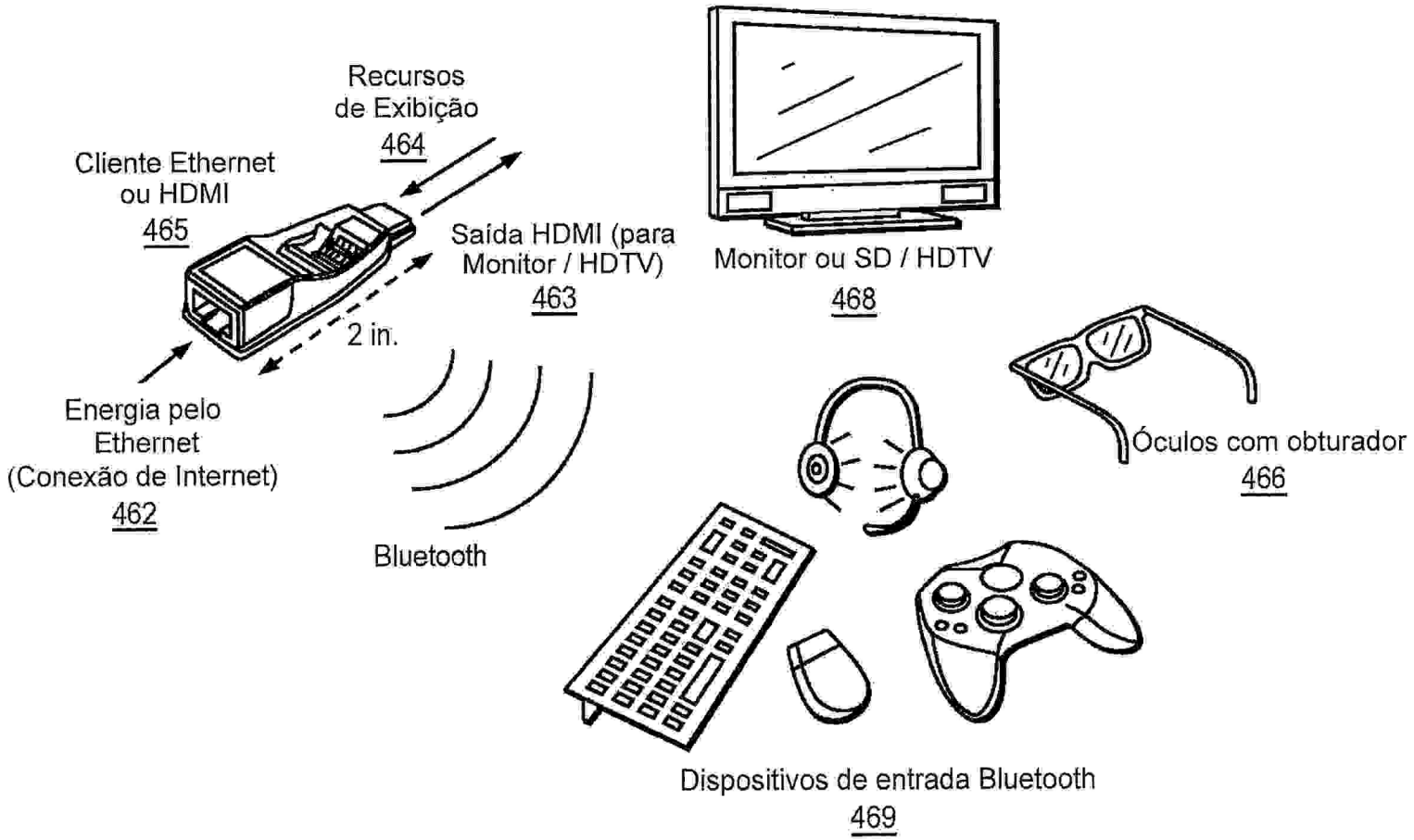

Recursos
de Exibição
464
Cliente Ethernet
ou HDMI
465
Saída HDMI (para
Monitor / HDTV)
463
Monitor ou SD / HDTV
468
2 in.
Energia pelo
Ethernet
(Conexão de Internet)
462
Bluetooth
Óculos com obturador
466
Dispositivos de entrada Bluetooth
469

FIG. 4c

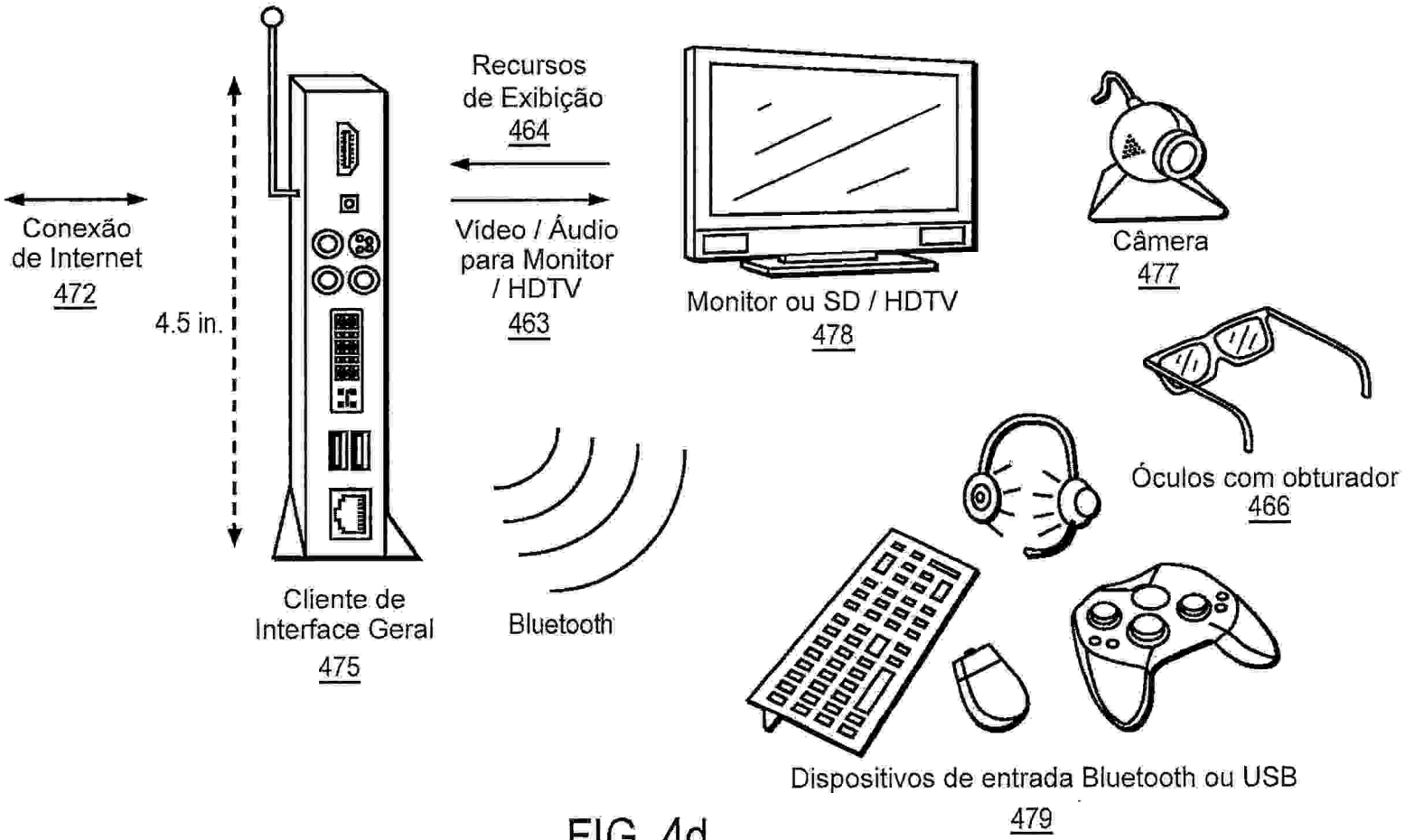

Conexão
de Internet
472
4.5 in.
Cliente de
Interface Geral
475
Recursos
de Exibição
464
Vídeo / Áudio
para Monitor
/ HDTV
463
Bluetooth
Monitor ou SD / HDTV
478
Câmera
477
Óculos com obturador
466
Dispositivos de entrada Bluetooth ou USB
479

FIG. 4d

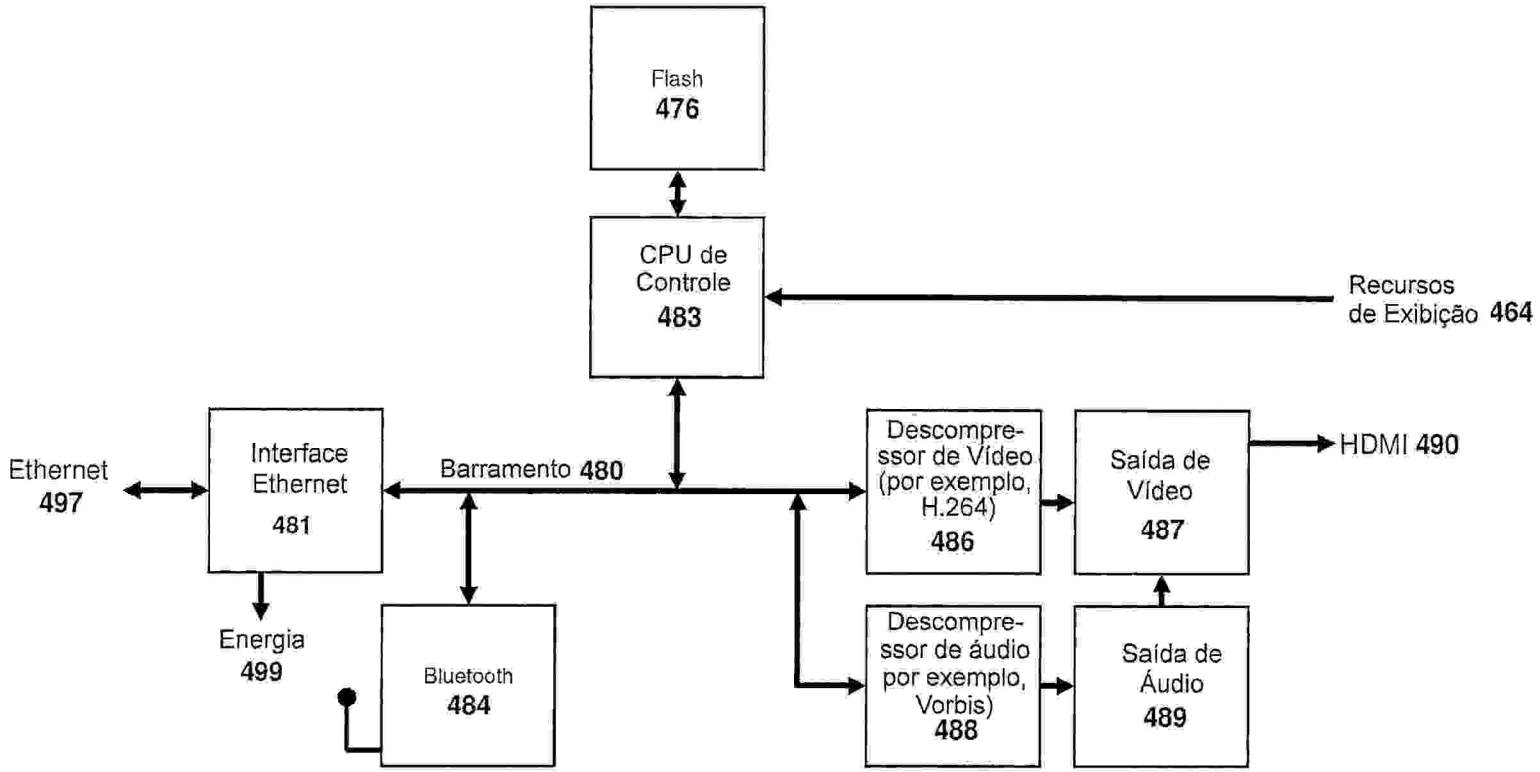

Flash
476
CPU de
Controle
483
Recursos
de Exibição 464
Ethernet
497
Interface
Ethernet
481
Barramento 480
Descompre-
ssor de Vídeo
(por exemplo,
H.264)
486
Saída de
Vídeo
487
HDMI 490
Energia
499
Bluetooth
484
Descompre-
ssor de áudio
por exemplo,
Vorbis)
488
Saída de
Áudio
489

Fig. 4e

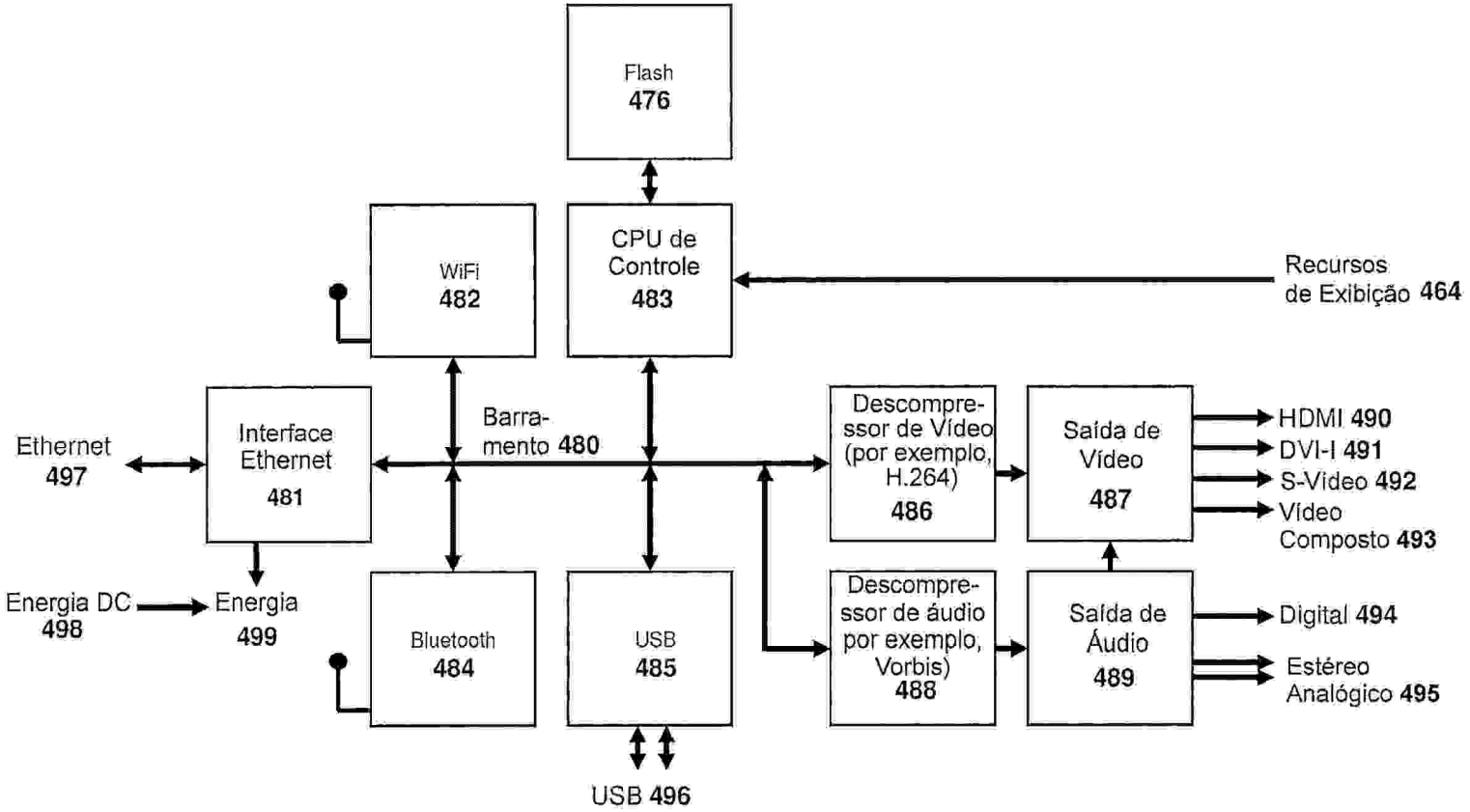

Flash
476
WiFi
482
CPU de
Controle
483
Recursos
de Exibição 464
Ethernet
497
Interface
Ethernet
481
Barra-
mento 480
Descompre-
ssor de Vídeo
(por exemplo,
H.264)
486
Saída de
Vídeo
487
HDMI 490
DVI-I 491
S-Video 492
Vídeo
Composto 493
Energia DC
498
Energia
499
Bluetooth
484
USB
485
Descompre-
ssor de áudio
por exemplo,
Vorbis)
488
Saída de
Áudio
489
Digital 494
Estéreo
Analógico 495
USB 496

Fig. 4f

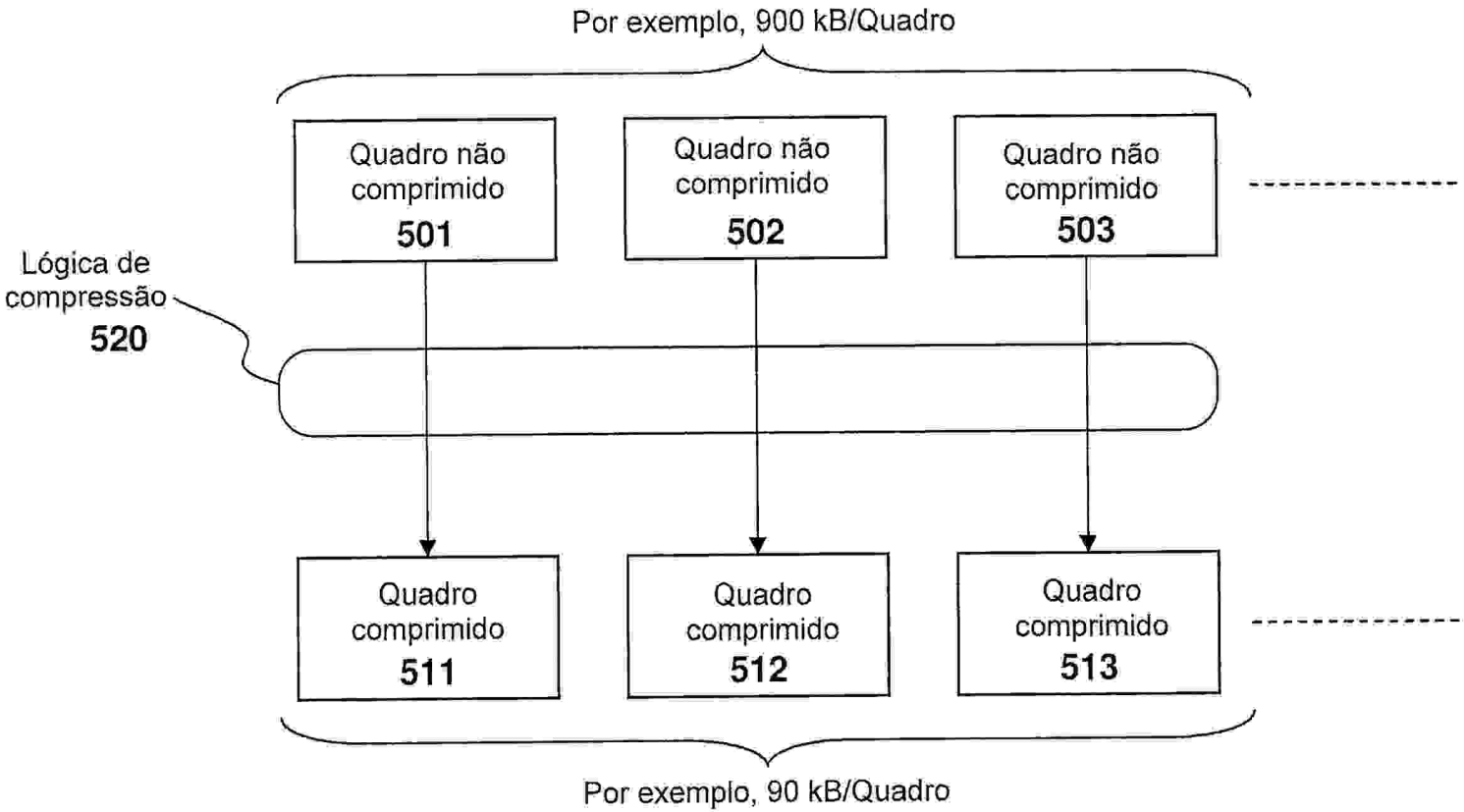


Fig. 5

(Técnica anterior)

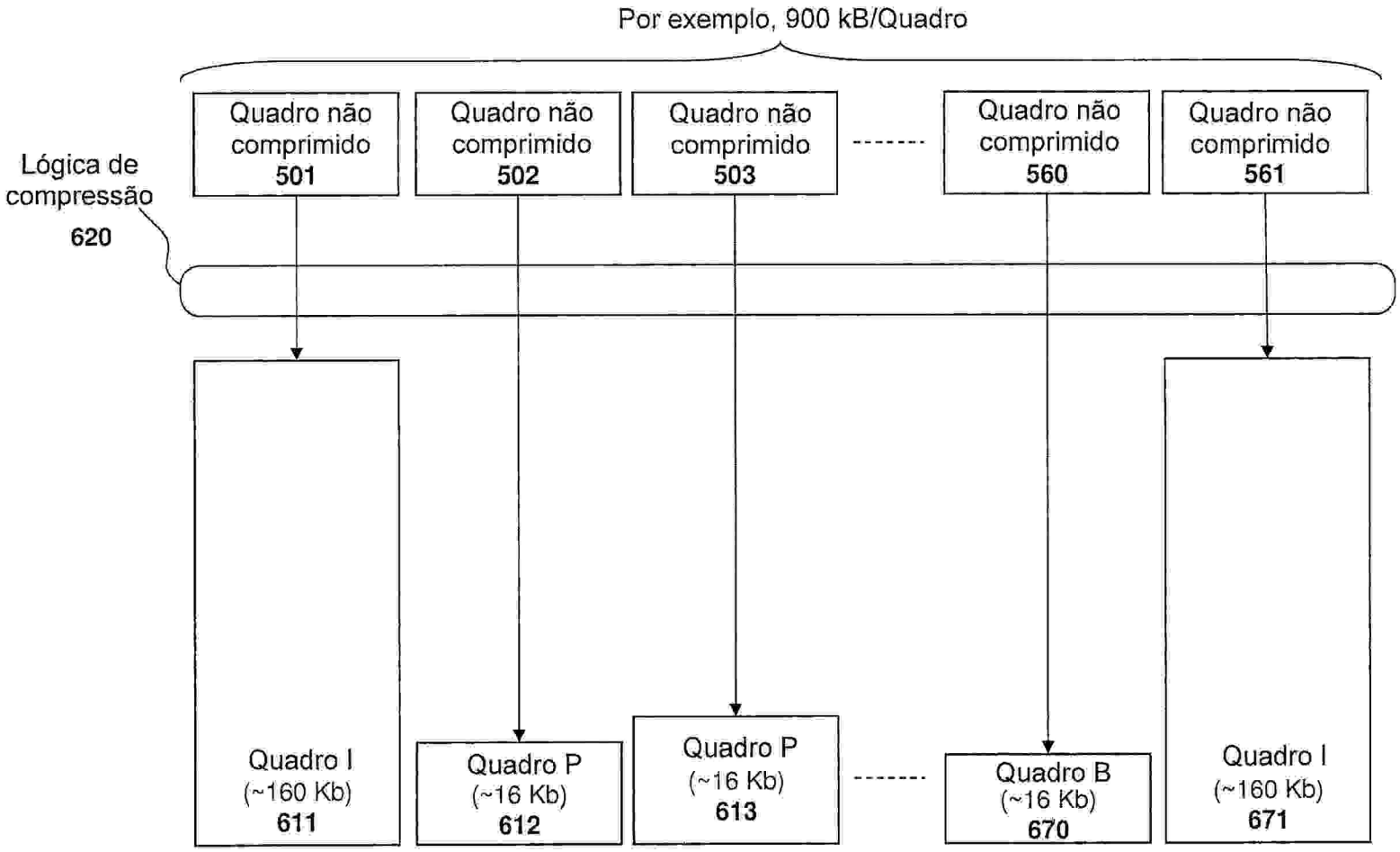


Fig. 6a

(Técnica anterior)

