



Network
Architecture
Res. Group



Multimedia
Information
System Lab.

ユーザレベルの品質保証を実現する QoSマッピング技術

大阪大学
若宮直紀

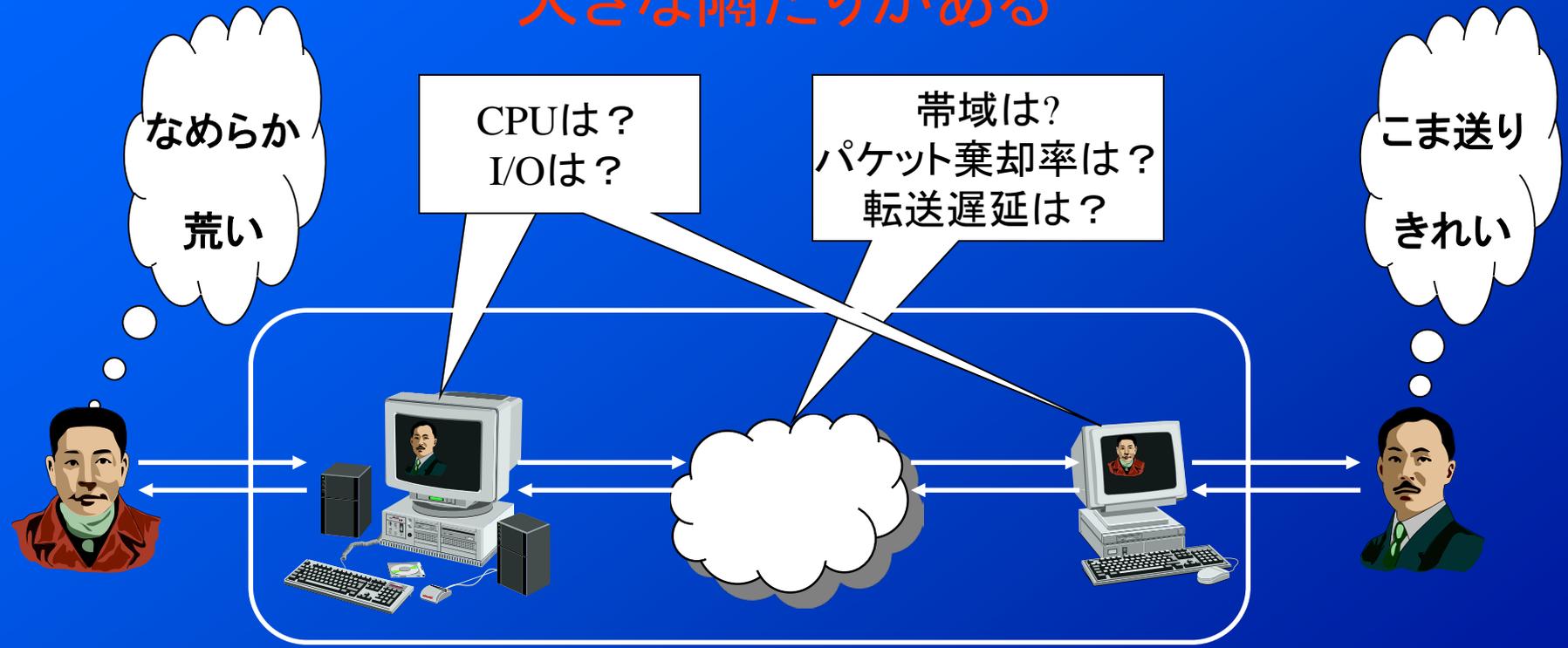
wakamiya@ics.es.osaka-u.ac.jp

QoSアーキテクチャ

- よりよい通信サービスを
 - 単なる通信路, 接続口
 - 他にない速さを謳うだけでは不十分
- サービス品質 (Quality of Service) を制御, 保証するためのQoSアーキテクチャが必要
 - どのようなQoSをどのように制御, 保証するか?
 - ユーザを向いた制御