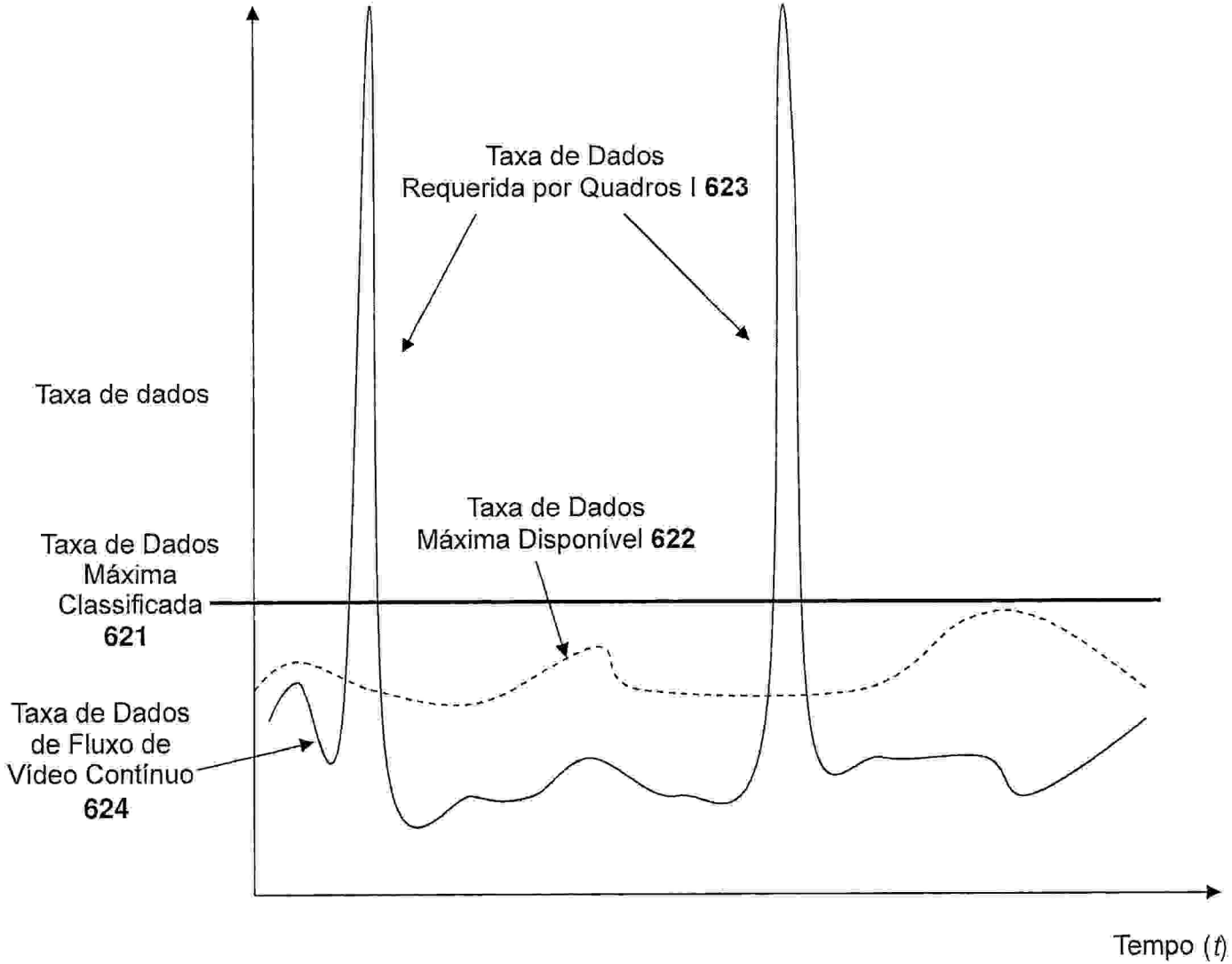

Taxa de Dados
Requerida por Quadros I 623
Taxa de dados
Taxa de Dados
Máxima Disponível 622
Taxa de Dados
Máxima
Classificada
621
Taxa de Dados
de Fluxo de
Vídeo Contínuo
624
Tempo (t)

Fig. 6b

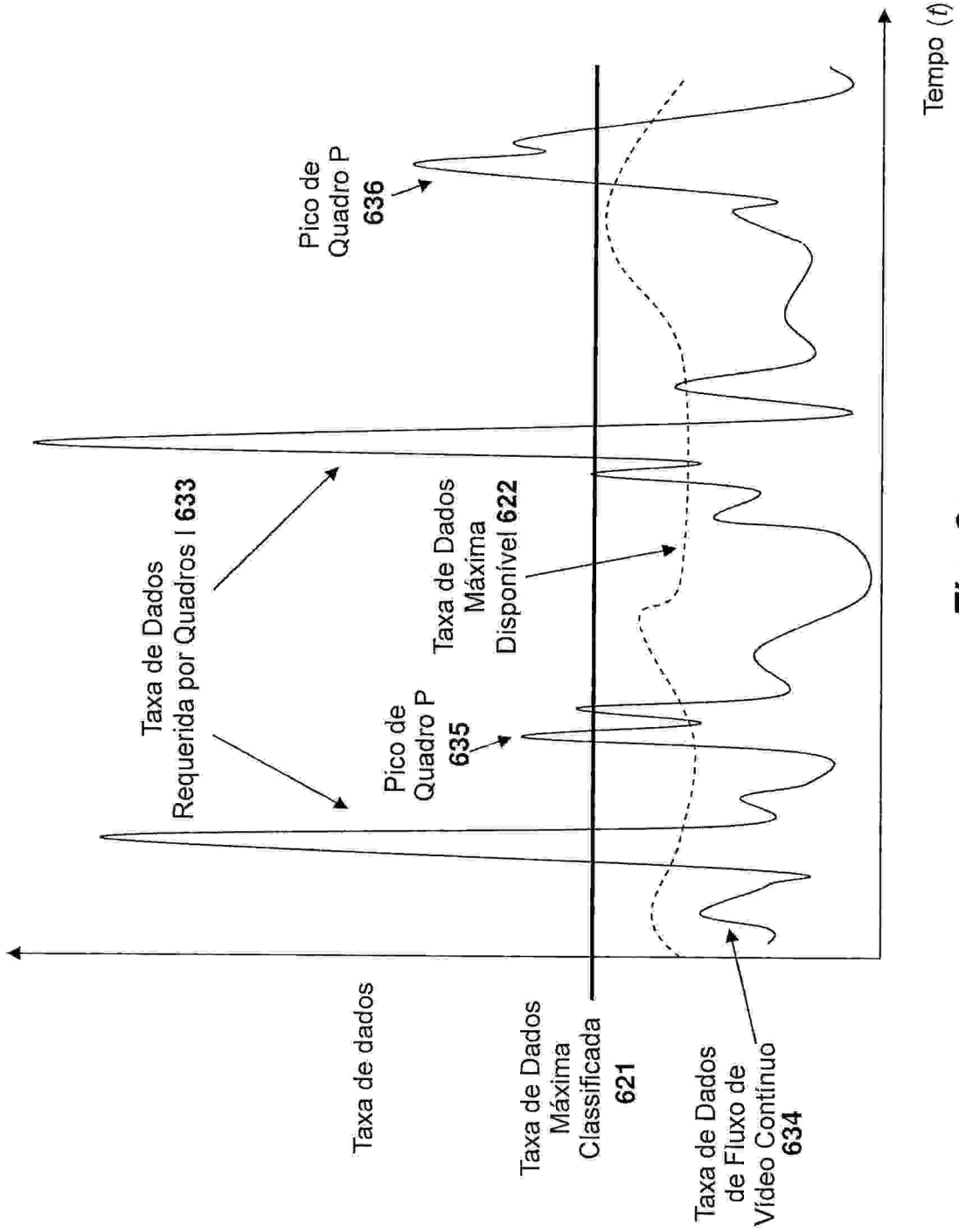

Taxa de dados
Tempo (t)
Taxa de Dados Requerida por Quadros I 633
Pico de Quadro P 635
Pico de Quadro P 636
Taxa de Dados Máxima Disponível 622
Taxa de Dados Máxima Classificada 621
Taxa de Dados de Fluxo de Vídeo Contínuo 634

Fig. 6c

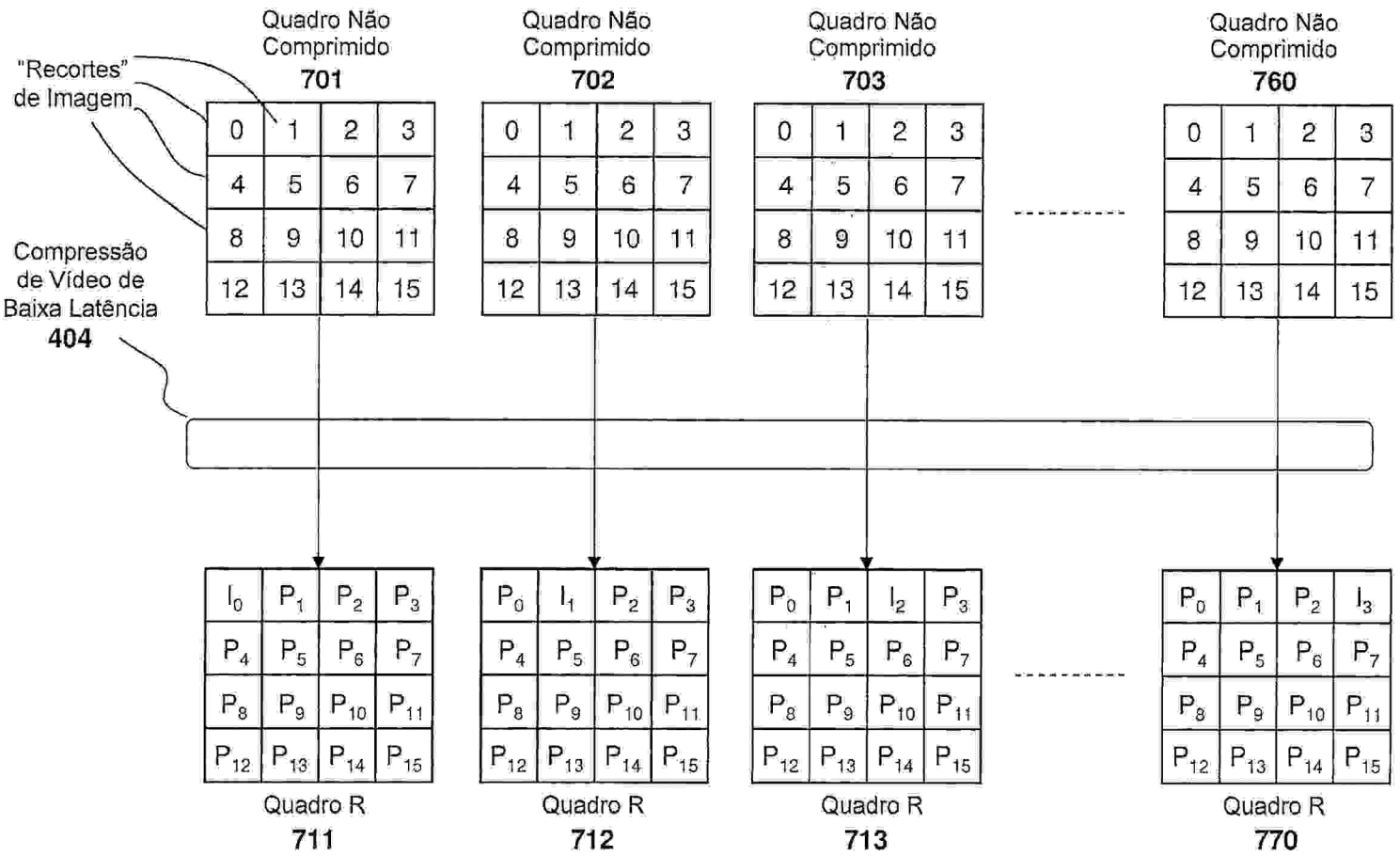

"Recortes" de Imagem
Compressão de Vídeo de Baixa Latência
404
Quadro Não Comprimido
701
Quadro Não Comprimido
702
Quadro Não Comprimido
703
Quadro Não Comprimido
760
0 1 2 3
4 5 6 7
8 9 10 11
12 13 14 15
I0 P1 P2 P3
P4 P5 P6 P7
P8 P9 P10 P11
P12 P13 P14 P15
P0 I1 P2 P3
P0 P1 I2 P3
P0 P1 P2 I3
Quadro R
711
Quadro R
712
Quadro R
713
Quadro R
770

Fig. 7a

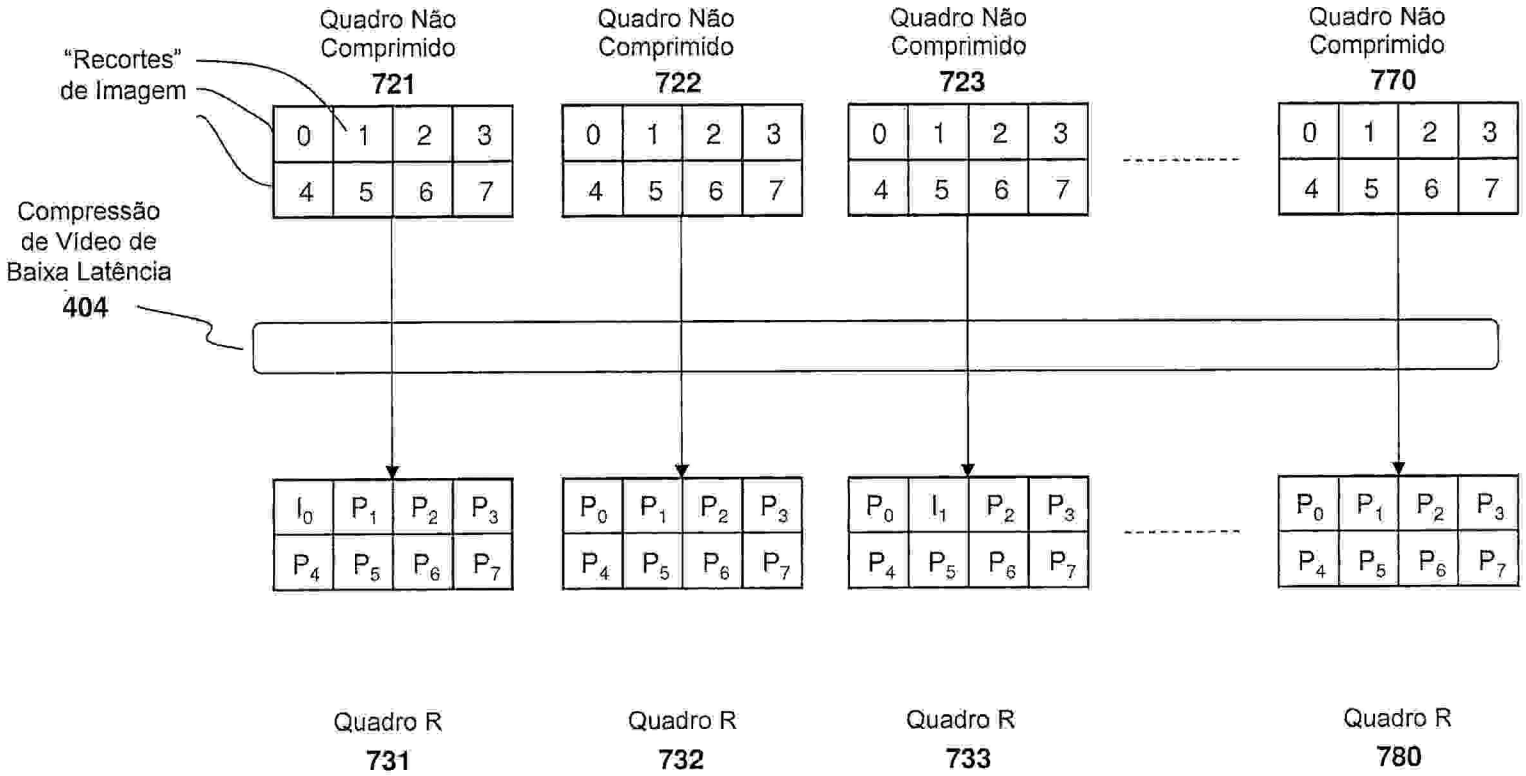

"Recortes" de Imagem
Quadro Não Comprimido
721
Quadro Não Comprimido
722
Quadro Não Comprimido
723
Quadro Não Comprimido
770
0 1 2 3
4 5 6 7
Compressão de Vídeo de Baixa Latência
404
I0 P1 P2 P3
P4 P5 P6 P7
P0 P1 P2 P3
P4 P5 P6 P7
P0 I1 P2 P3
P4 P5 P6 P7
P0 P1 P2 P3
P4 P5 P6 P7
Quadro R
731
Quadro R
732
Quadro R
733
Quadro R
780

Fig. 7b

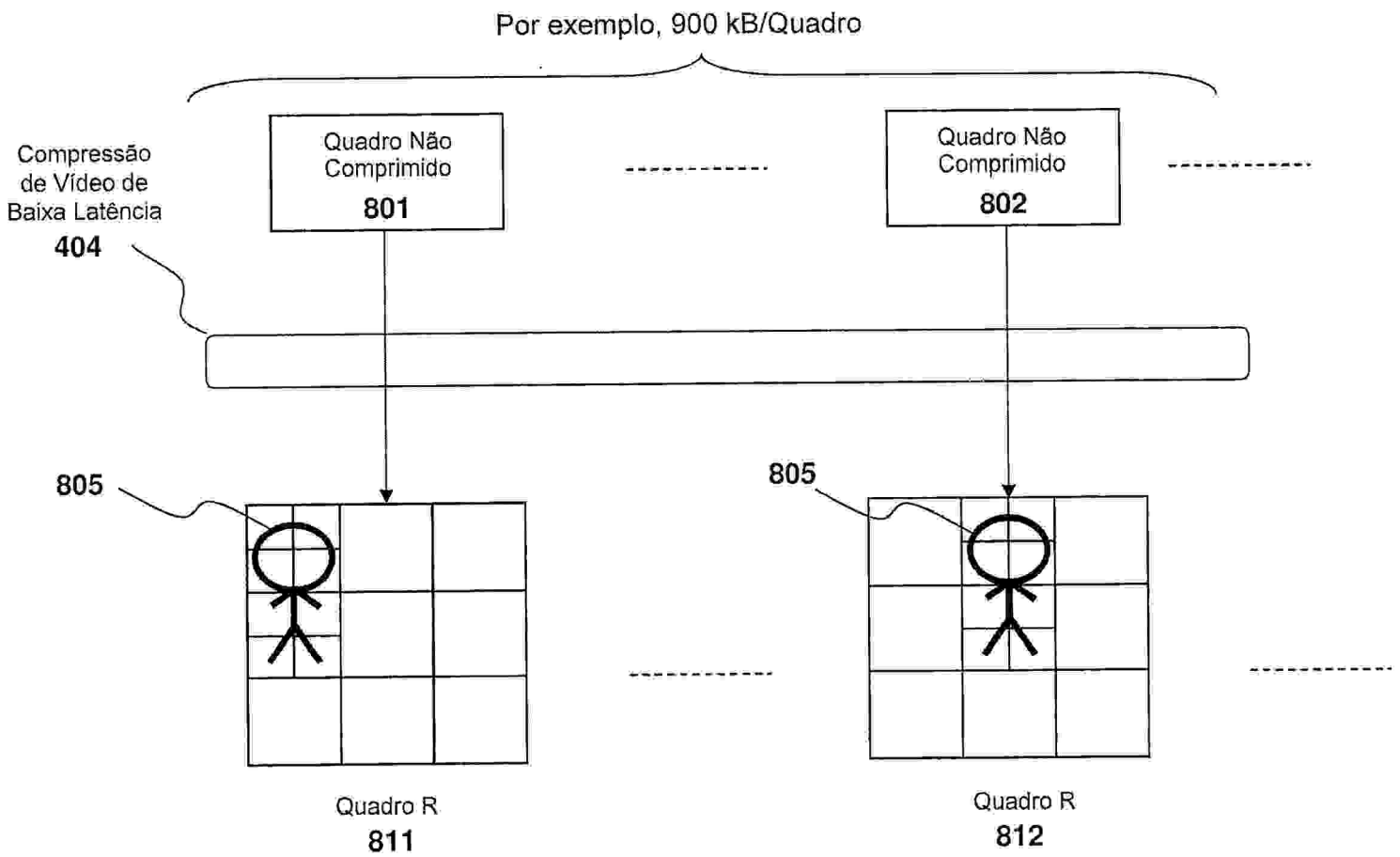

Por exemplo, 900 kB/Quadro
Compressão de Vídeo de Baixa Latência
404
Quadro Não Comprimido
801
Quadro Não Comprimido
802
805
805
Quadro R
811
Quadro R
812

Fig. 8

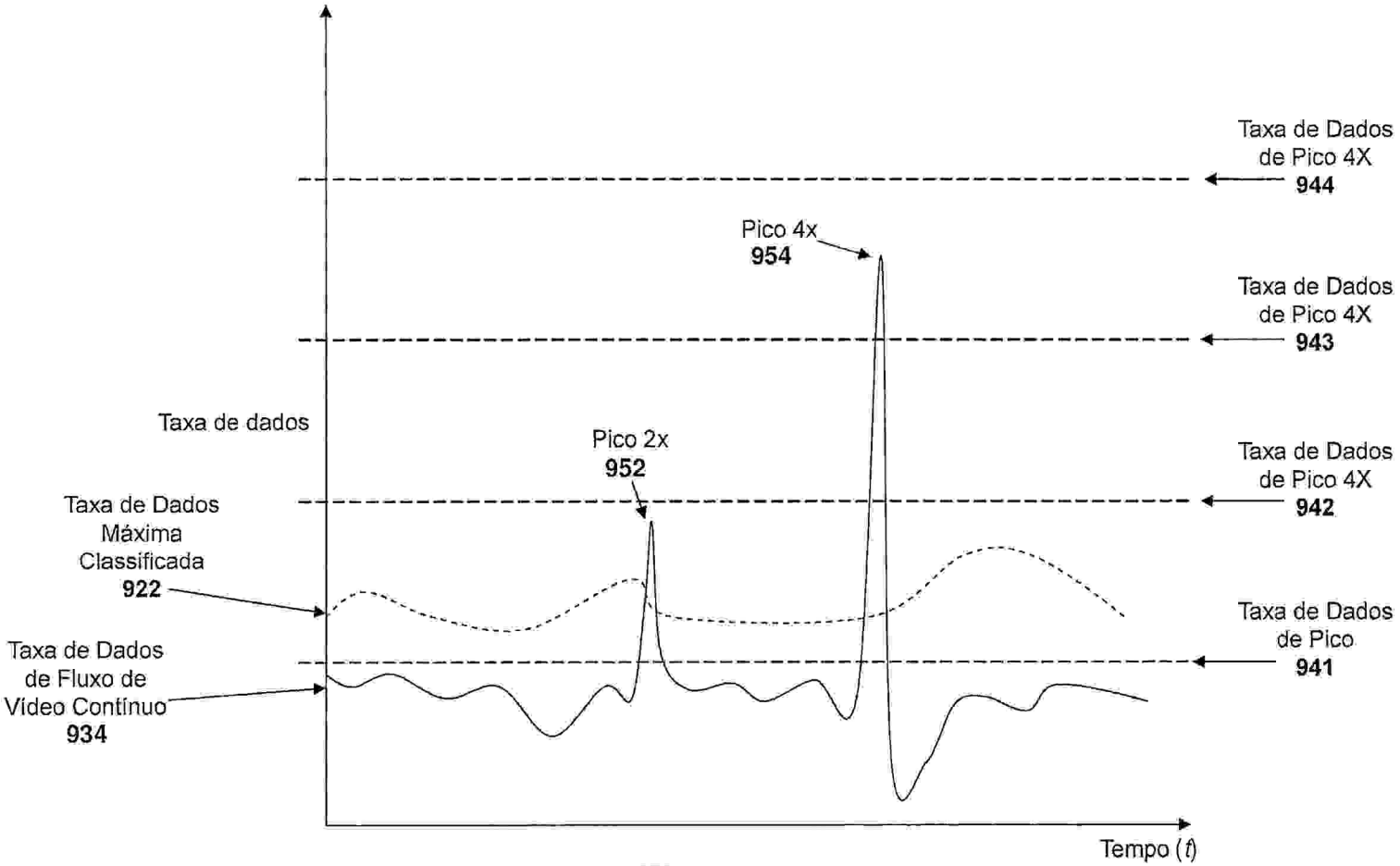

Taxa de Dados
de Pico 4X
944
Pico 4x
954
Taxa de Dados
de Pico 4X
943
Taxa de dados
Pico 2x
952
Taxa de Dados
de Pico 4X
942
Taxa de Dados
Máxima
Classificada
922
Taxa de Dados
de Pico
941
Taxa de Dados
de Fluxo de
Vídeo Contínuo
934
Tempo (t)

Fig. 9a

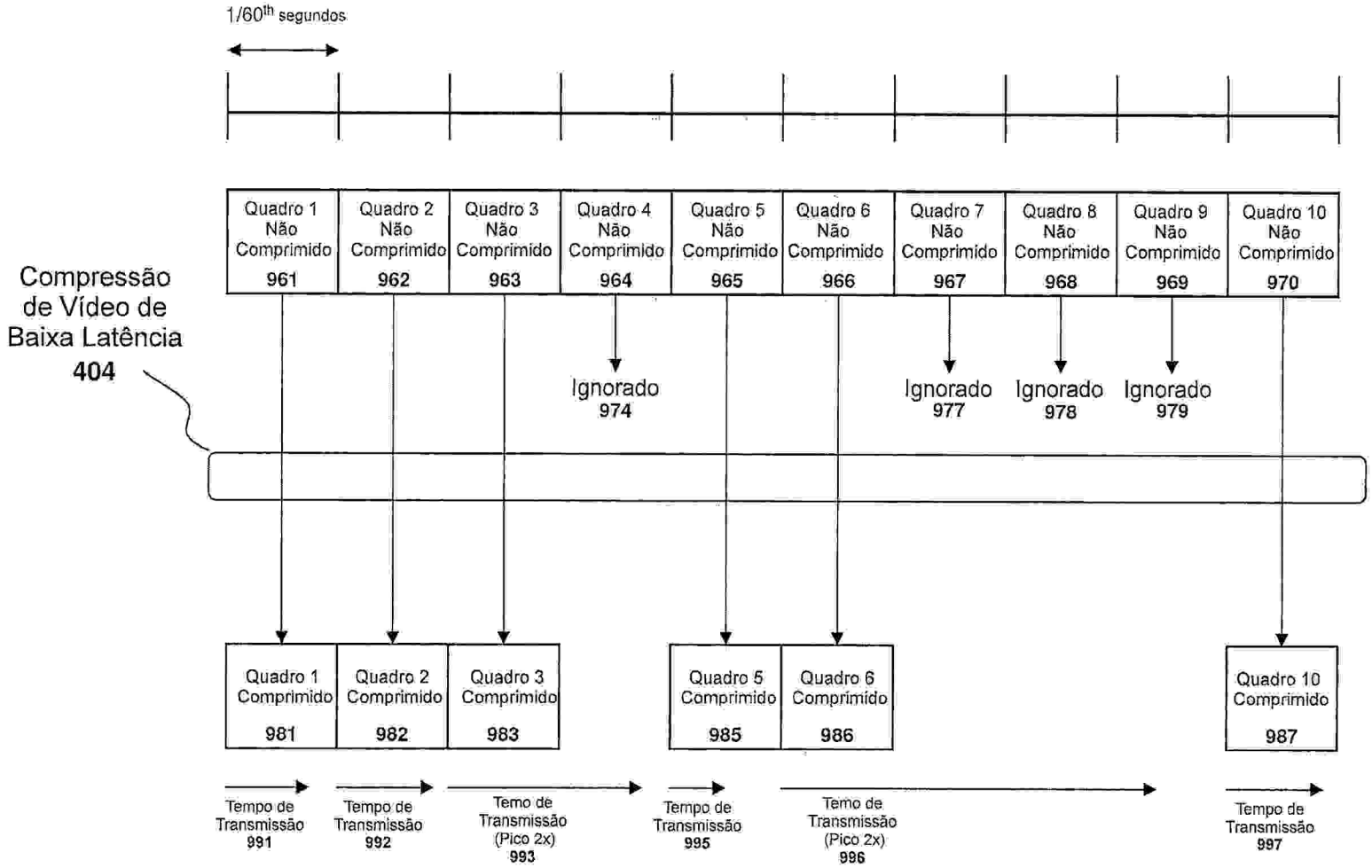

1/60th segundos
Compressão de Vídeo de Baixa Latência
404
Quadro 1 Não Comprimido 961
Quadro 2 Não Comprimido 962
Quadro 3 Não Comprimido 963
Quadro 4 Não Comprimido 964
Quadro 5 Não Comprimido 965
Quadro 6 Não Comprimido 966
Quadro 7 Não Comprimido 967
Quadro 8 Não Comprimido 968
Quadro 9 Não Comprimido 969
Quadro 10 Não Comprimido 970
Ignorado 974
Ignorado 977
Ignorado 978
Ignorado 979
Quadro 1 Comprimido 981
Quadro 2 Comprimido 982
Quadro 3 Comprimido 983
Quadro 5 Comprimido 985
Quadro 6 Comprimido 986
Quadro 10 Comprimido 987
Tempo de Transmissão 991
Tempo de Transmissão 992
Temo de Transmissão (Pico 2x) 993
Tempo de Transmissão 995
Temo de Transmissão (Pico 2x) 996
Tempo de Transmissão 997

Fig. 9b

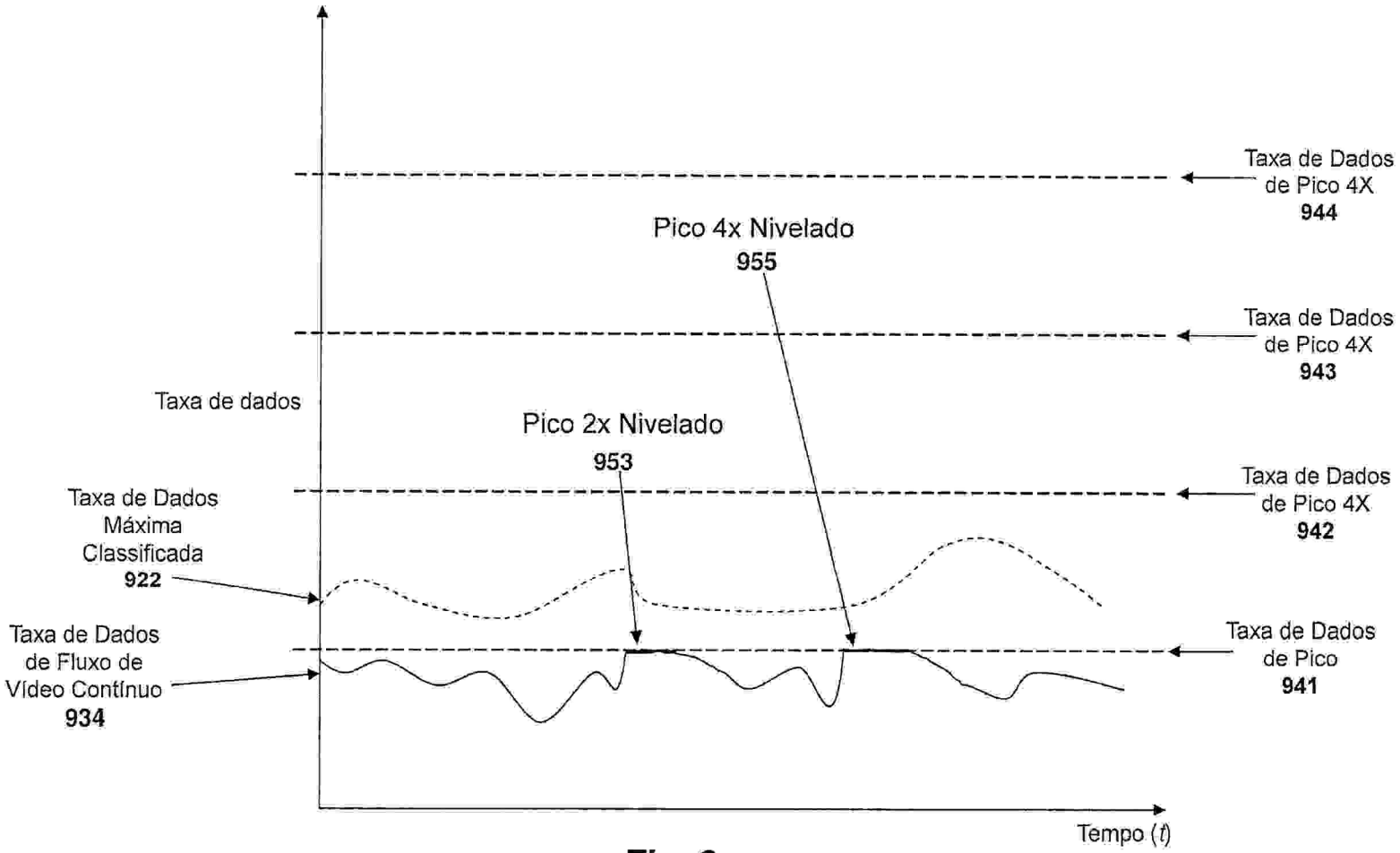

Taxa de Dados
de Pico 4X
944
Pico 4x Nivelado
955
Taxa de Dados
de Pico 4X
943
Taxa de dados
Pico 2x Nivelado
953
Taxa de Dados
de Pico 4X
942
Taxa de Dados
Máxima
Classificada
922
Taxa de Dados
de Fluxo de
Vídeo Contínuo
934
Taxa de Dados
de Pico
941
Tempo (t)

Fig. 9c

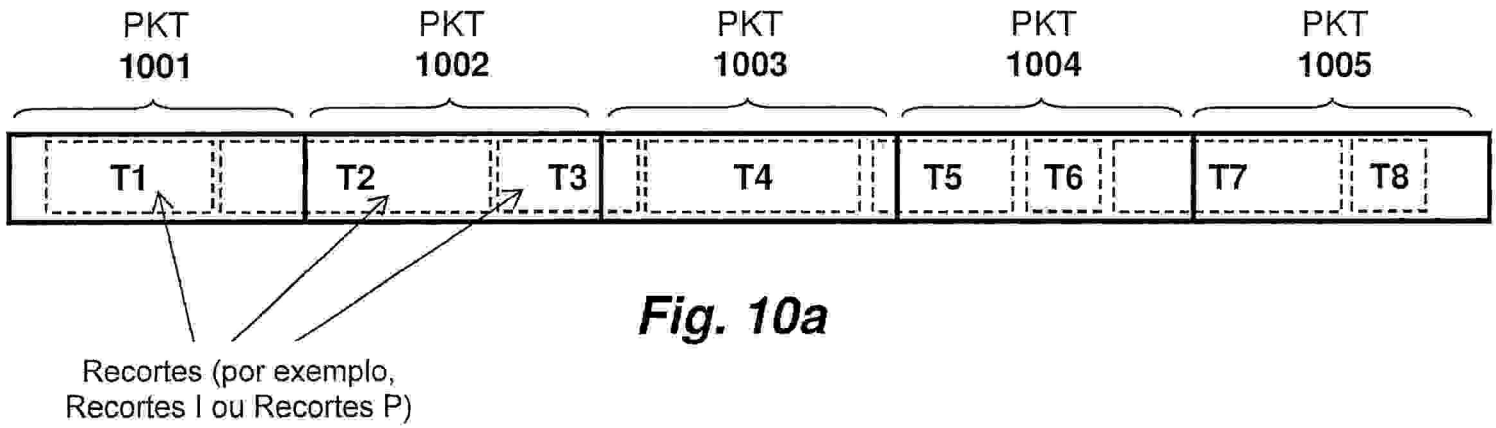

PKT
1001
PKT
1002
PKT
1003
PKT
1004
PKT
1005
T1
T2
T3
T4
T5
T6
T7
T8
Recortes (por exemplo,
Recortes I ou Recortes P)

Fig. 10a

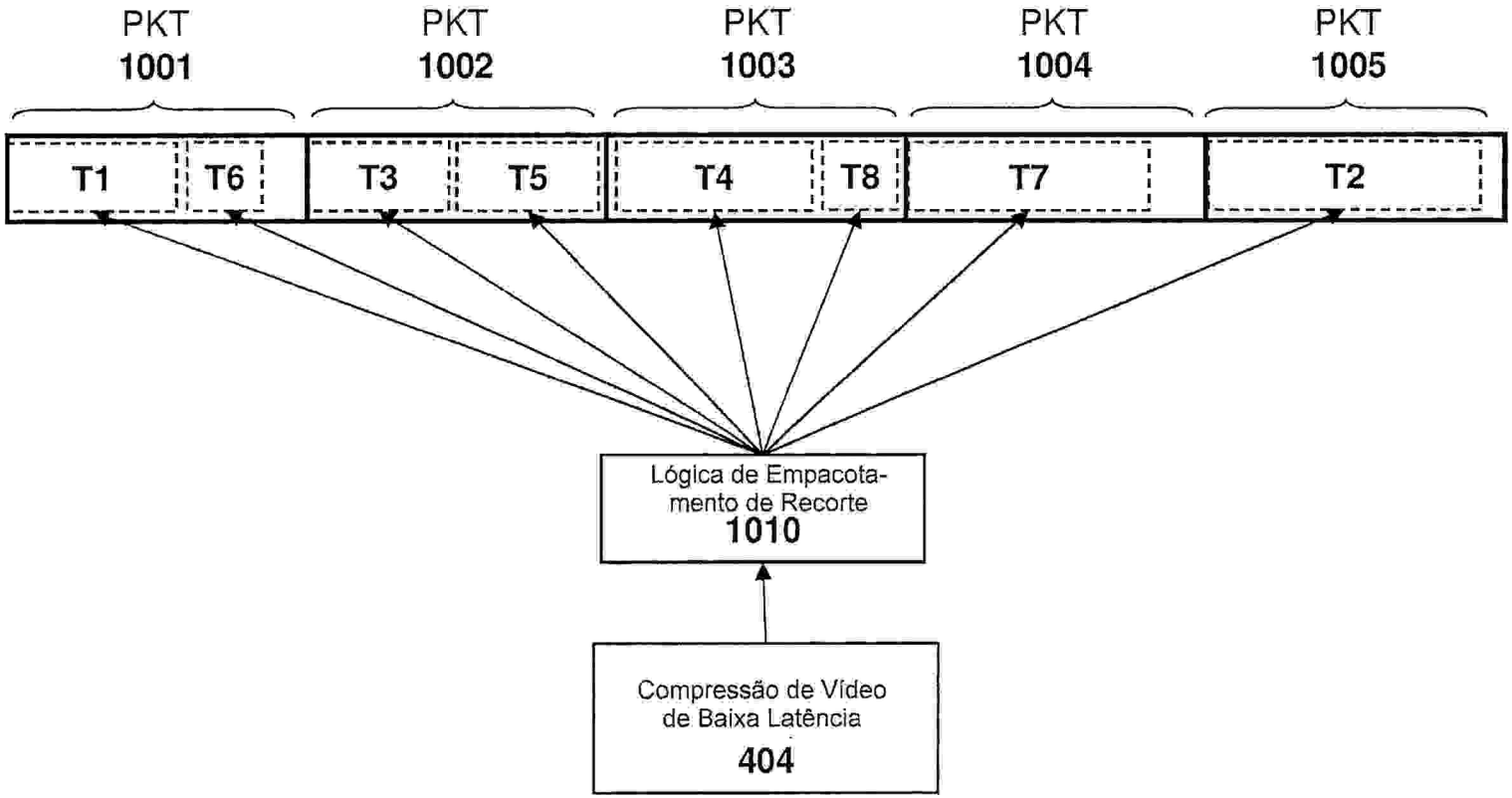

PKT
1001
PKT
1002
PKT
1003
PKT
1004
PKT
1005
T1
T6
T3
T5
T4
T8
T7
T2
Lógica de Empacota-
mento de Recorte
1010
Compressão de Vídeo
de Baixa Latência
404

Fig. 10b

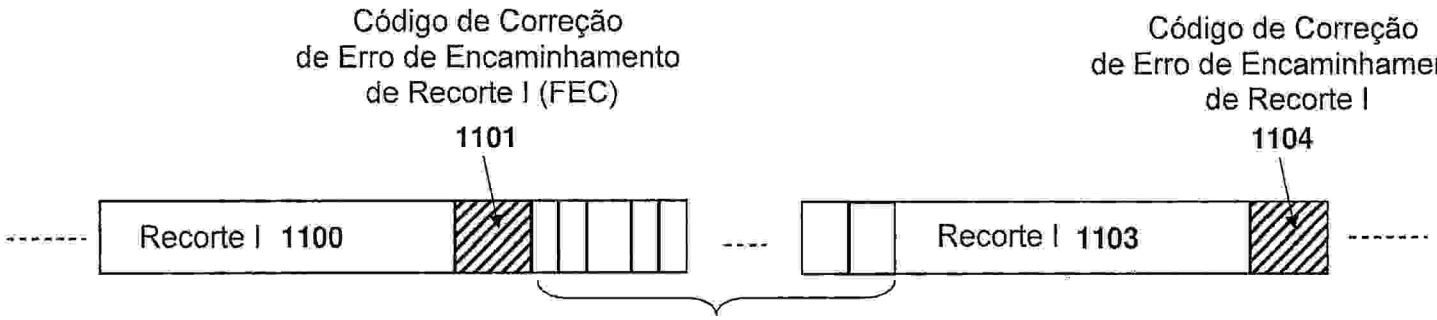


Fig. 11a

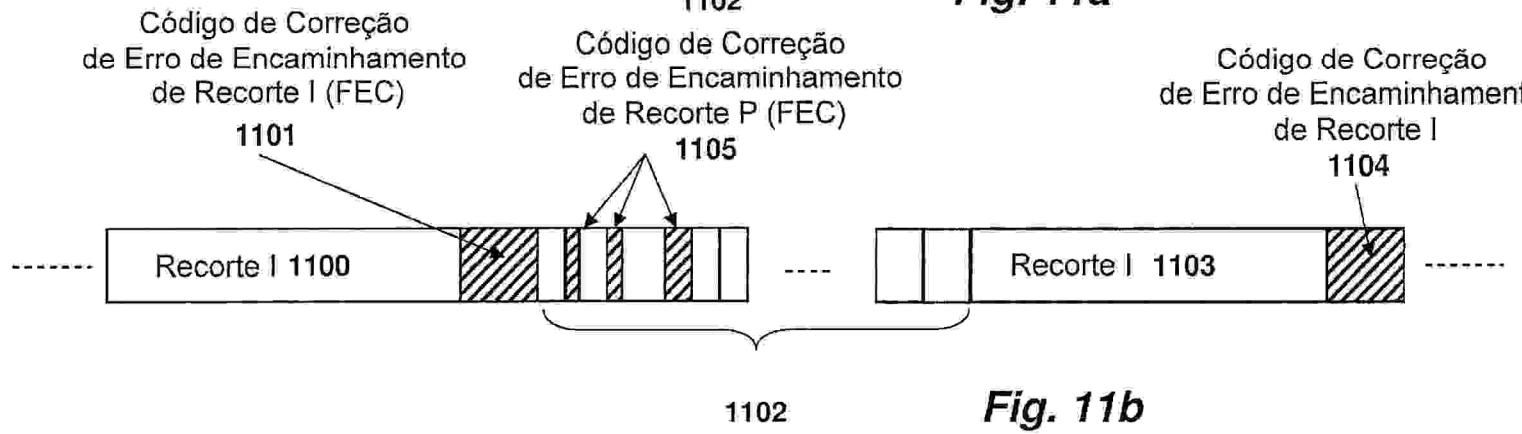


Fig. 11b

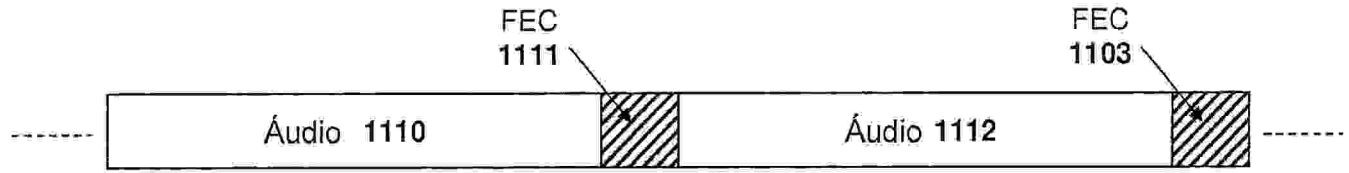


Fig. 11c

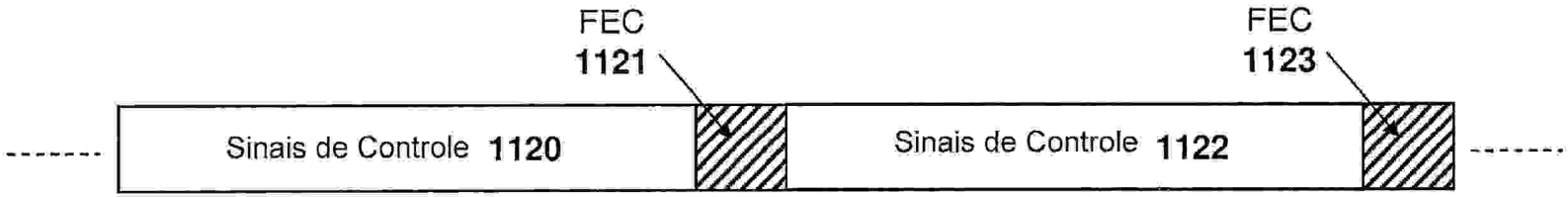

FEC
1121
FEC
1123
Sinais de Controle 1120
Sinais de Controle 1122

Fig. 11d

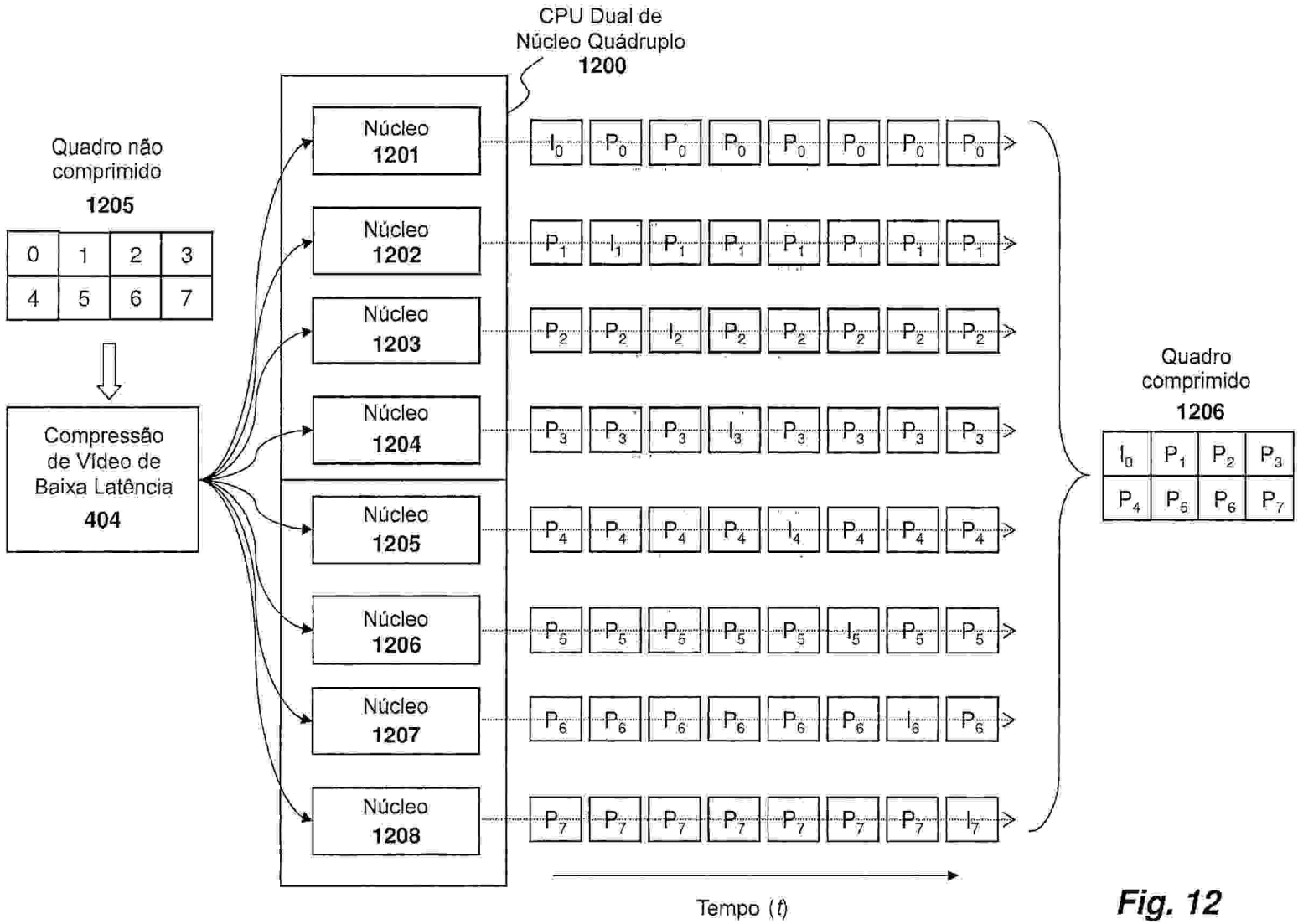

CPU Dual de
Núcleo Quádruplo
1200
Quadro não
comprimido
1205
0
1
2
3
4
5
6
7
Compressão
de Vídeo de
Baixa Latência
404
Núcleo
1201
Núcleo
1202
Núcleo
1203
Núcleo
1204
Núcleo
1205
Núcleo
1206
Núcleo
1207
Núcleo
1208
Quadro
comprimido
1206
Tempo (t)