ユーザとシステム

大きな隔たりがある



TV電話システム

さまざまなQoSパラメータ: ユーザQoS

- ユーザの思うサービス品質
 - データをはやく正確にダウンロードできる
 - ホームページがすぐに見られる
 - 音声, 動画像がきれい



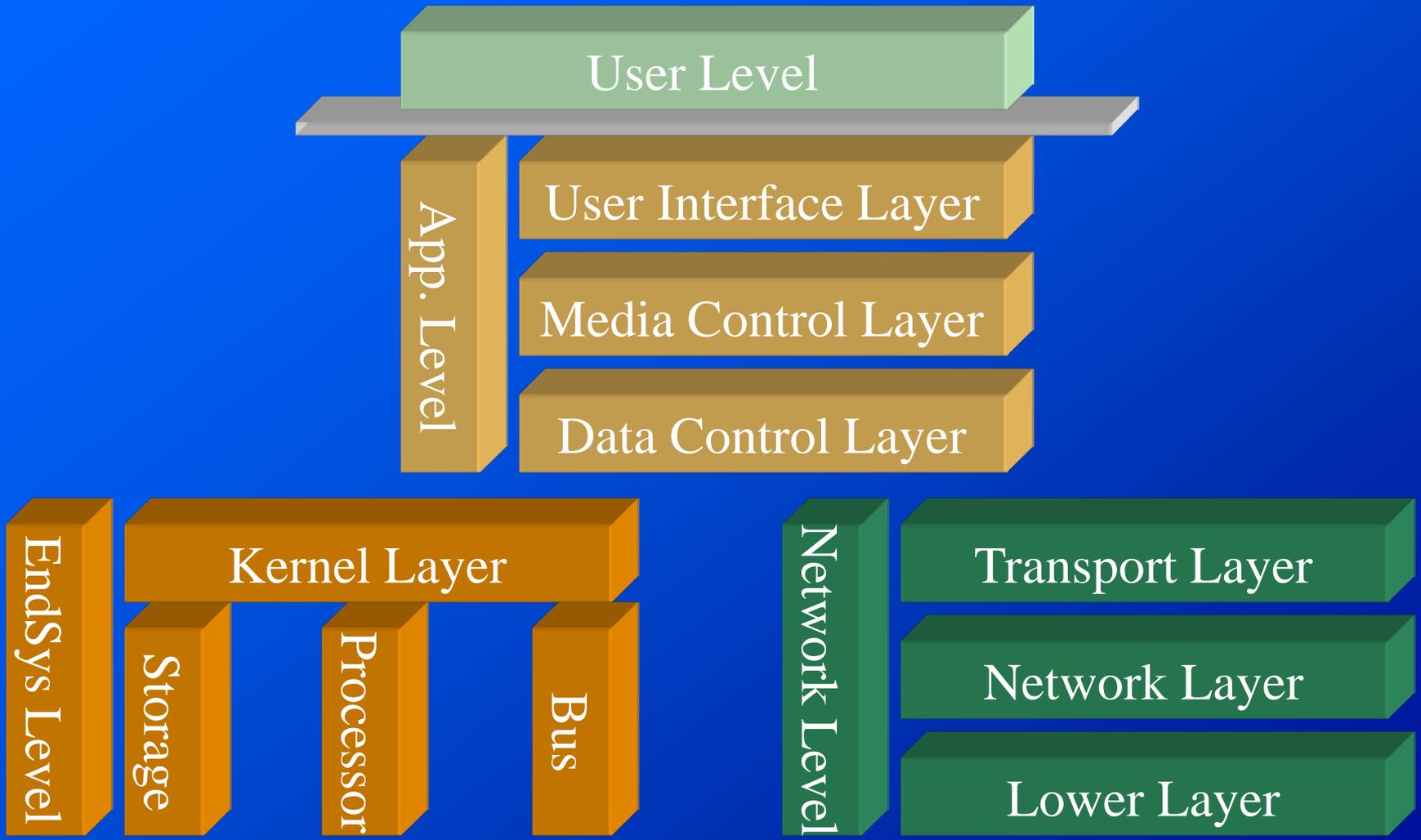
さまざまなQoSパラメータ: ネットワークQoS

- ネットワークレベルで制御できるQoS
 - 呼損率
 - 転送レート
 - 棄却率
 - 転送遅延
 - 転送遅延揺らぎ
 - ビットエラー率

さまざまなQoSパラメータ: エンドシステムQoS

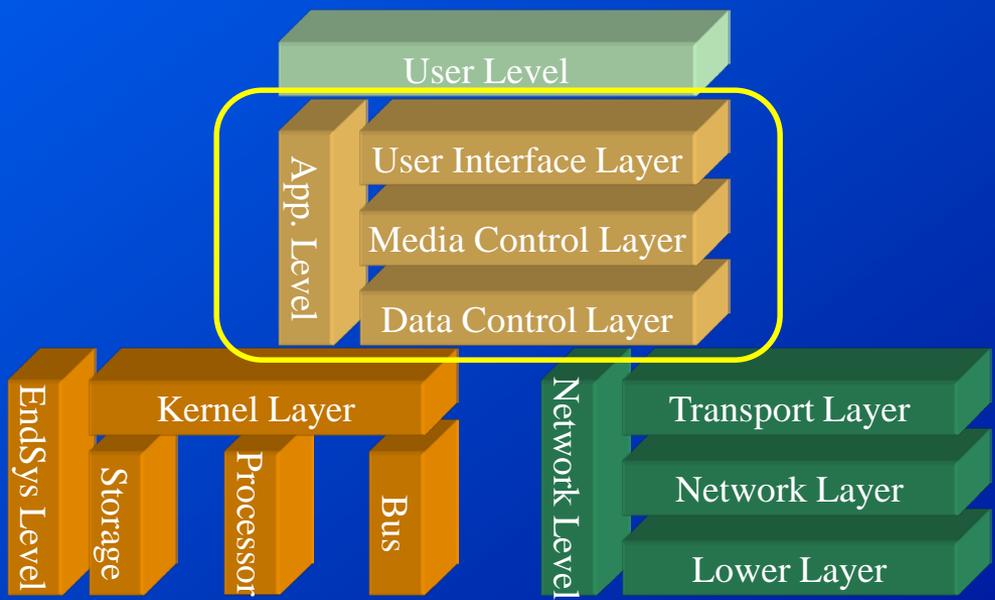
- エンドシステムで制御できるQoS
 - CPU処理能力
 - メモリ容量
 - 処理の周期性
 - デッドライン
 - バス速度
 - I/O処理能力

QoSマッピング



アプリケーションレベルの役割

- ユーザQoSと下位レベルQoSの橋渡し
- 下位レベルQoS間の橋渡し
- ユーザQoSに応じた制御手法の選択
- 下位レベルQoSの変化の吸収



ユーザQoSとアプリケーションQoS

マッピングの例

| | ユーザQoS | | アプリケーションQoS |
|-------------|--------------------|----------|-------------------------------------|
| データ | データを速く正確にダウンロードできる | 品質 時間 | 応答時間, 転送時間など |
| WWW | ホームページが すぐに見られる | 時間 | 起動時間, 応答時間, 表示速度など |
| マルチ メディア | 音声, 動画像が きれい | 品質 | 標本化レート, 符号化手法, 解像度, フレームレートなど |