Fig. 12

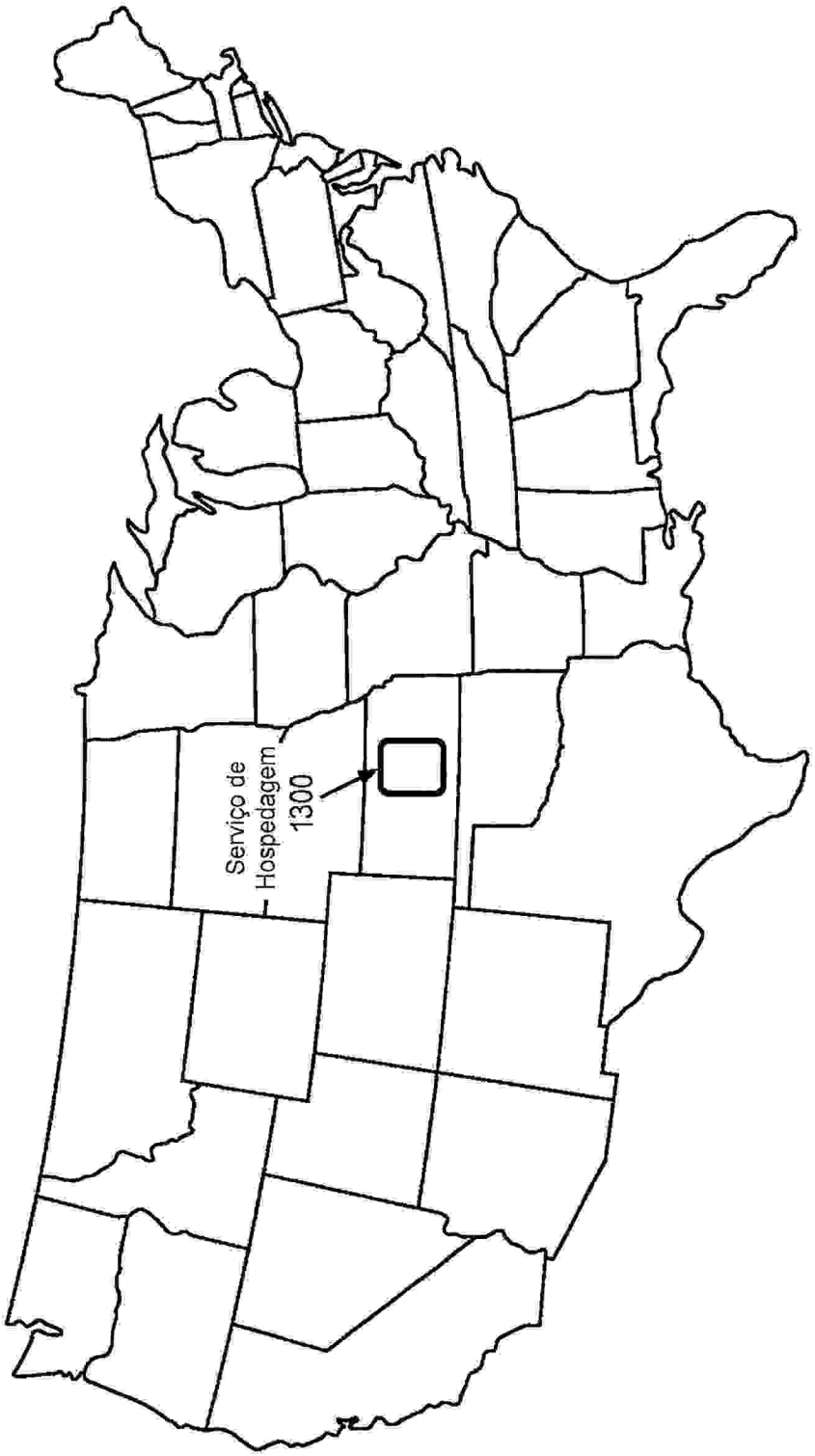


FIG. 13a

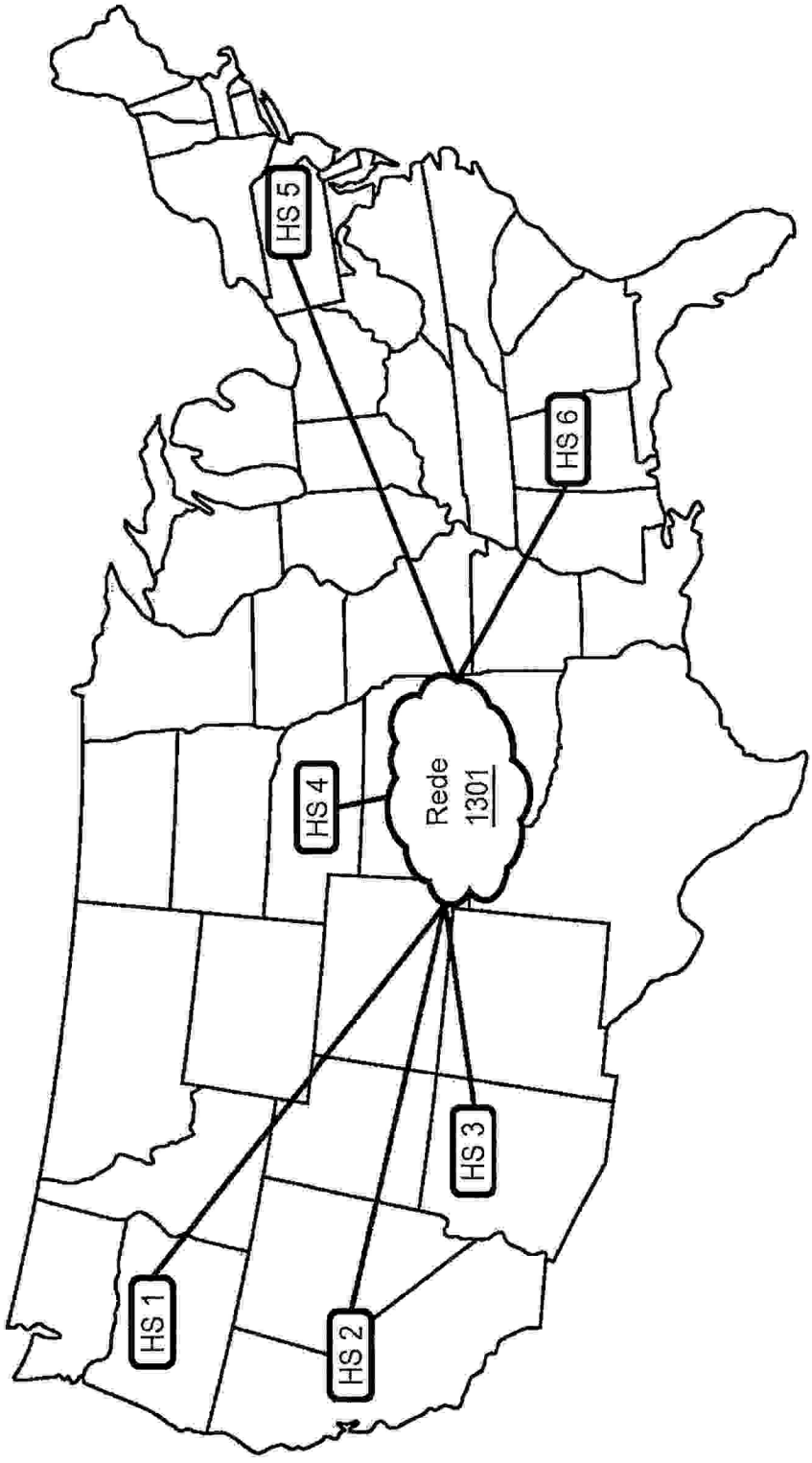


FIG. 13b

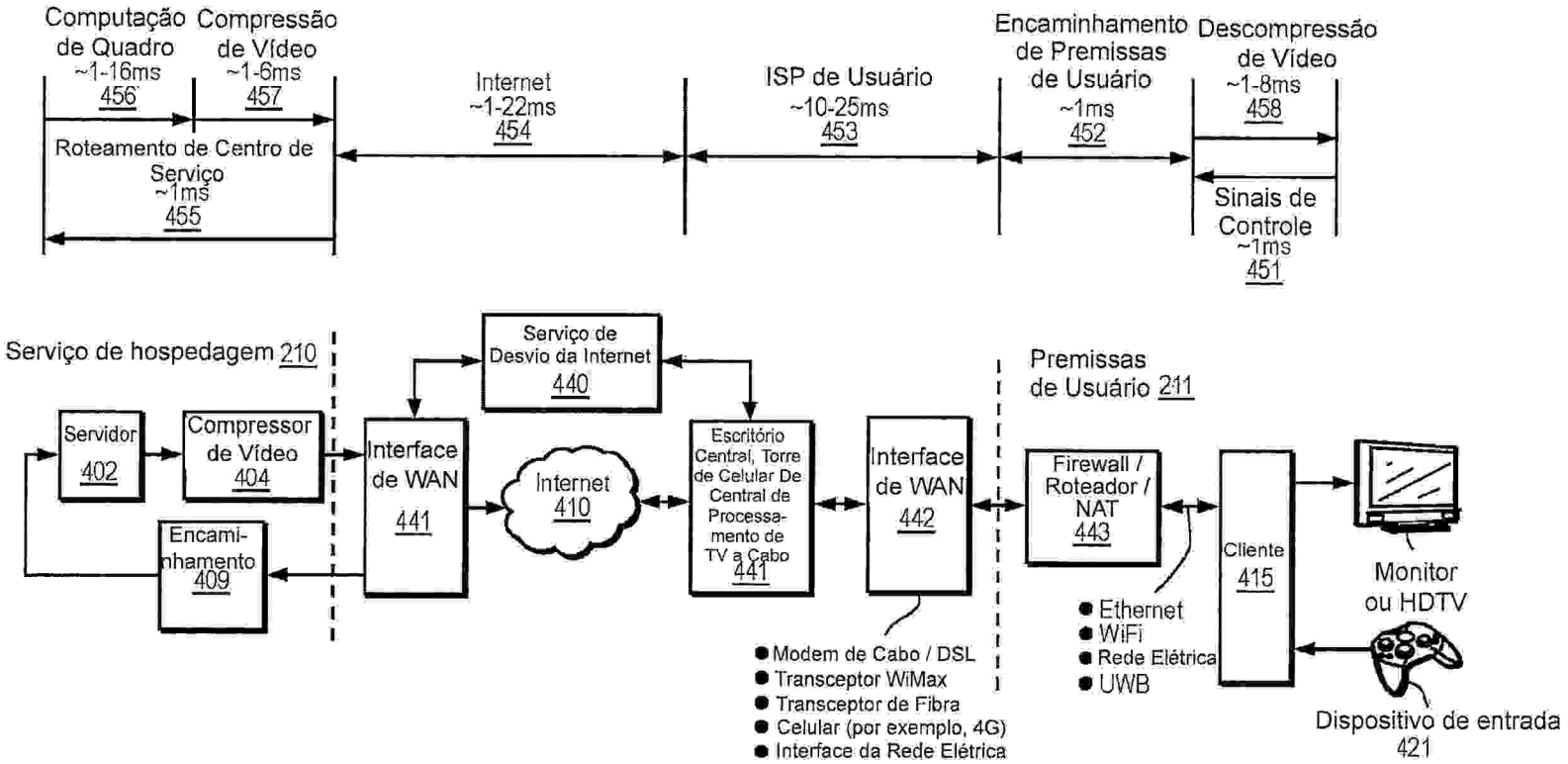

Computação de Quadro ~1-16ms 456
Compressão de Vídeo ~1-6ms 457
Roteamento de Centro de Serviço ~1ms 455
Internet ~1-22ms 454
ISP de Usuário ~10-25ms 453
Encaminhamento de Premissas de Usuário ~1ms 452
Descompressão de Vídeo ~1-8ms 458
Sinais de Controle ~1ms 451
Serviço de hospedagem 210
Servidor 402
Compressor de Vídeo 404
Encami-nhamento 409
Interface de WAN 441
Serviço de Desvio da Internet 440
Internet 410
Escritório Central, Torre de Celular De Central de Processa-mento de TV a Cabo 441
Interface de WAN 442
Premissas de Usuário 211
Firewall / Roteador / NAT 443
Cliente 415
Monitor ou HDTV
Ethernet
WiFi
Rede Elétrica
UWB
Modem de Cabo / DSL
Transceptor WiMax
Transceptor de Fibra
Celular (por exemplo, 4G)
Interface da Rede Elétrica
Dispositivo de entrada 421

FIG. 14

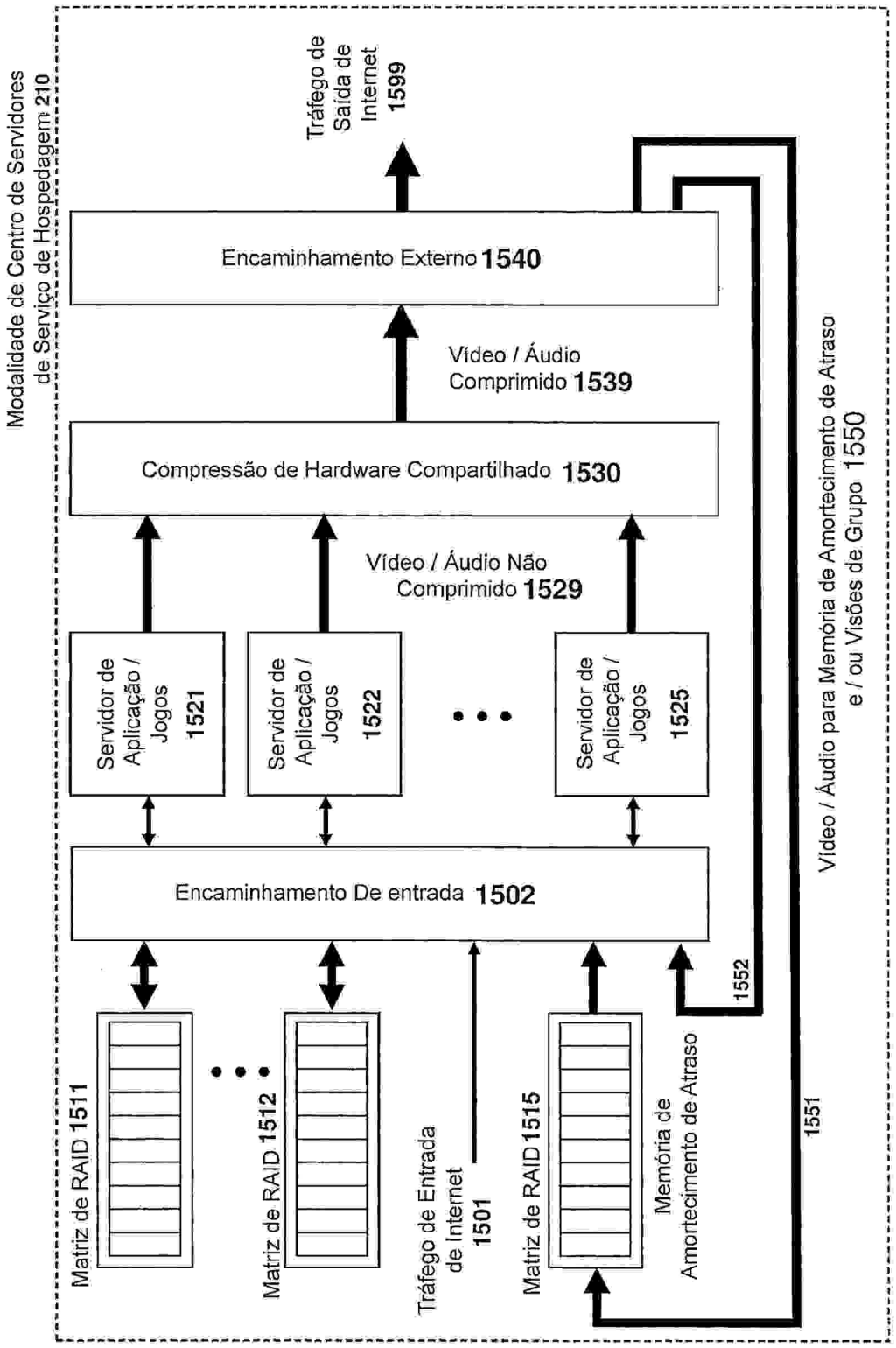

Modalidade de Centro de Servidores de Serviço de Hospedagem 210
Tráfego de Saída de Internet 1599
Encaminhamento Externo 1540
Vídeo / Áudio Comprimido 1539
Compressão de Hardware Compartilhado 1530
Vídeo / Áudio Não Comprimido 1529
Servidor de Aplicação / Jogos 1521
Servidor de Aplicação / Jogos 1522
Servidor de Aplicação / Jogos 1525
Encaminhamento De entrada 1502
Matriz de RAID 1511
Matriz de RAID 1512
Tráfego de Entrada de Internet 1501
Matriz de RAID 1515
Memória de Amortecimento de Atraso
1552
1551
Vídeo / Áudio para Memória de Amortecimento de Atraso e / ou Visões de Grupo 1550

Fig. 15

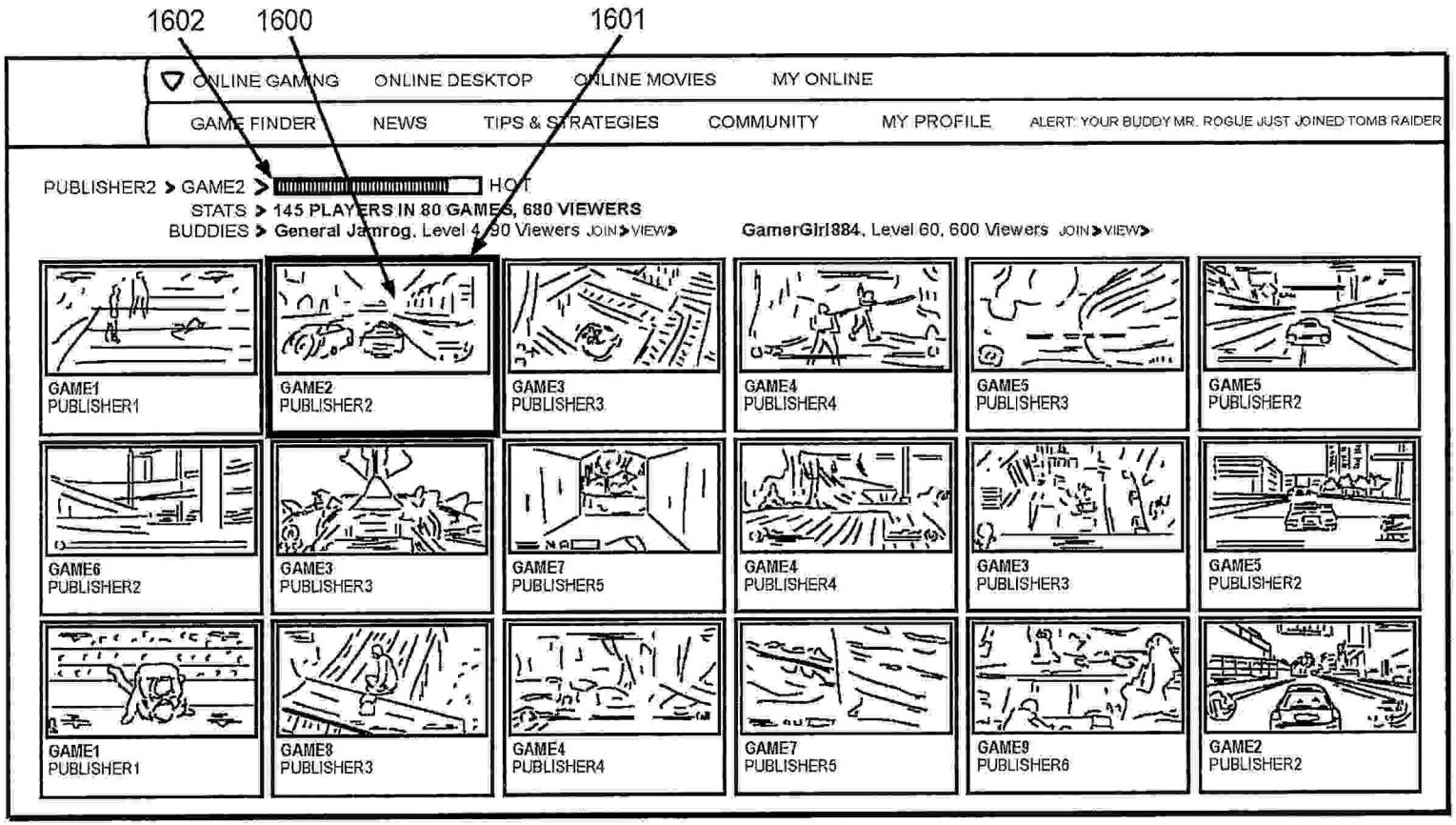

1602
1600
1601
ONLINE GAMING
ONLINE DESKTOP
ONLINE MOVIES
MY ONLINE
GAME FINDER
NEWS
TIPS & STRATEGIES
COMMUNITY
MY PROFILE
ALERT: YOUR BUDDY MR. ROGUE JUST JOINED TOMB RAIDER
PUBLISHER2 > GAME2 > HOT
STATS > 145 PLAYERS IN 80 GAMES, 680 VIEWERS
BUDDIES > General Jamrog, Level 4, 90 Viewers JOIN>VIEW>
GamerGirl884, Level 60, 600 Viewers JOIN>VIEW>
GAME1 PUBLISHER1
GAME2 PUBLISHER2
GAME3 PUBLISHER3
GAME4 PUBLISHER4
GAME5 PUBLISHER3
GAME5 PUBLISHER2
GAME6 PUBLISHER2
GAME3 PUBLISHER3
GAME7 PUBLISHER5
GAME4 PUBLISHER4
GAME3 PUBLISHER3
GAME5 PUBLISHER2
GAME1 PUBLISHER1
GAME8 PUBLISHER3
GAME4 PUBLISHER4
GAME7 PUBLISHER5
GAME9 PUBLISHER6
GAME2 PUBLISHER2

FIG. 16

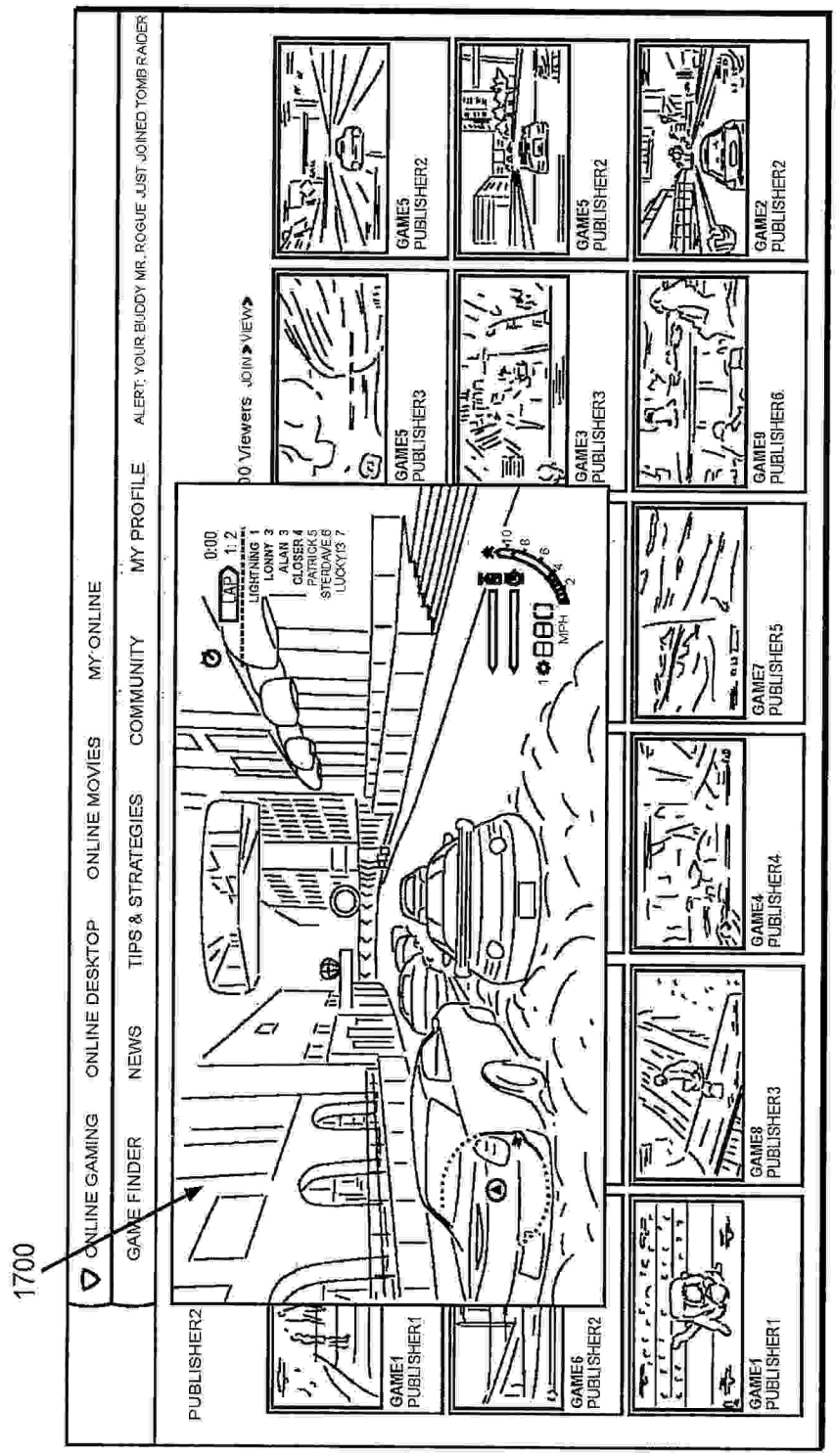

1700
ONLINE GAMING
ONLINE DESKTOP
ONLINE MOVIES
MY ONLINE
GAME FINDER
NEWS
TIPS & STRATEGIES
COMMUNITY
MY PROFILE
ALERT: YOUR BUDDY MR. ROGUE JUST JOINED TOMB RAIDER
PUBLISHER2
0 Viewers JOIN VIEW
LAP 0:00
1: 2
LIGHTNING 1
LONNY 3
ALAN 3
CLOSER 4
PATRICK 5
STERDAVE 6
LUCKY13 7
880
MPH
GAME1
PUBLISHER1
GAME8
PUBLISHER3
GAME4
PUBLISHER4
GAME7
PUBLISHER5
GAME9
PUBLISHER6
GAME2
PUBLISHER2
GAME6
PUBLISHER2
GAME1
PUBLISHER1
GAME3
PUBLISHER3
GAME5
PUBLISHER2
GAME5
PUBLISHER3

FIG. 17

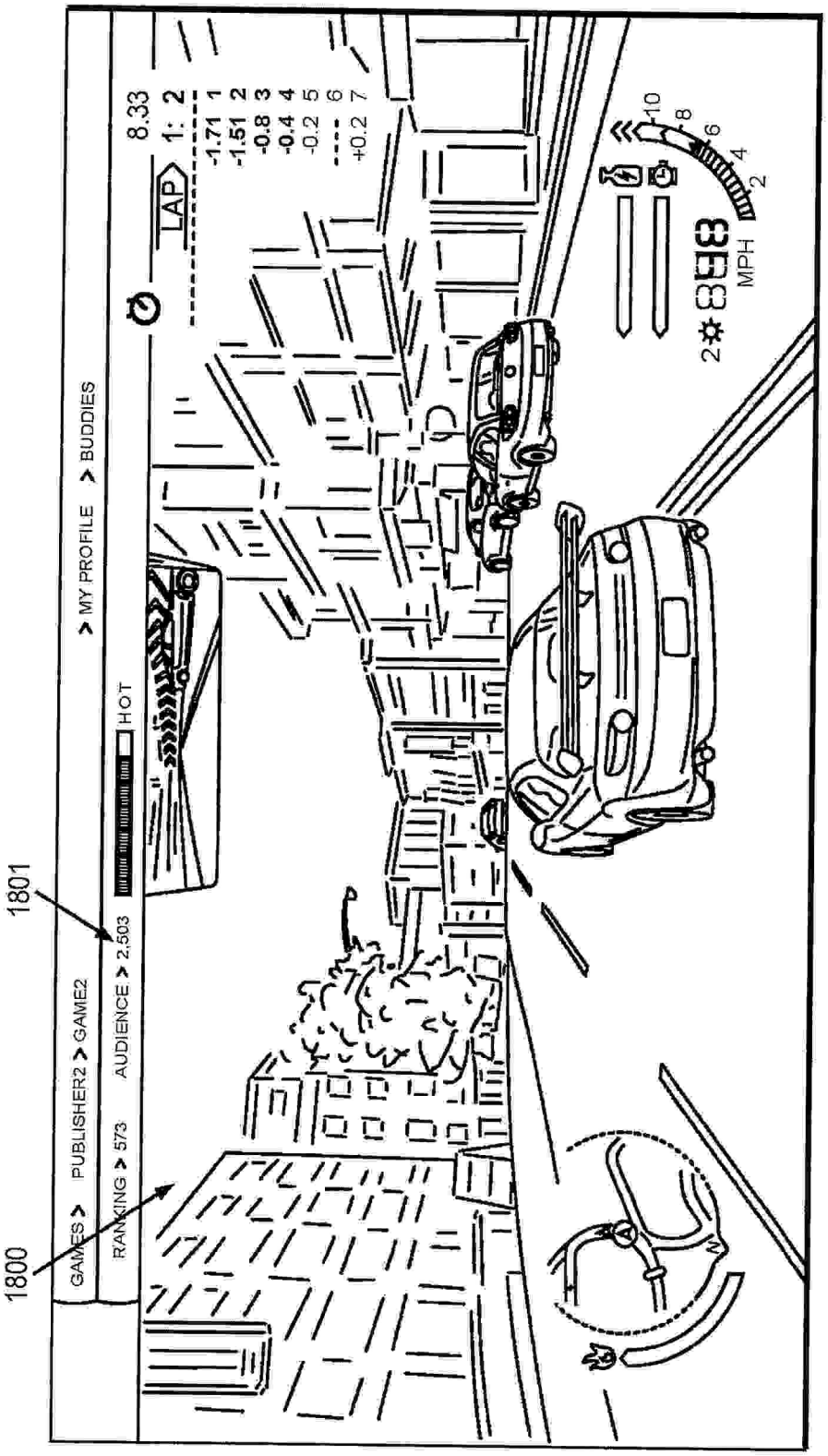

1800
1801
GAMES > PUBLISHER2 > GAME2
> MY PROFILE > BUDDIES
RANKING > 573
AUDIENCE > 2,503
HOT
LAP
8.33
1: 2
-1.71 1
-1.51 2
-0.8 3
-0.4 4
-0.2 5
6
+0.2 7
MPH

FIG. 18

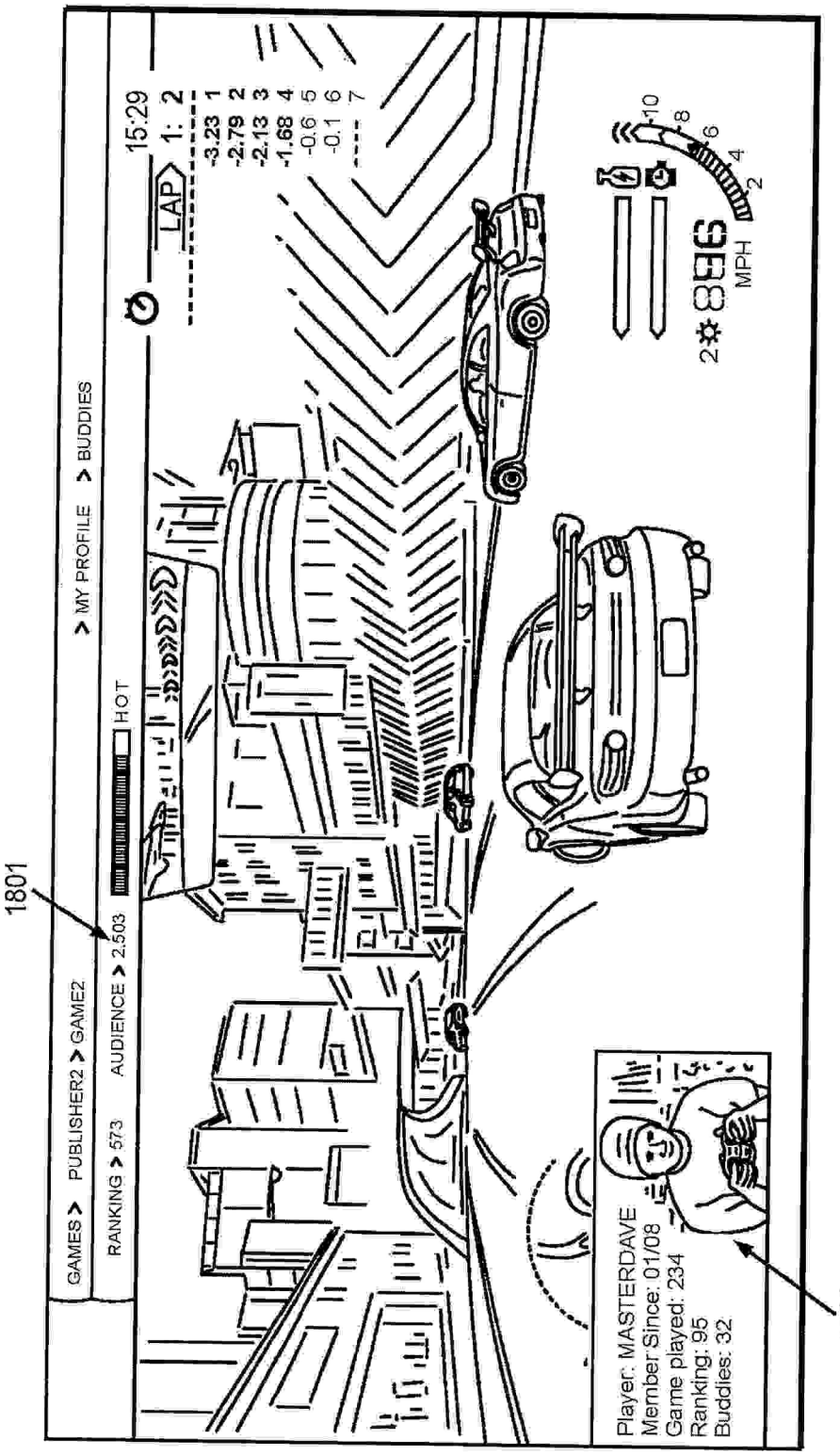

GAMES > PUBLISHER2 > GAME2
> MY PROFILE > BUDDIES
RANKING > 573 AUDIENCE > 2.503 HOT
15:29
LAP 1: 2
-3.23 1
-2.79 2
-2.13 3
-1.68 4
-0.6 5
-0.1 6
---- 7
MPH
Player: MASTERDAVE
Member Since: 01/08
Game played: 234
Ranking: 95
Buddies: 32
1801
1900

FIG. 19

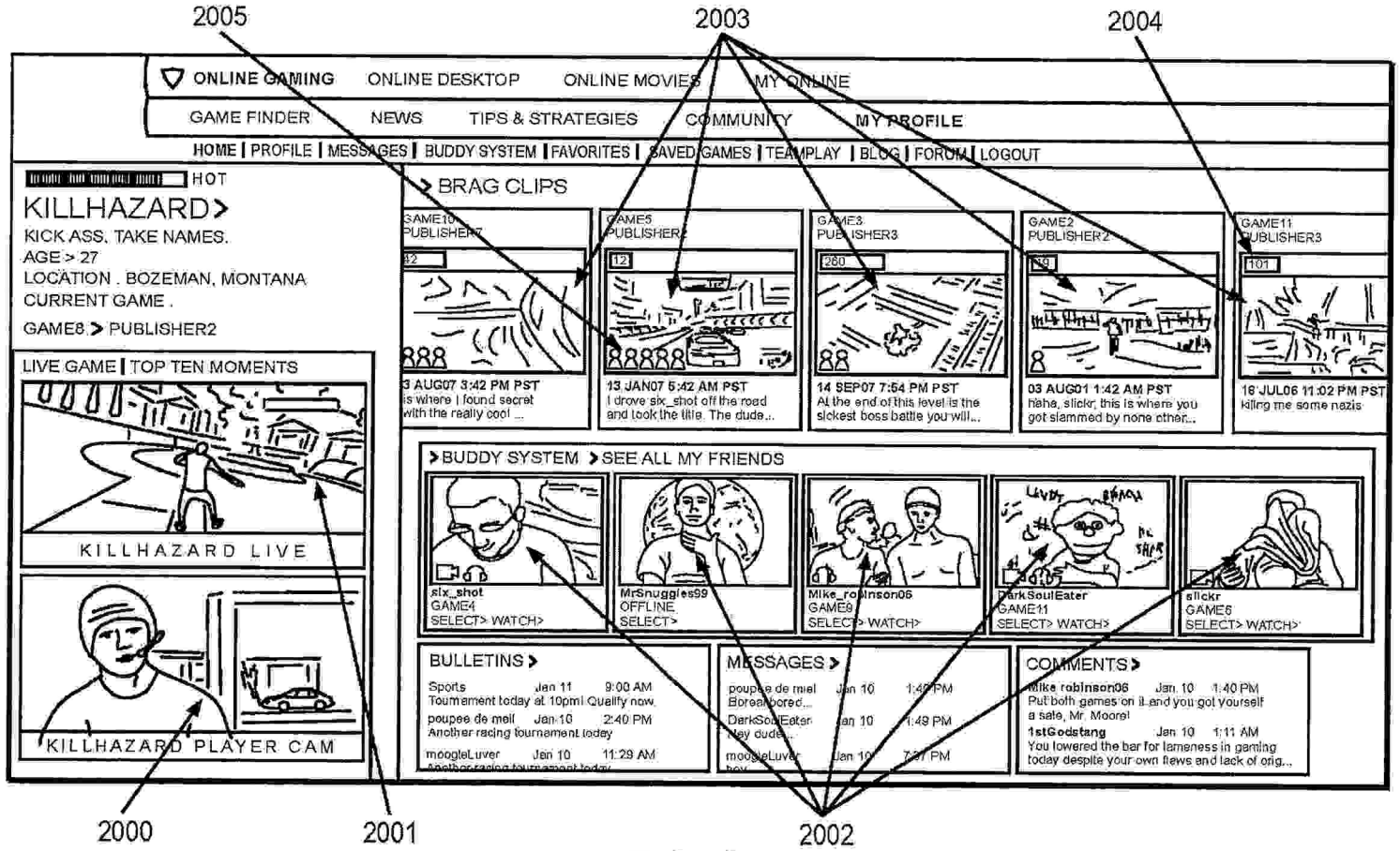

2005
2003
2004
ONLINE GAMING
ONLINE DESKTOP
ONLINE MOVIES
MY ONLINE
GAME FINDER
NEWS
TIPS & STRATEGIES
COMMUNITY
MY PROFILE
HOME | PROFILE | MESSAGES | BUDDY SYSTEM | FAVORITES | SAVED GAMES | TEAMPLAY | BLOG | FORUM | LOGOUT
HOT
KILLHAZARD>
KICK ASS, TAKE NAMES.
AGE > 27
LOCATION . BOZEMAN, MONTANA
CURRENT GAME .
GAME8 > PUBLISHER2
LIVE GAME | TOP TEN MOMENTS
KILLHAZARD LIVE
KILLHAZARD PLAYER CAM
> BRAG CLIPS
GAME10
PUBLISHER1
GAME5
PUBLISHER2
GAME3
PUBLISHER3
GAME2
PUBLISHER2
GAME11
PUBLISHER3
13 JAN07 5:42 AM PST
I drove six_shot off the road and took the title. The dude...
14 SEP07 7:54 PM PST
At the end of this level is the sickest boss battle you will...
03 AUG01 1:42 AM PST
haha, slickr, this is where you got slammed by none other...
16 JUL06 11:02 PM PST
killing me some nazis
> BUDDY SYSTEM > SEE ALL MY FRIENDS
six_shot
GAME4
SELECT> WATCH>
MrSnuggles99
OFFLINE
SELECT>
Mike_robinson06
GAME8
SELECT> WATCH>
DarkSoulEater
GAME11
SELECT> WATCH>
slickr
GAME6
SELECT> WATCH>
BULLETINS >
Sports Jan 11 9:00 AM
Tournament today at 10pm! Qualify now.
poupee de miel Jan 10 2:40 PM
Another racing tournament today
MESSAGES >
poupee de miel Jan 10 1:49 PM
COMMENTS >
Mike robinson06 Jan 10 1:40 PM
Put both games on it and you got yourself a sale, Mr. Moore!
1stGodstang Jan 10 1:11 AM
You lowered the bar for lameness in gaming today despite your own flews and lack of orig...
2000
2001
2002

FIG. 20

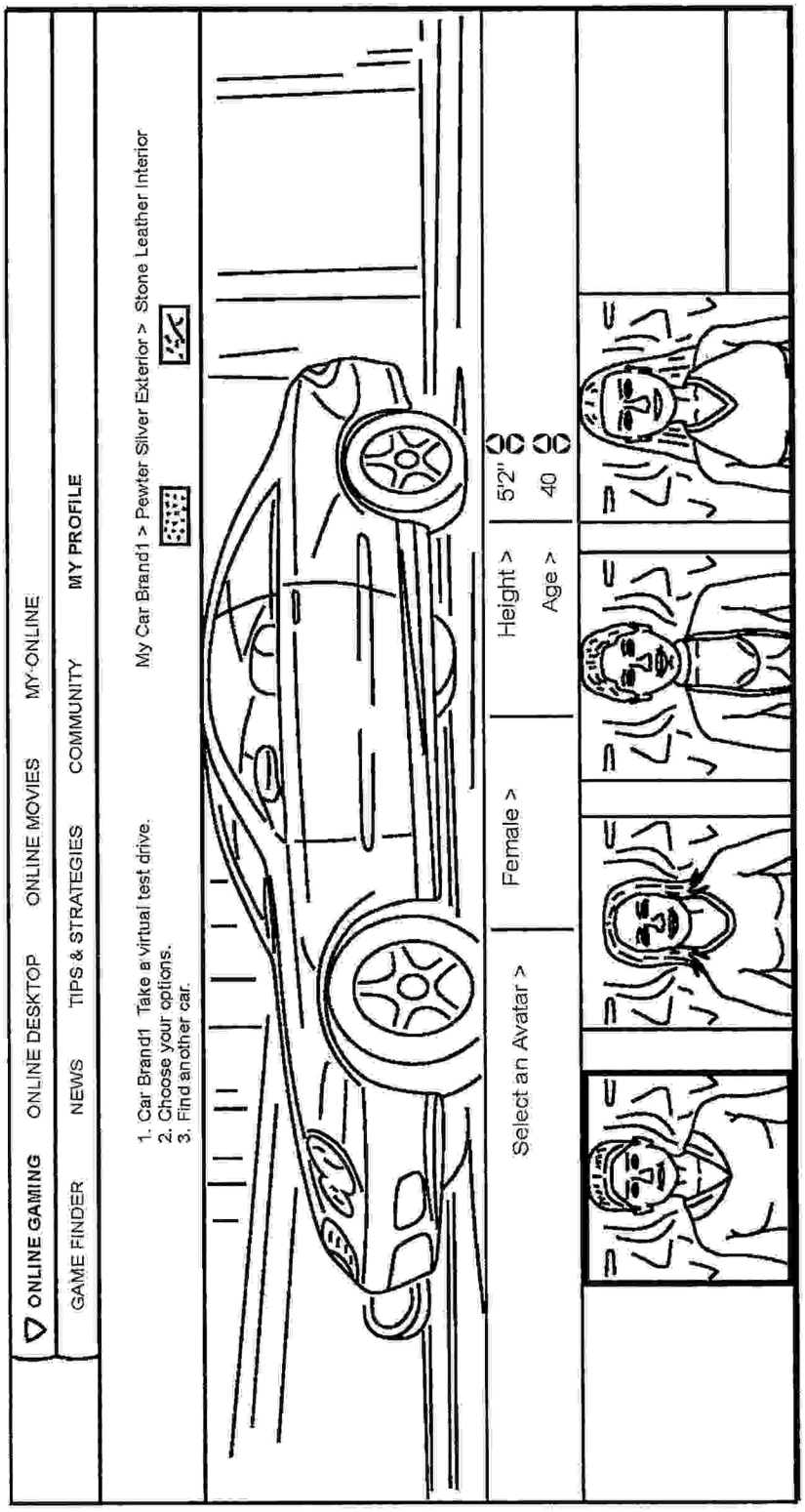

ONLINE GAMING
ONLINE DESKTOP
ONLINE MOVIES
MY-ONLINE
GAME FINDER
NEWS
TIPS & STRATEGIES
COMMUNITY
MY PROFILE
1. Car Brand1 Take a virtual test drive.
2. Choose your options.
3. Find another car.
My Car Brand1 > Pewter Silver Exterior > Stone Leather Interior
Select an Avatar >
Female >
Height >
5'2"
Age >
40

FIG. 21

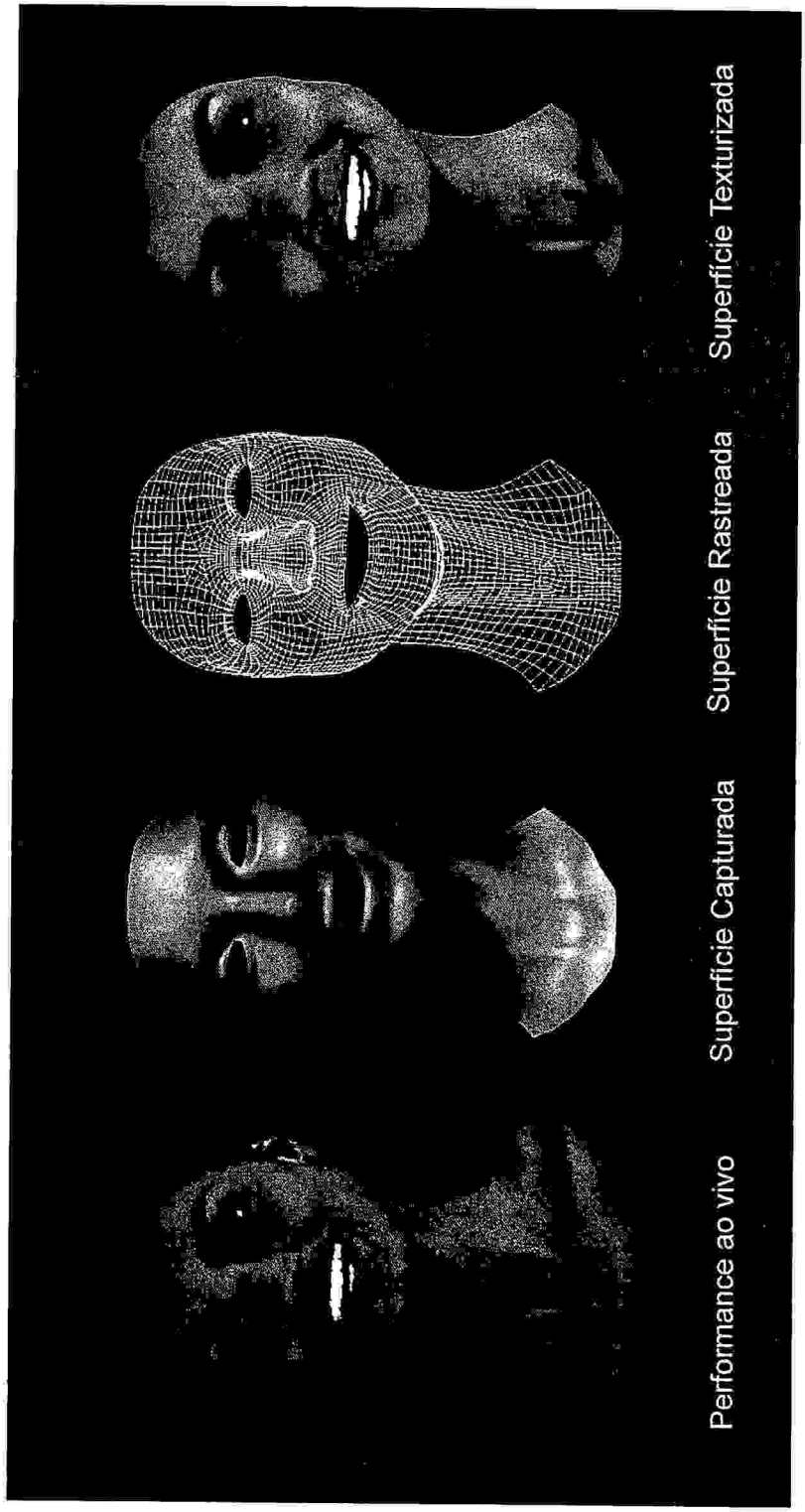

Performance ao vivo
Superfície Capturada
Superfície Rastreada
Superfície Texturizada

Fig. 22

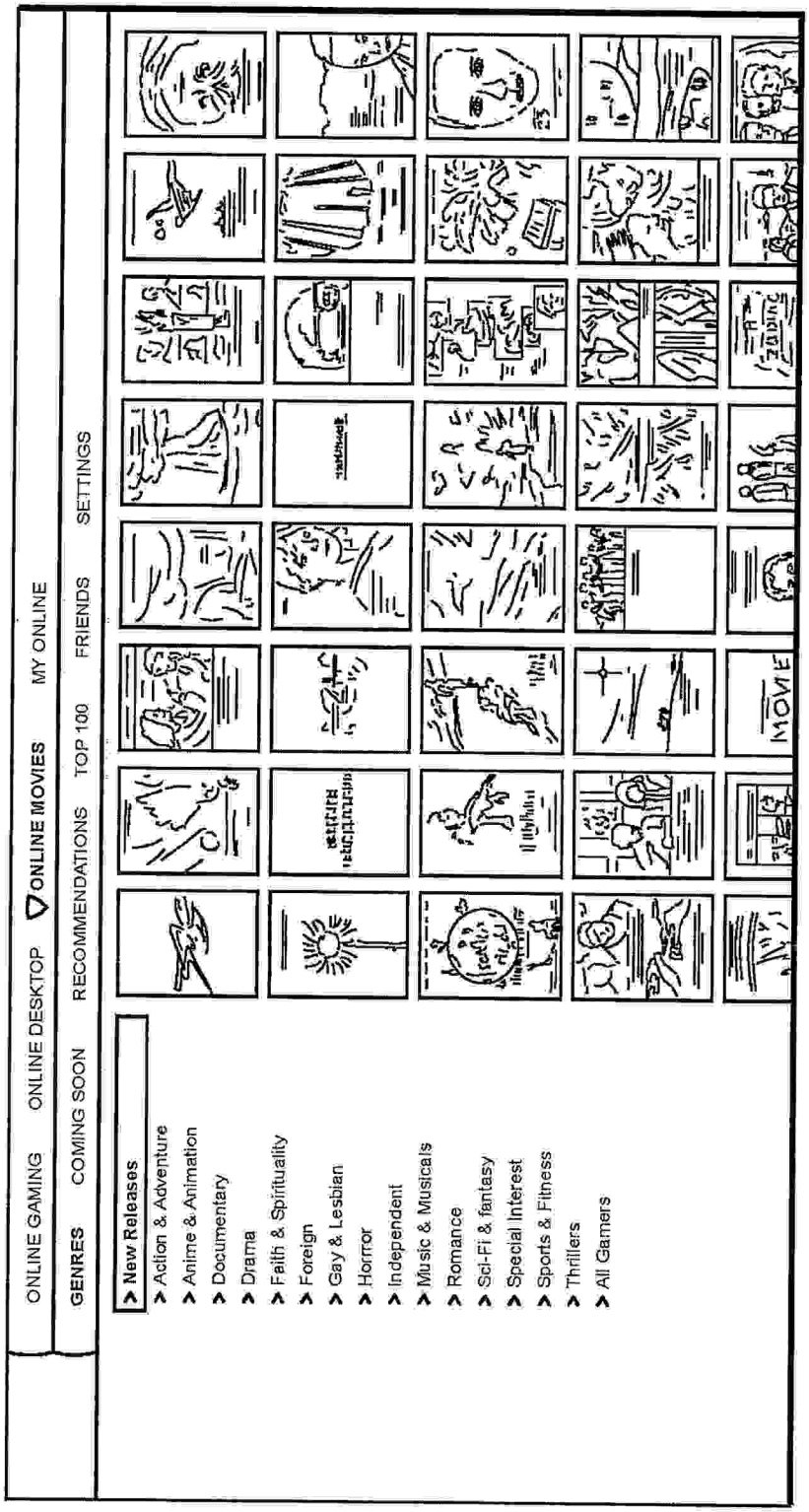

ONLINE GAMING
ONLINE DESKTOP
ONLINE MOVIES
MY ONLINE
GENRES
COMING SOON
RECOMMENDATIONS
TOP 100
FRIENDS
SETTINGS
> New Releases
> Action & Adventure
> Anime & Animation
> Documentary
> Drama
> Faith & Spirituality
> Foreign
> Gay & Lesbian
> Horror
> Independent
> Music & Musicals
> Romance
> Sci-Fi & fantasy
> Special Interest
> Sports & Fitness
> Thrillers
> All Gamers