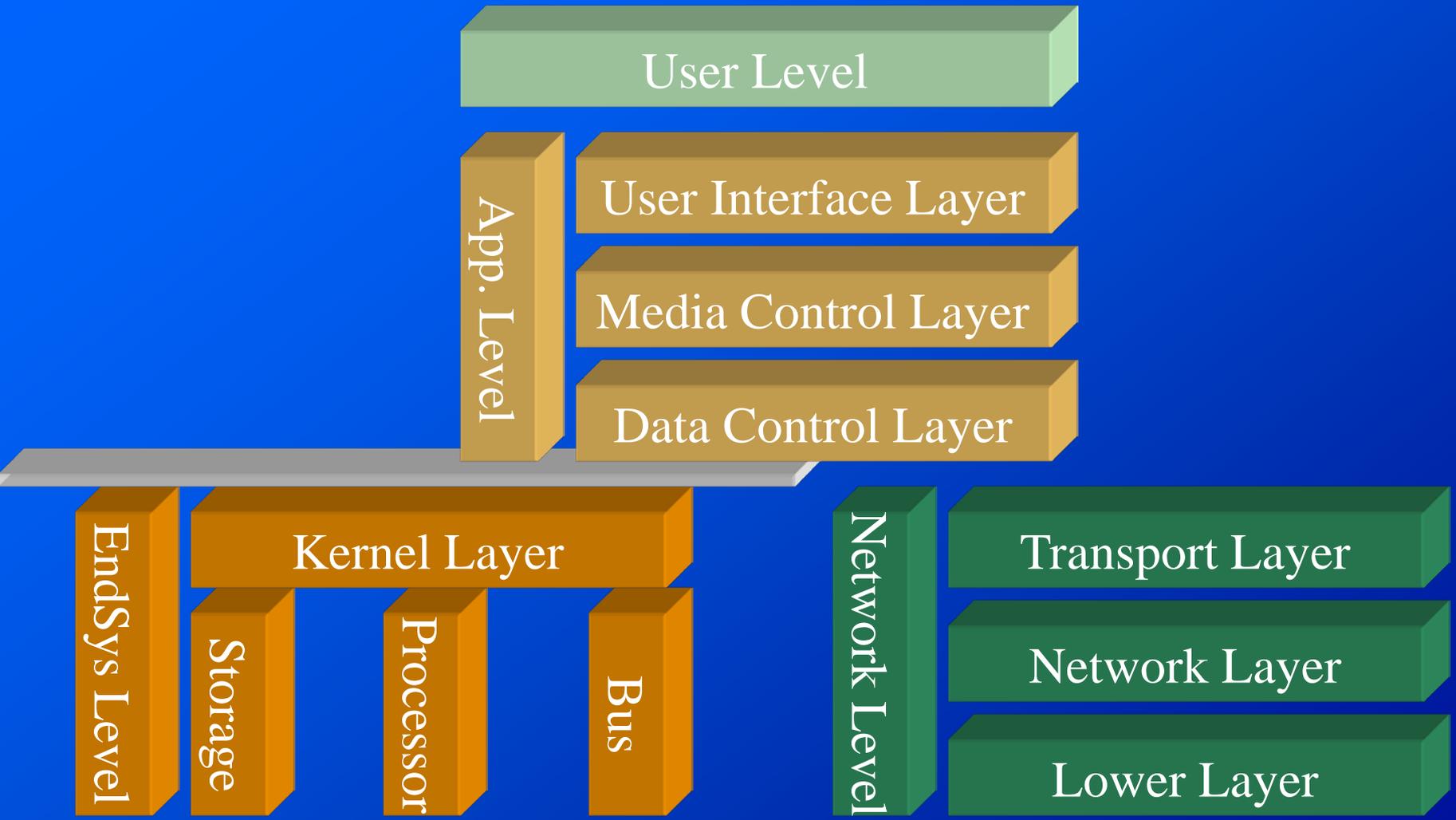
ユーザQoSの数值化

- 制御, 保証するためには実体化が必要
- 時間
 - 何ミリ秒以内なら「速く」なのか？
 - 何秒以内なら「すぐ」なのか？（8秒ルール）
- 品質
 - 音声：
 - MOS (Mean Opinion Score)
 - PSNR (Peak Signal to Noise Ratio)
 - PSQN (Perceptual Speech Quality Measure)
 - 動画像：
 - DSCQS (Double Stimulus Continuous Quality Scale)
 - SSCQE (Single Stimulus Continuous Quality Evaluation)
 - PSNR (Peak Signal to Noise Ratio)

ユーザQoSを考慮した アプリケーションQoS保証, 制御技術

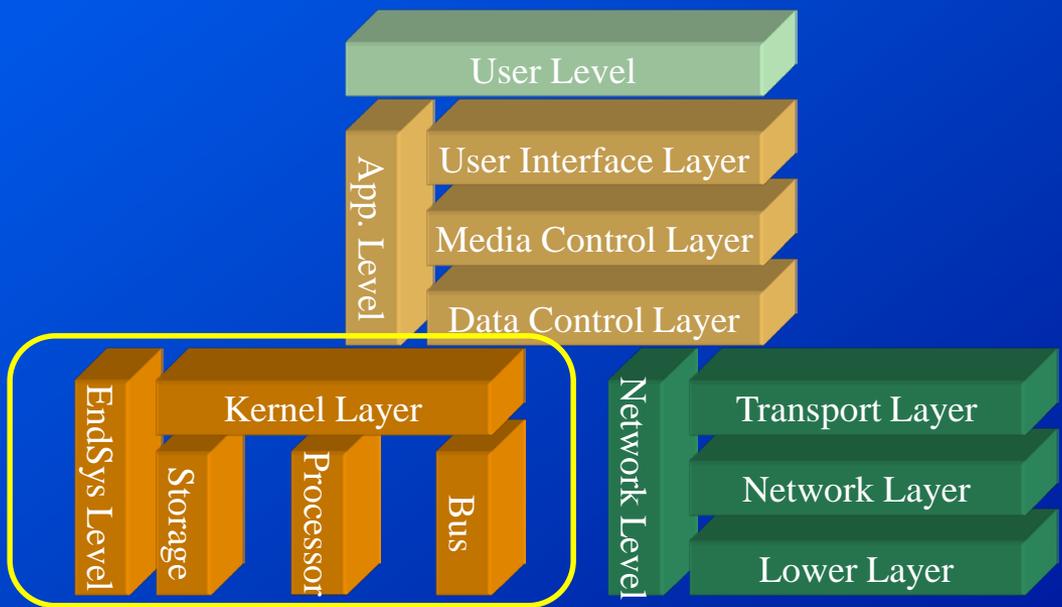
- ホームページがすぐに見られる
 - アプリケーション起動の高速化
 - persistent connection, pipelining
 - プロキシ, サーバ負荷分散
 - Interlaced GIF, Progressive JPEG
 - 先読み, キャッシュ
- 音声, 動画像がきれい
 - 符号化手法
 - サンプリングレート
 - 解像度, フレームレート, 色数