FIG. 23

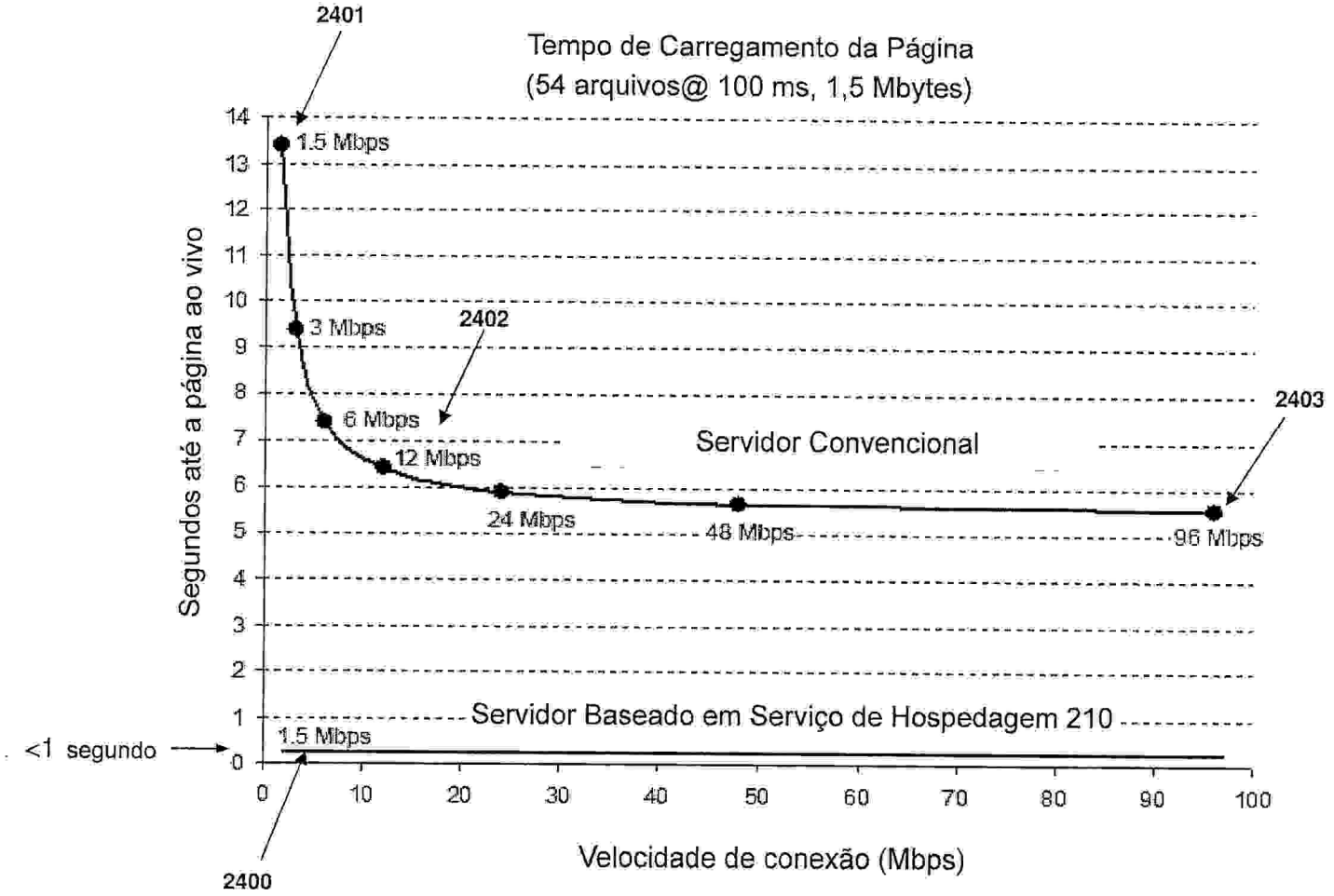

2401
Tempo de Carregamento da Página
(54 arquivos@ 100 ms, 1,5 Mbytes)
Segundos até a página ao vivo
1.5 Mbps
3 Mbps
2402
6 Mbps
12 Mbps
24 Mbps
48 Mbps
96 Mbps
2403
Servidor Convencional
Servidor Baseado em Serviço de Hospedagem 210
1.5 Mbps
<1 segundo
2400
Velocidade de conexão (Mbps)
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
0
10
20
30
40
50
60
70
80
90
100

Fig. 24

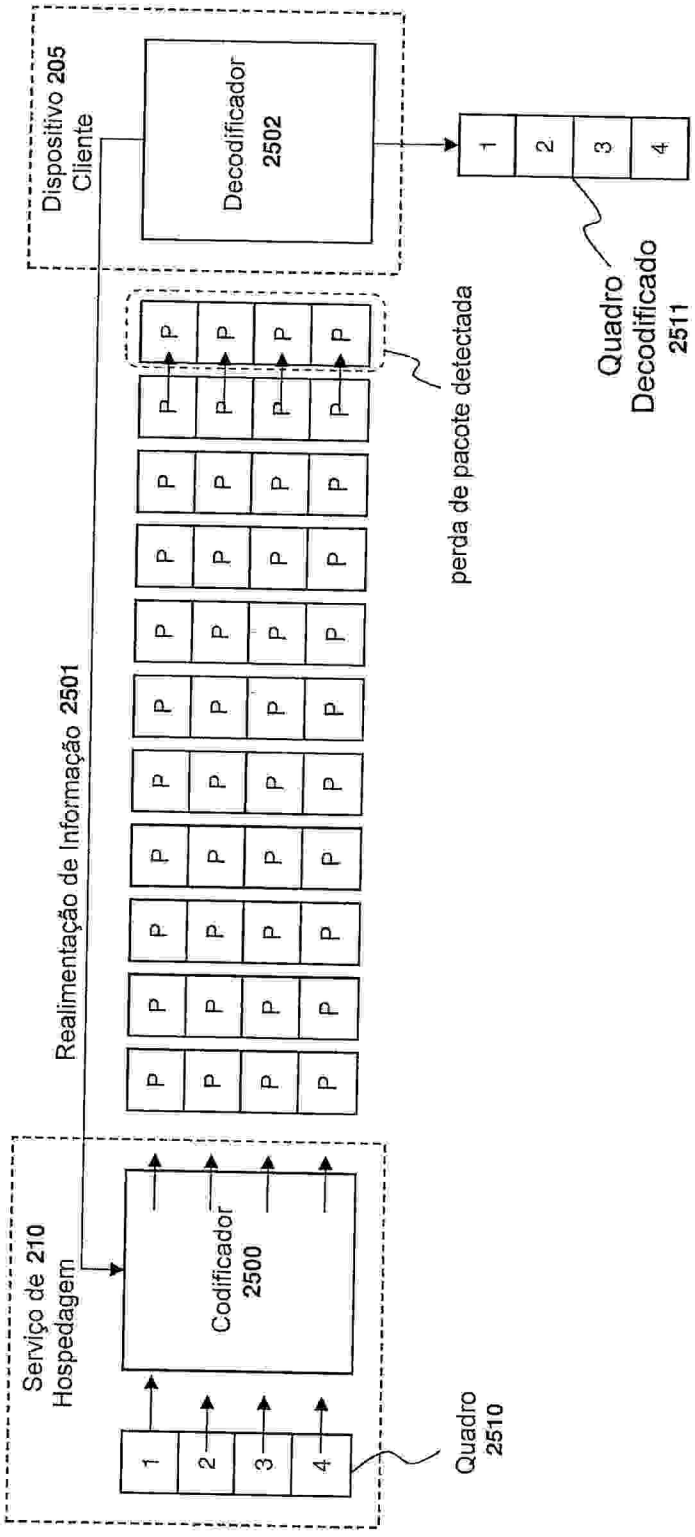

Serviço de 210
Hospedagem
Codificador
2500
Quadro
2510
Realimentação de Informação 2501
Dispositivo 205
Cliente
Decodificador
2502
perda de pacote detectada
Quadro
Decodificado
2511
1
2
3
4
P

Fig. 25a

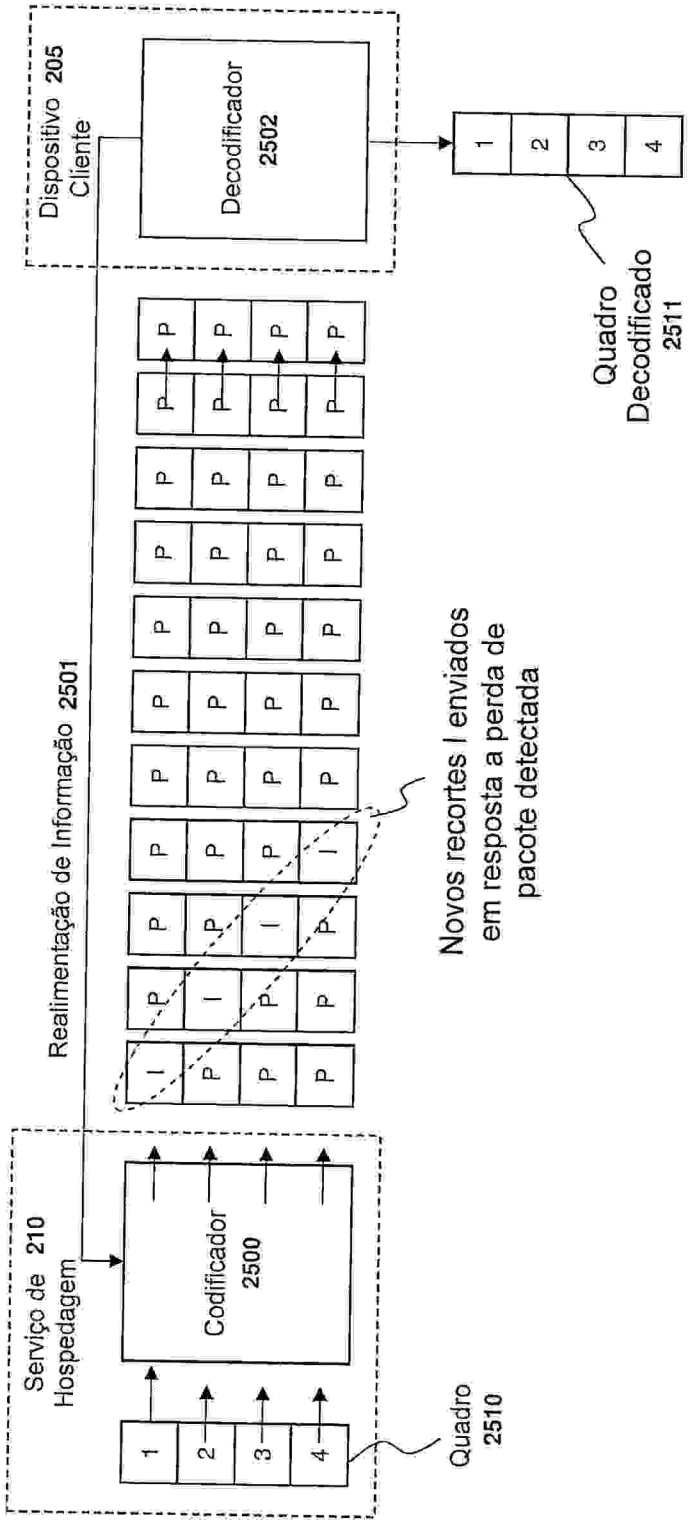

Serviço de Hospedagem 210
Codificador 2500
Quadro 2510
Realimentação de Informação 2501
Novos recortes I enviados em resposta a perda de pacote detectada
Dispositivo Cliente 205
Decodificador 2502
Quadro Decodificado 2511

Fig. 25b

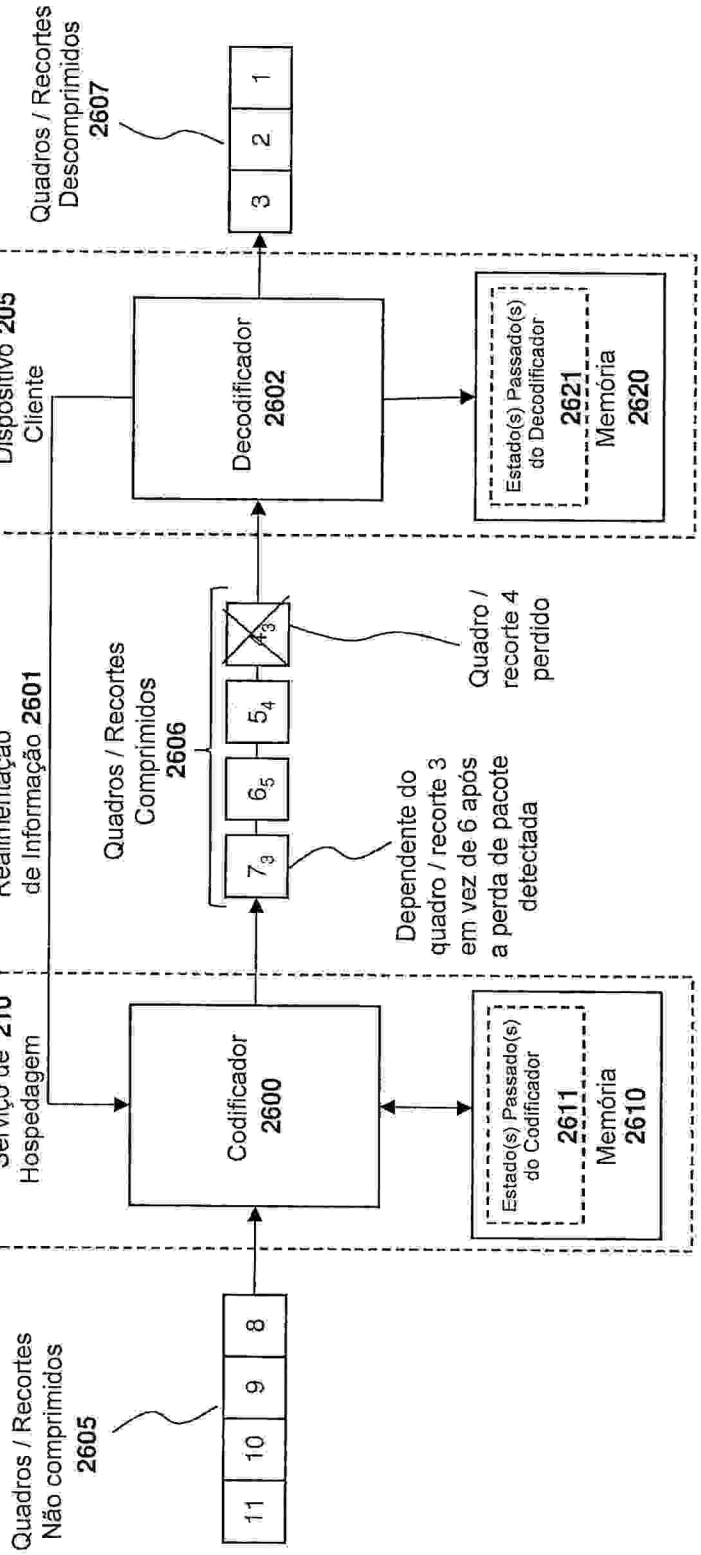

Quadros / Recortes Não comprimidos 2605
11
10
9
8
Serviço de 210 Hospedagem
Codificador 2600
Estado(s) Passado(s) do Codificador 2611
Memória 2610
Realimentação de Informação 2601
Quadros / Recortes Comprimidos 2606
7_3
6_5
5_4
4_3
Dependente do quadro / recorte 3 em vez de 6 após a perda de pacote detectada
Quadro / recorte 4 perdido
Dispositivo 205 Cliente
Decodificador 2602
Estado(s) Passado(s) do Decodificador 2621
Memória 2620
Quadros / Recortes Descomprimidos 2607
3
2
1

Fig. 26a

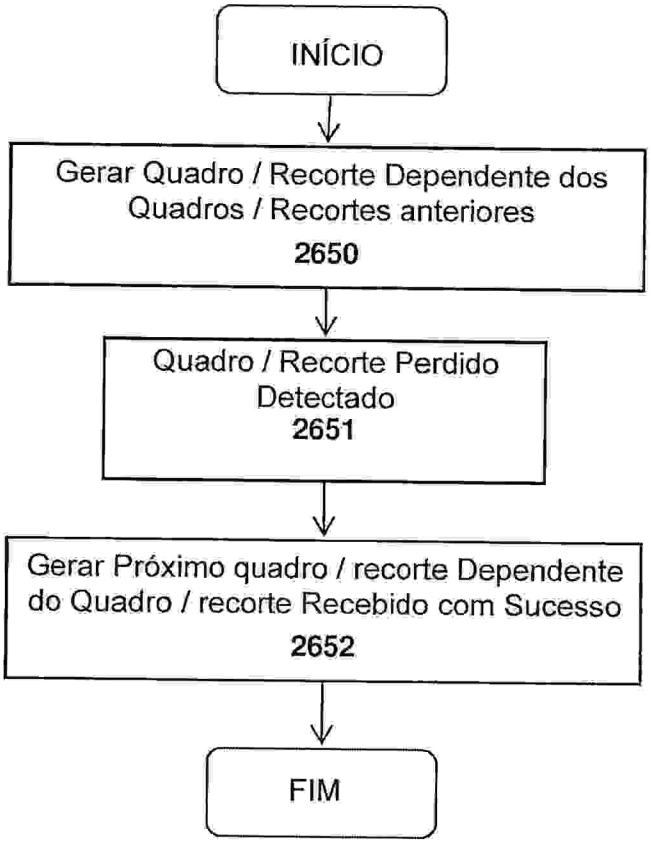

INÍCIO
Gerar Quadro / Recorte Dependente dos Quadros / Recortes anteriores
2650
Quadro / Recorte Perdido Detectado
2651
Gerar Próximo quadro / recorte Dependente do Quadro / recorte Recebido com Sucesso
2652
FIM

Fig. 26b

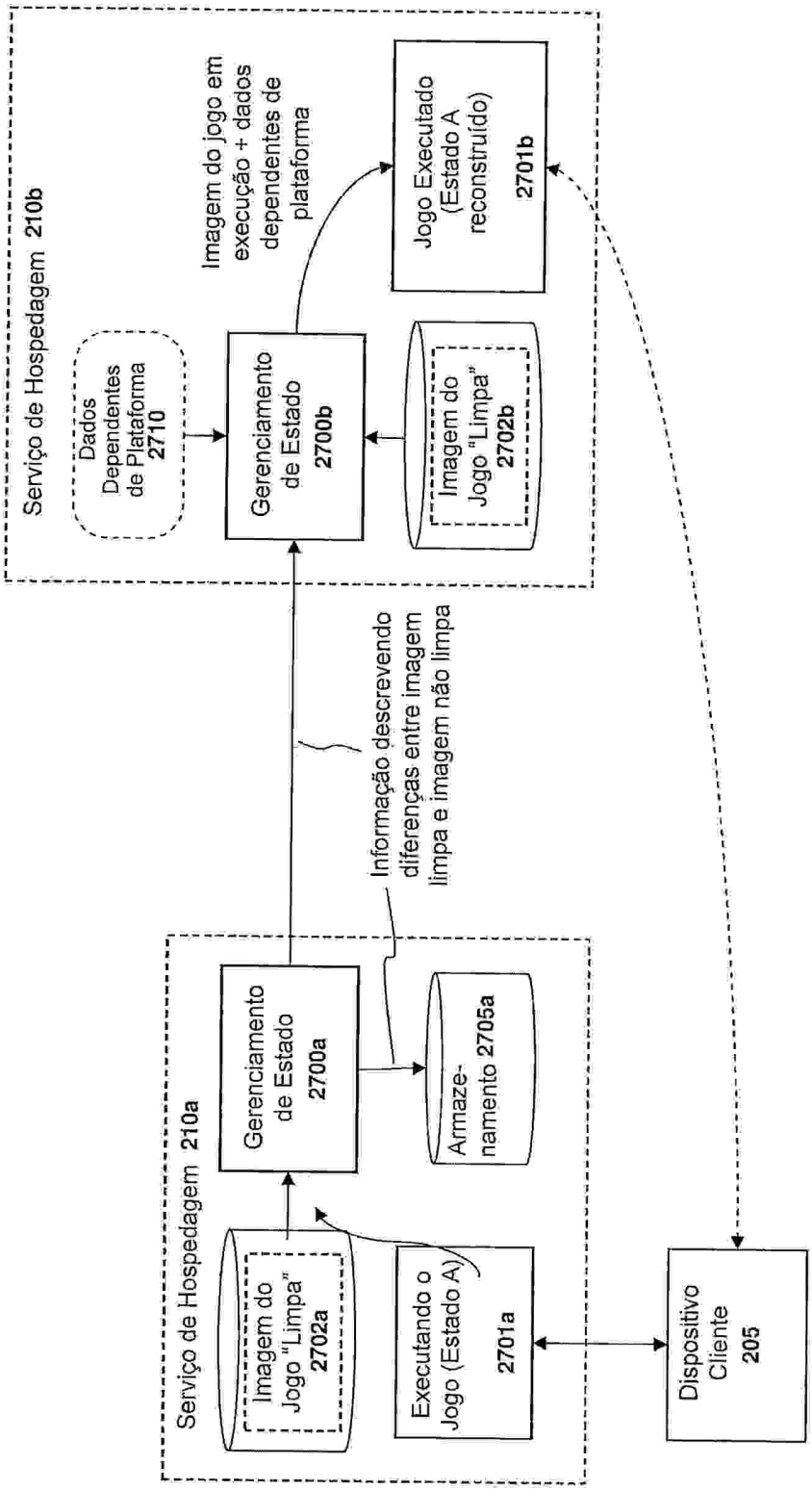

Serviço de Hospedagem 210b
Dados Dependentes de Plataforma 2710
Gerenciamento de Estado 2700b
Imagem do jogo em execução + dados dependentes de plataforma
Jogo Executado (Estado A reconstruído) 2701b
Imagem do Jogo "Limpa" 2702b
Informação descrevendo diferenças entre imagem limpa e imagem não limpa
Serviço de Hospedagem 210a
Gerenciamento de Estado 2700a
Armaze-namento 2705a
Imagem do Jogo "Limpa" 2702a
Executando o Jogo (Estado A) 2701a
Dispositivo Cliente 205

Fig. 27a

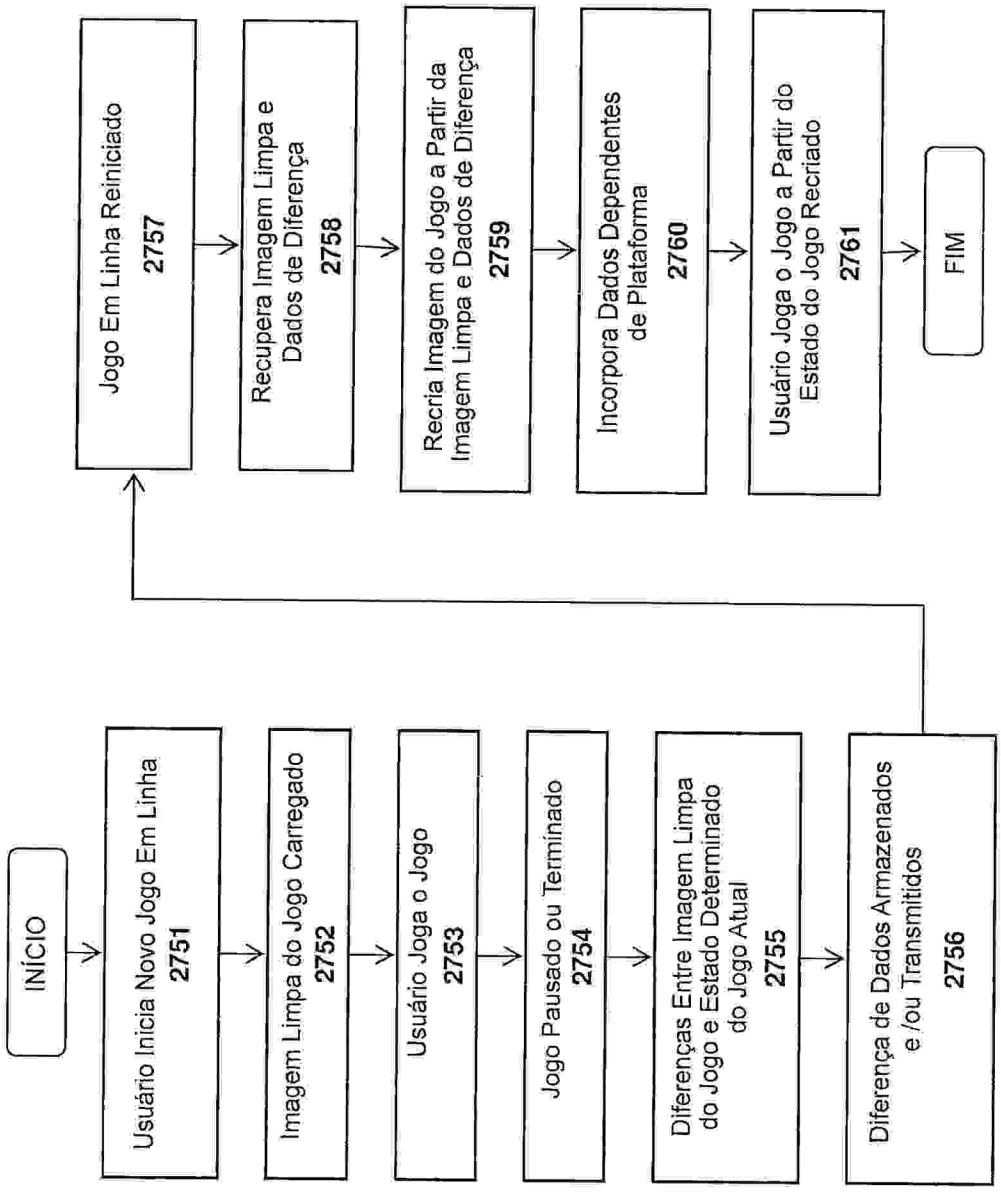

INÍCIO
Usuário Inicia Novo Jogo Em Linha
2751
Imagem Limpa do Jogo Carregado
2752
Usuário Joga o Jogo
2753
Jogo Pausado ou Terminado
2754
Diferenças Entre Imagem Limpa do Jogo e Estado Determinado do Jogo Atual
2755
Diferença de Dados Armazenados e /ou Transmitidos
2756
Jogo Em Linha Reiniciado
2757
Recupera Imagem Limpa e Dados de Diferença
2758
Recria Imagem do Jogo a Partir da Imagem Limpa e Dados de Diferença
2759
Incorpora Dados Dependentes de Plataforma
2760
Usuário Joga o Jogo a Partir do Estado do Jogo Recriado
2761
FIM

Fig. 27b

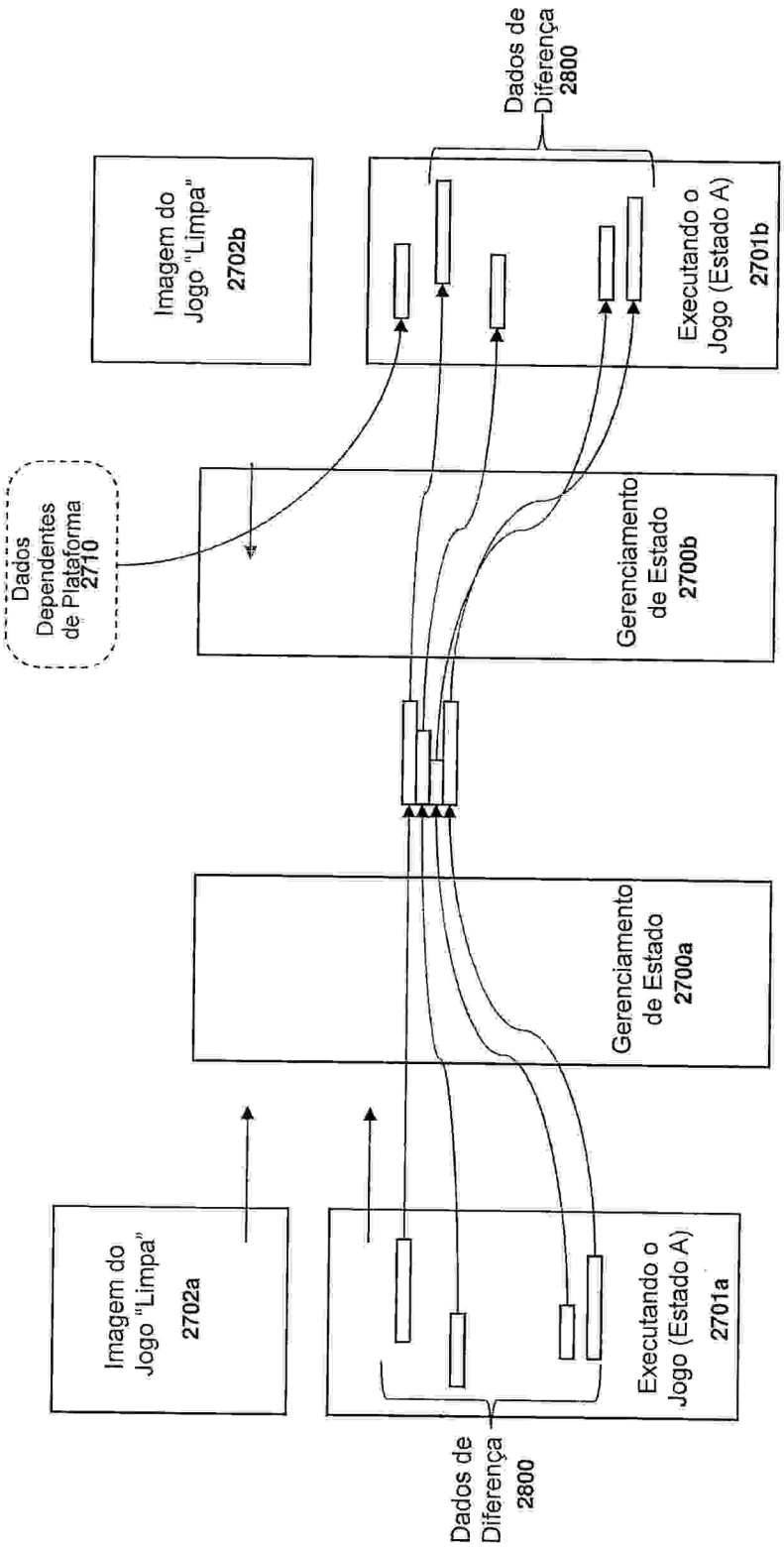


Imagem do Jogo "Limpa" 2702a
Executando o Jogo (Estado A) 2701a
Dados de Diferença 2800
Gerenciamento de Estado 2700a
Gerenciamento de Estado 2700b
Dados Dependentes de Plataforma 2710
Imagem do Jogo "Limpa" 2702b
Executando o Jogo (Estado A) 2701b
Dados de Diferença 2800

Fig. 28

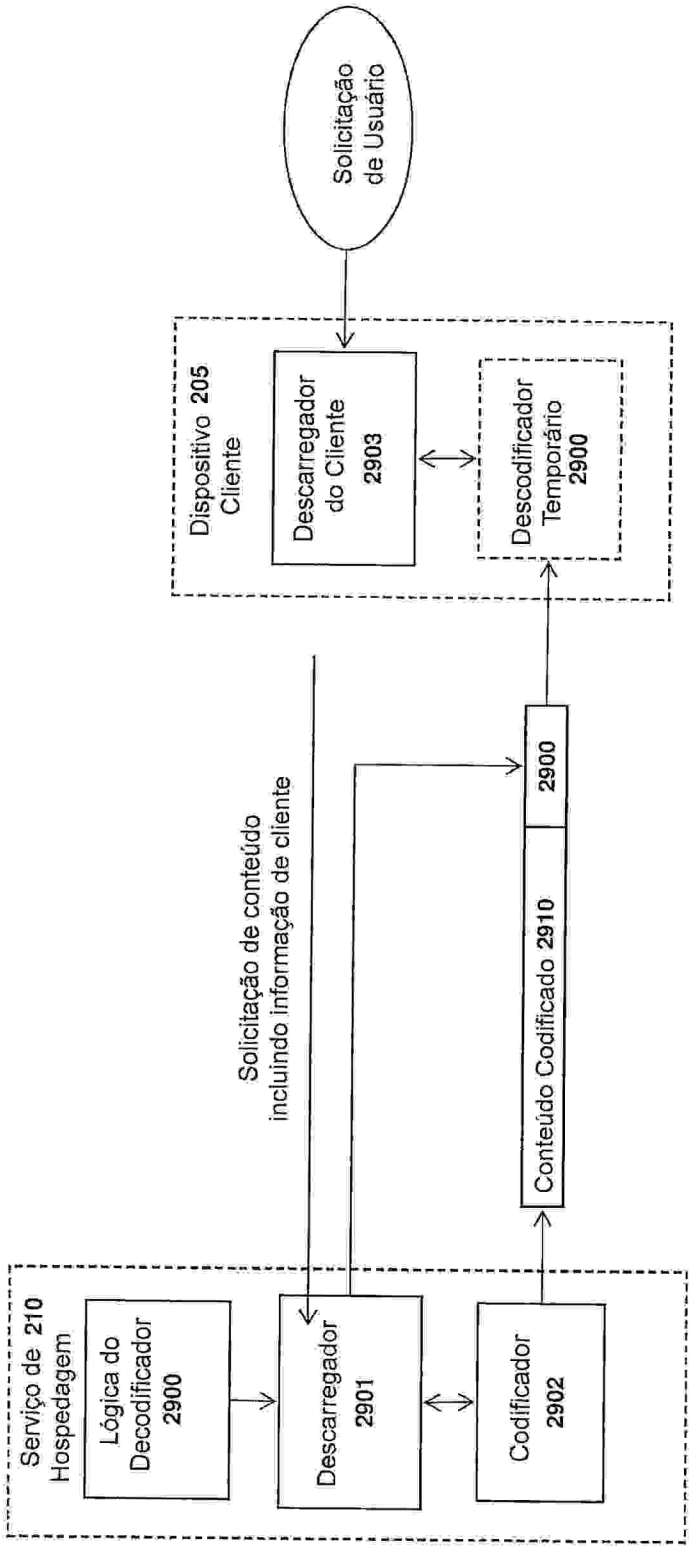

Solicitação de Usuário
Dispositivo 205 Cliente
Descarregador do Cliente 2903
Descodificador Temporário 2900
Solicitação de conteúdo incluindo informação de cliente
2900
Conteúdo Codificado 2910
Serviço de 210 Hospedagem
Lógica do Decodificador 2900
Descarregador 2901
Codificador 2902

Fig. 29

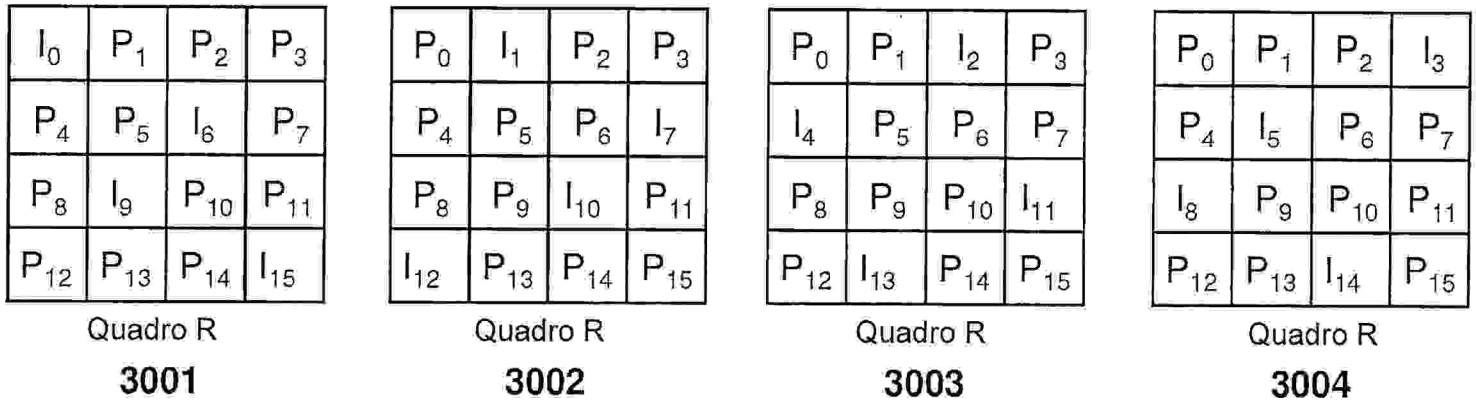

I0 P1 P2 P3
P4 P5 I6 P7
P8 I9 P10 P11
P12 P13 P14 I15
Quadro R
3001
P0 I1 P2 P3
P4 P5 P6 I7
P8 P9 I10 P11
I12 P13 P14 P15
Quadro R
3002
P0 P1 I2 P3
I4 P5 P6 P7
P8 P9 P10 I11
P12 I13 P14 P15
Quadro R
3003
P0 P1 P2 I3
P4 I5 P6 P7
I8 P9 P10 P11
P12 P13 I14 P15
Quadro R
3004

Fig. 30

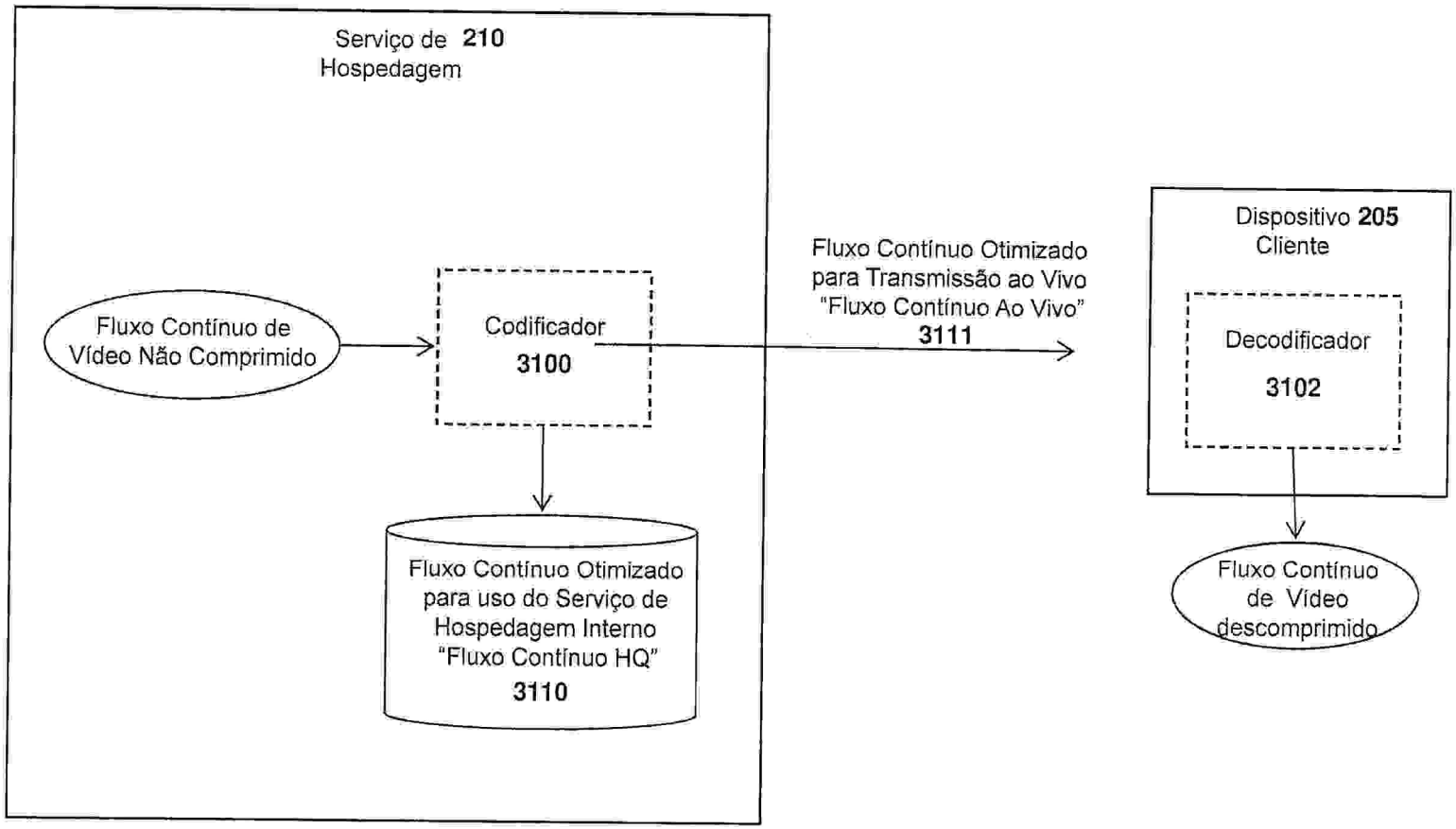

Serviço de 210
Hospedagem
Fluxo Contínuo de
Vídeo Não Comprimido
Codificador
3100
Fluxo Contínuo Otimizado
para uso do Serviço de
Hospedagem Interno
"Fluxo Contínuo HQ"
3110
Fluxo Contínuo Otimizado
para Transmissão ao Vivo
"Fluxo Contínuo Ao Vivo"
3111
Dispositivo 205
Cliente
Decodificador
3102
Fluxo Contínuo
de Vídeo
descomprimido

Fig. 31a

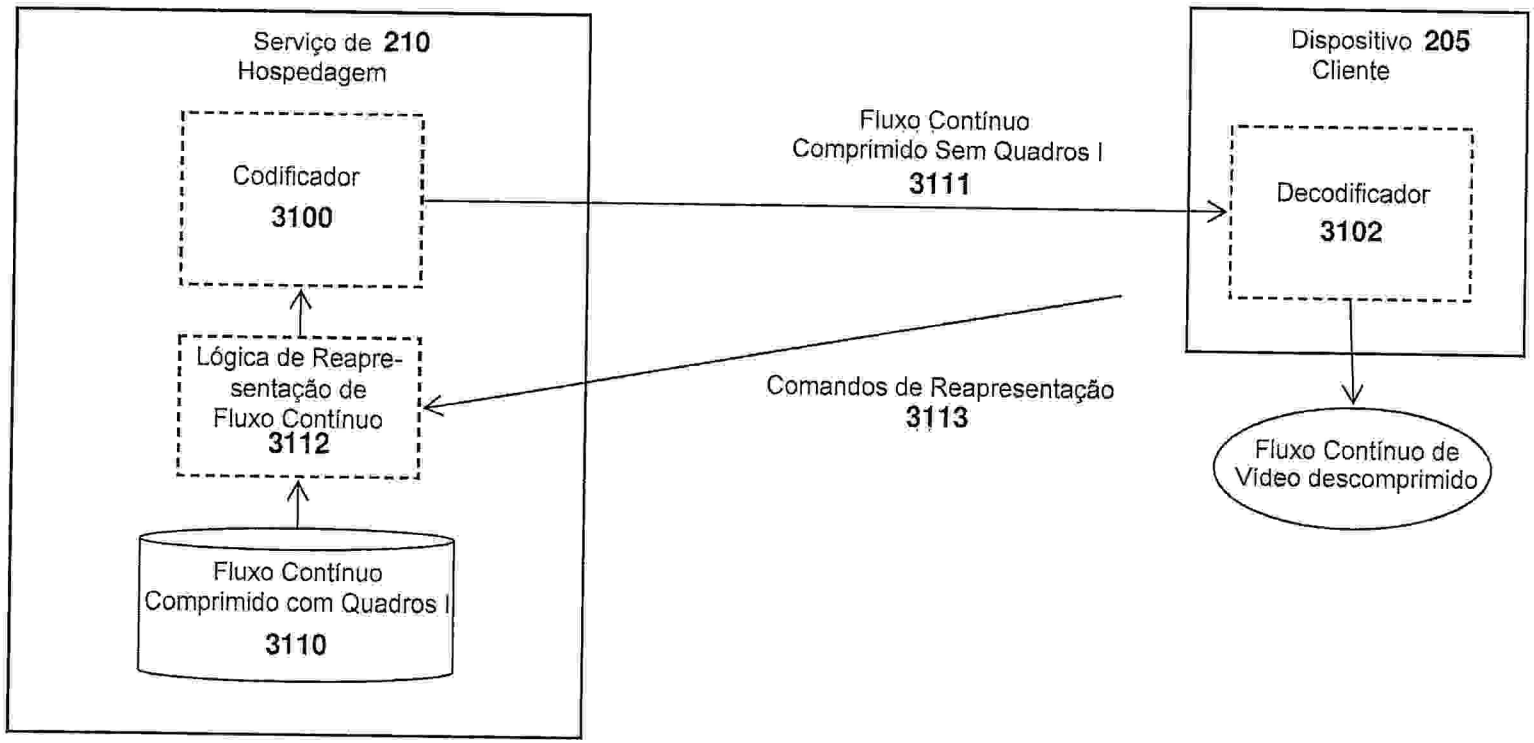

Serviço de 210
Hospedagem
Codificador
3100
Lógica de Reapre-
sentação de
Fluxo Contínuo
3112
Fluxo Contínuo
Comprimido com Quadros I
3110
Fluxo Contínuo
Comprimido Sem Quadros I
3111
Comandos de Reapresentação
3113
Dispositivo 205
Cliente
Decodificador
3102
Fluxo Contínuo de
Vídeo descomprimido

Fig. 31b

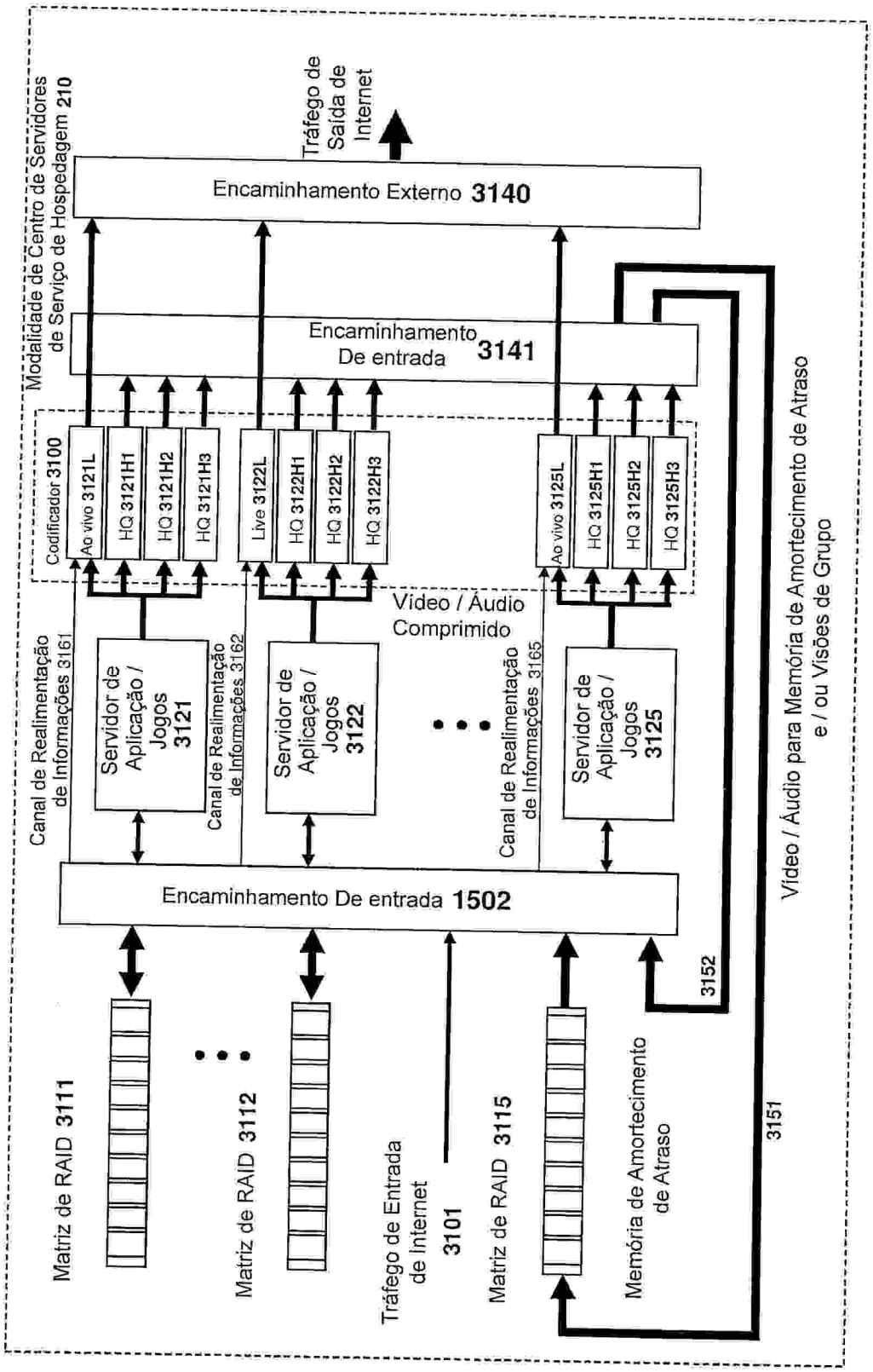

Modalidade de Centro de Servidores de Serviço de Hospedagem 210
Tráfego de Saída de Internet
Encaminhamento Externo 3140
Encaminhamento De entrada 3141
Codificador 3100
Ao vivo 3121L
HQ 3121H1
HQ 3121H2
HQ 3121H3
Live 3122L
HQ 3122H1
HQ 3122H2
HQ 3122H3
Ao vivo 3125L
HQ 3125H1
HQ 3125H2
HQ 3125H3
Vídeo / Áudio Comprimido
Canal de Realimentação de Informações 3161
Canal de Realimentação de Informações 3162
Canal de Realimentação de Informações 3165
Servidor de Aplicação / Jogos 3121
Servidor de Aplicação / Jogos 3122
Servidor de Aplicação / Jogos 3125
Encaminhamento De entrada 1502
Matriz de RAID 3111
Matriz de RAID 3112
Tráfego de Entrada de Internet 3101
Matriz de RAID 3115
Memória de Amortecimento de Atraso
3152
3151
Vídeo / Áudio para Memória de Amortecimento de Atraso e / ou Visões de Grupo

Fig. 31c

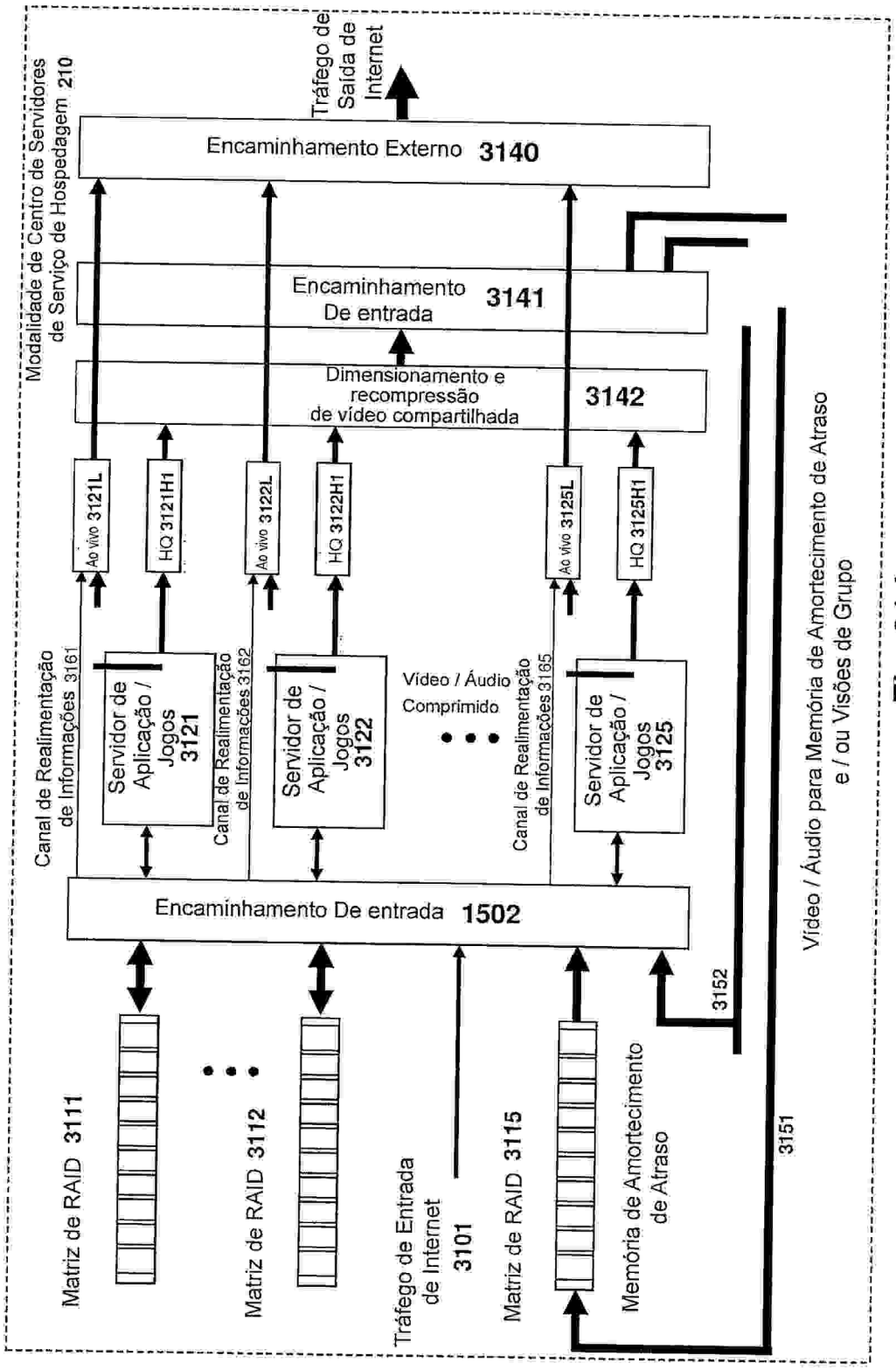

Modalidade de Centro de Servidores de Serviço de Hospedagem 210
Tráfego de Saída de Internet
Encaminhamento Externo 3140
Encaminhamento De entrada 3141
Dimensionamento e recompressão de vídeo compartilhada 3142
Ao vivo 3121L
HQ 3121H1
Ao vivo 3122L
HQ 3122H1
Ao vivo 3125L
HQ 3125H1
Canal de Realimentação de Informações 3161
Canal de Realimentação de Informações 3162
Canal de Realimentação de Informações 3165
Servidor de Aplicação / Jogos 3121
Servidor de Aplicação / Jogos 3122
Vídeo / Áudio Comprimido
Servidor de Aplicação / Jogos 3125
Encaminhamento De entrada 1502
Matriz de RAID 3111
Matriz de RAID 3112
Tráfego de Entrada de Internet 3101
Matriz de RAID 3115
Memória de Amortecimento de Atraso
3152
3151
Vídeo / Áudio para Memória de Amortecimento de Atraso e / ou Visões de Grupo

Fig. 31d

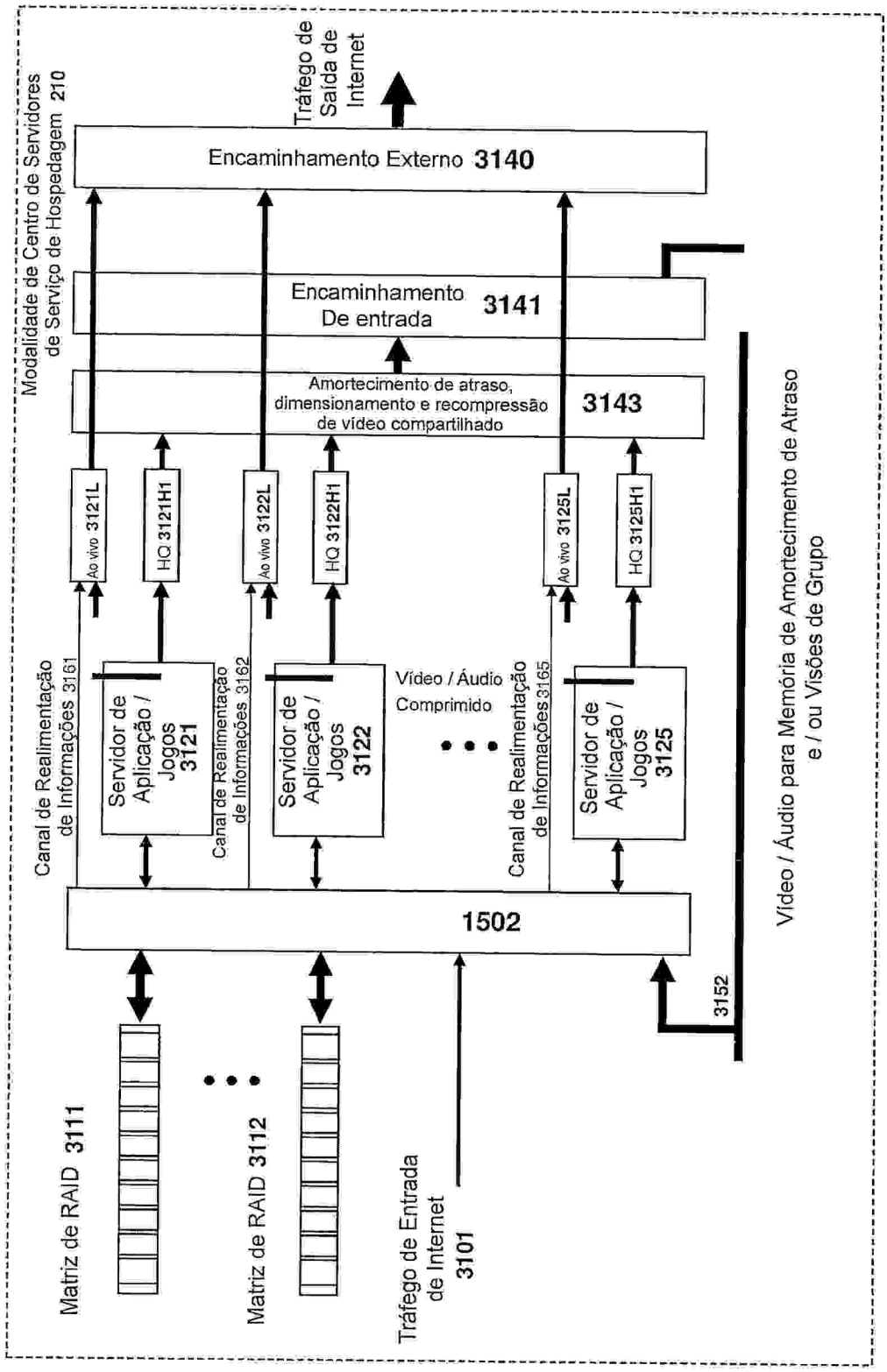

Modalidade de Centro de Servidores de Serviço de Hospedagem 210
Tráfego de Saída de Internet
Encaminhamento Externo 3140
Encaminhamento De entrada 3141
Amortecimento de atraso, dimensionamento e recompressão de vídeo compartilhado 3143
Ao vivo 3121L
HQ 3121H1
Ao vivo 3122L
HQ 3122H1
Ao vivo 3125L
HQ 3125H1
Canal de Realimentação de Informações 3161
Canal de Realimentação de Informações 3162
Canal de Realimentação de Informações 3165
Servidor de Aplicação / Jogos 3121
Servidor de Aplicação / Jogos 3122
Servidor de Aplicação / Jogos 3125
Vídeo / Áudio Comprimido
1502
3152
Vídeo / Áudio para Memória de Amortecimento de Atraso e / ou Visões de Grupo
Matriz de RAID 3111
Matriz de RAID 3112
Tráfego de Entrada de Internet 3101

Fig. 31e

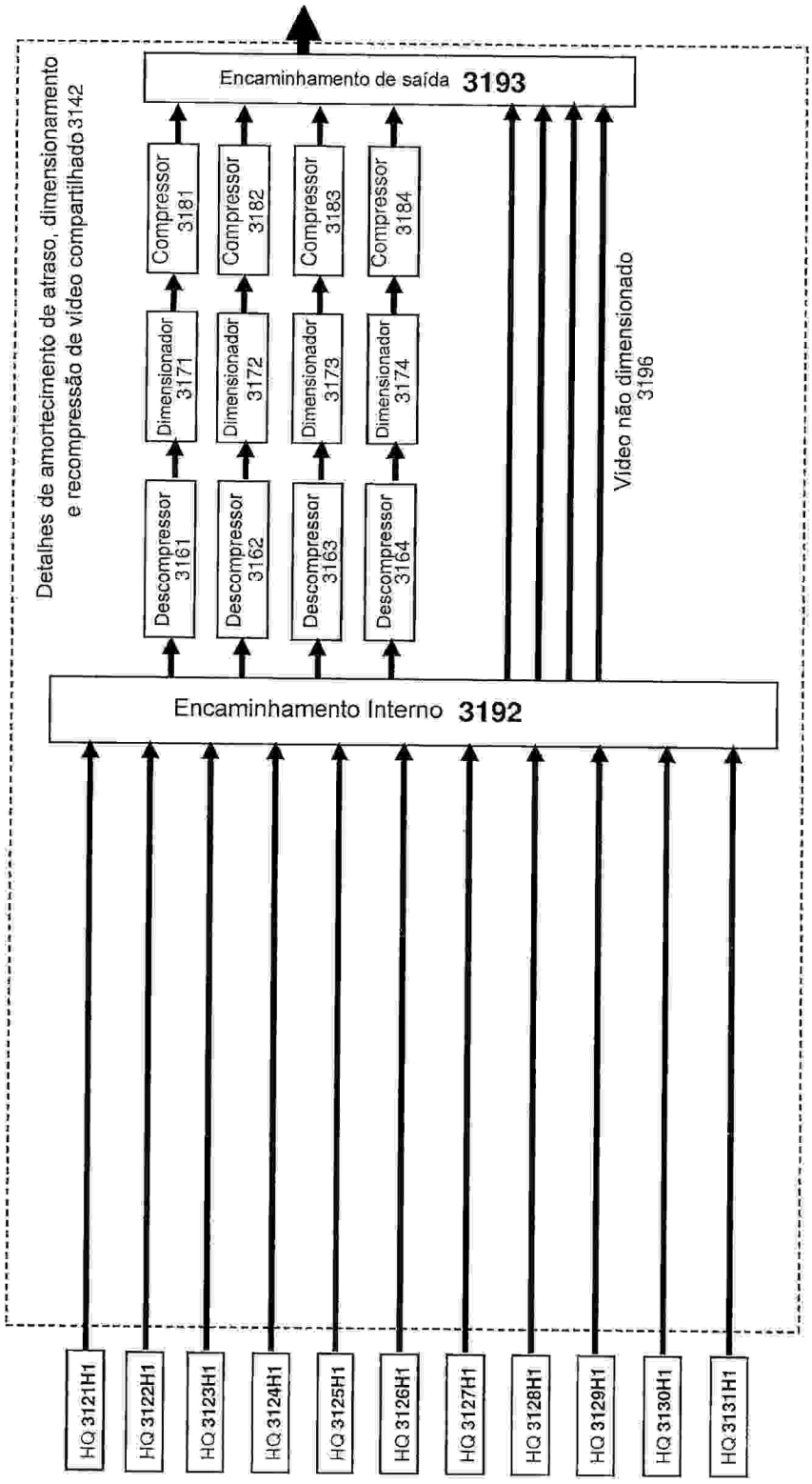

Detalhes de amortecimento de atraso, dimensionamento e recompressão de vídeo compartilhado 3142
Encaminhamento de saída 3193
Compressor 3181
Compressor 3182
Compressor 3183
Compressor 3184
Dimensionador 3171
Dimensionador 3172
Dimensionador 3173
Dimensionador 3174
Descompressor 3161
Descompressor 3162
Descompressor 3163
Descompressor 3164
Vídeo não dimensionado 3196
Encaminhamento Interno 3192
HQ 3121H1
HQ 3122H1
HQ 3123H1
HQ 3124H1
HQ 3125H1
HQ 3126H1
HQ 3127H1
HQ 3128H1
HQ 3129H1
HQ 3130H1
HQ 3131H1

Fig. 31f

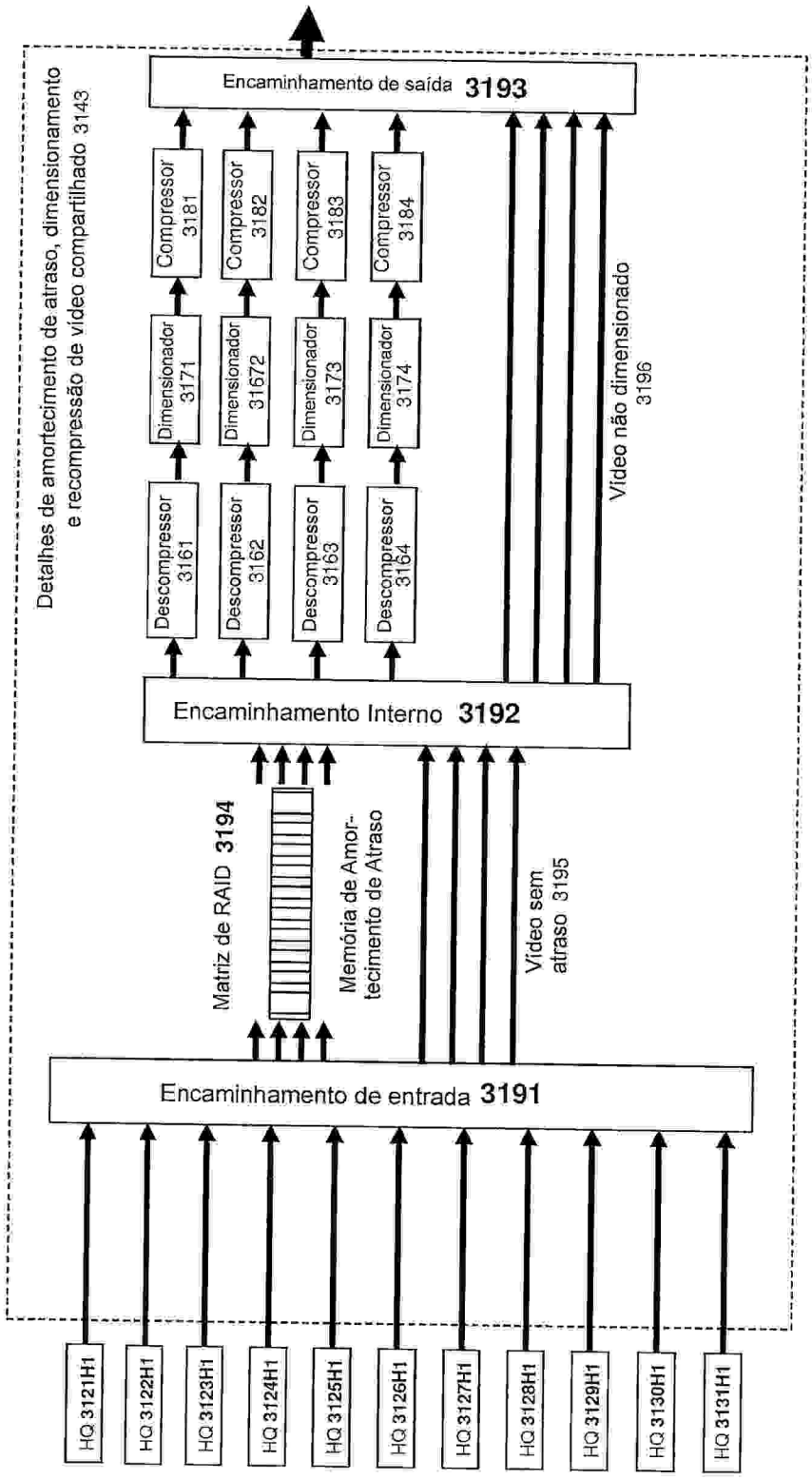

Detalhes de amortecimento de atraso, dimensionamento e recompressão de vídeo compartilhado 3143
Encaminhamento de saída 3193
Compressor 3181
Compressor 3182
Compressor 3183
Compressor 3184
Dimensionador 3171
Dimensionador 31672
Dimensionador 3173
Dimensionador 3174
Descompressor 3161
Descompressor 3162
Descompressor 3163
Descompressor 3164
Vídeo não dimensionado 3196
Encaminhamento Interno 3192
Matriz de RAID 3194
Memória de Amortecimento de Atraso
Vídeo sem atraso 3195
Encaminhamento de entrada 3191
HQ 3121H1
HQ 3122H1
HQ 3123H1
HQ 3124H1
HQ 3125H1
HQ 3126H1
HQ 3127H1
HQ 3128H1
HQ 3129H1
HQ 3130H1
HQ 3131H1

Fig. 31g

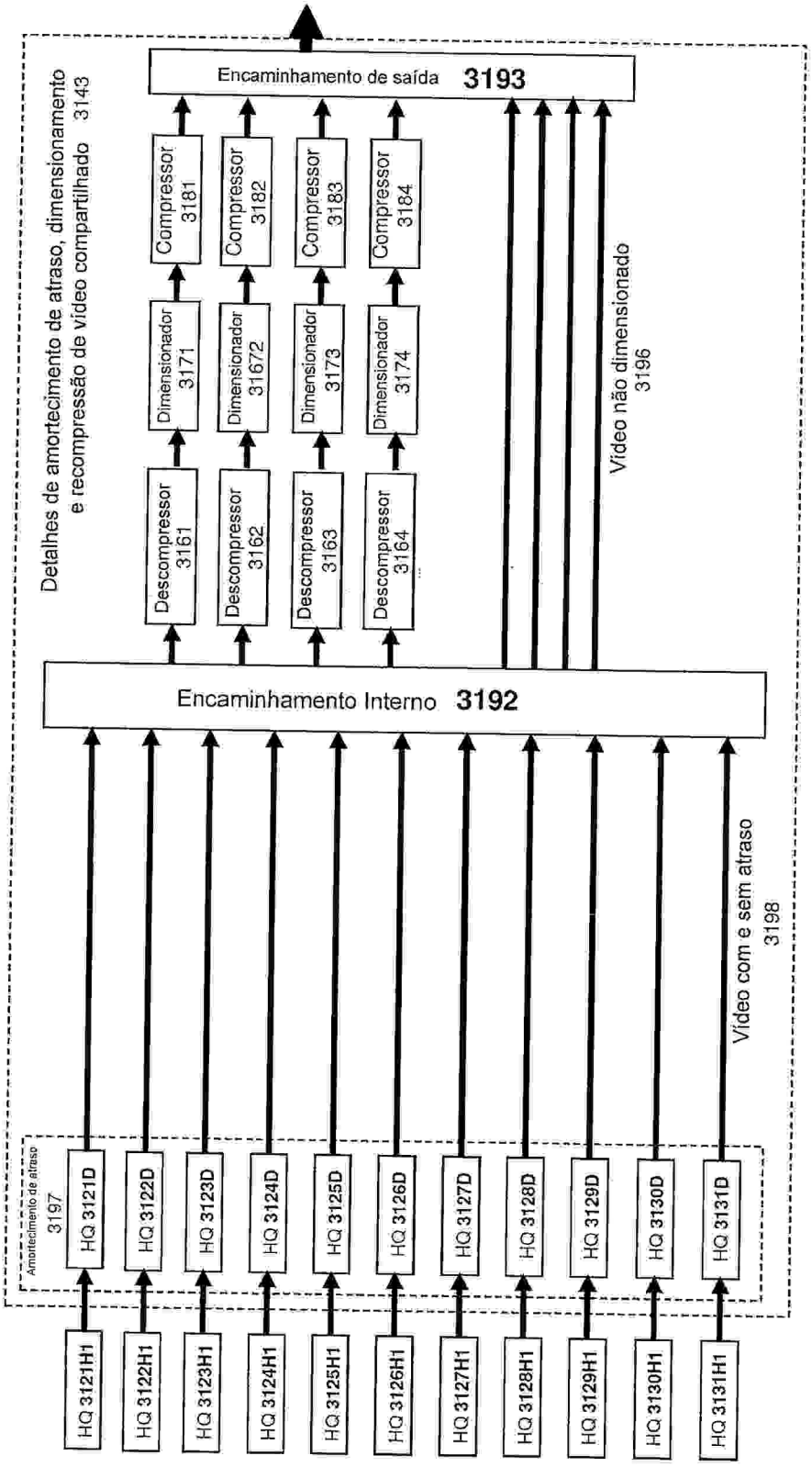

Detalhes de amortecimento de atraso, dimensionamento e recompressão de vídeo compartilhado 3143
Encaminhamento de saída 3193
Compressor 3181
Compressor 3182
Compressor 3183
Compressor 3184
Dimensionador 3171
Dimensionador 31672
Dimensionador 3173
Dimensionador 3174
Descompressor 3161
Descompressor 3162
Descompressor 3163
Descompressor 3164
Vídeo não dimensionado 3196
Encaminhamento Interno 3192
Amortecimento de atraso 3197
HQ 3121D
HQ 3122D
HQ 3123D
HQ 3124D
HQ 3125D
HQ 3126D
HQ 3127D
HQ 3128D
HQ 3129D
HQ 3130D
HQ 3131D
Vídeo com e sem atraso 3198
HQ 3121H1
HQ 3122H1
HQ 3123H1
HQ 3124H1
HQ 3125H1
HQ 3126H1
HQ 3127H1
HQ 3128H1
HQ 3129H1
HQ 3130H1
HQ 3131H1

Fig. 31h

RESUMO

Patente de Invenção: **"SISTEMA E MÉTODO PARA COMPRESSÃO DE VÍDEO MULTIFLUXO".**

A presente invenção refere-se a um servidor de vídeo game recebendo entradas de usuário relacionados a um vídeo game online, para render uma sequência de imagens de vídeo; um primeiro codificador de fluxo para comprimir a sequência de imagens de vídeo e gerar um fluxo de vídeo ao vivo durante uma sessão de jogo ao vivo com um usuário de um aparelho de cliente, o primeiro codificador de fluxo, recebendo sinais de retorno de canal do aparelho do cliente e responsivamente adaptando a compressão da sequência de imagens de vídeo baseado nos sinais de retorno de canal, o primeiro codificador de fluxo transmitindo continuamente o fluxo de vídeo ao vivo ao aparelho do cliente durante a sessão de jogo ao vivo com o usuário; um segundo codificador de fluxo para comprimir a sequência de imagens de vídeo numa qualidade de vídeo específica e/ou taxa de compressão não relacionada ao sinal de retorno do canal durante a sessão de jogo ao vivo com o usuário, assim gerando qualidade alta de fluxo de vídeo (HQ), a HQ de fluxo de vídeo tendo qualidade de vídeo relativamente maior e/ou taxa de compressão menor que o fluxo de vídeo ao vivo; e um dispositivo de armazenamento para armazenar o fluxo de vídeo HQ para a subsequente reprodução ao usuário do aparelho de cliente e a outros usuários por requisição.