QoSマッピング



エンドシステムレベルの役割

- エンドシステム資源の管理
- アプリケーション間の調整



アプリケーションQoSとエンドシステムQoS

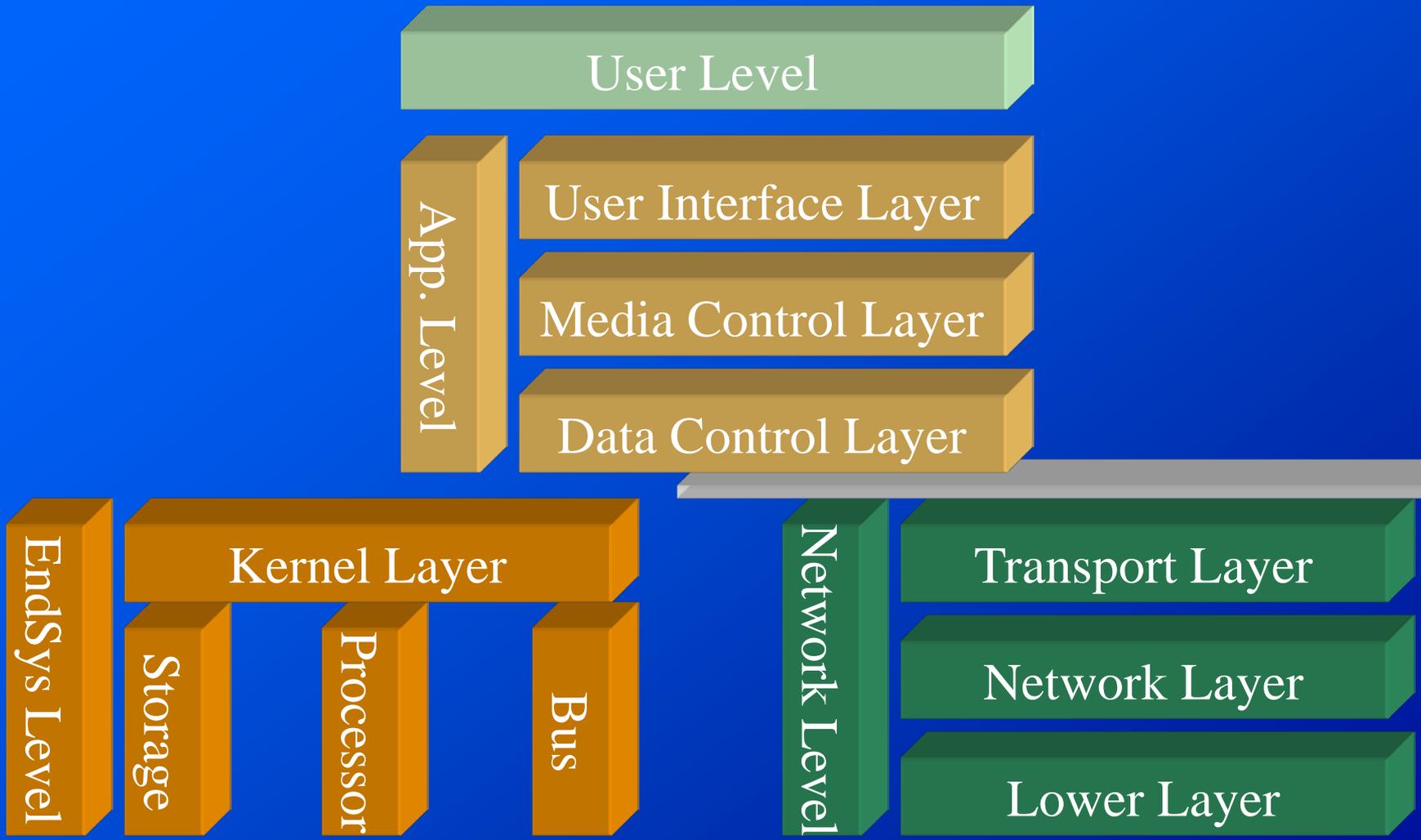
マッピングの例

| | アプリケーションQoS | エンドシステムQoS |
|---------|-------------------------------------|-------------------------------------|
| データ | 応答時間, 転送時間など | バス速度, I/O処理能力など |
| WWW | 起動時間, 応答時間, 表示時間など | CPU処理能力, バス速度, I/O処理能力など |
| マルチメディア | 標本化レート, 符号化手法, 解像度, フレームレートなど | CPU処理能力, バス速度, I/O処理能力, 周期性など |

アプリケーションQoSを考慮した エンドシステムQoS保証, 制御技術

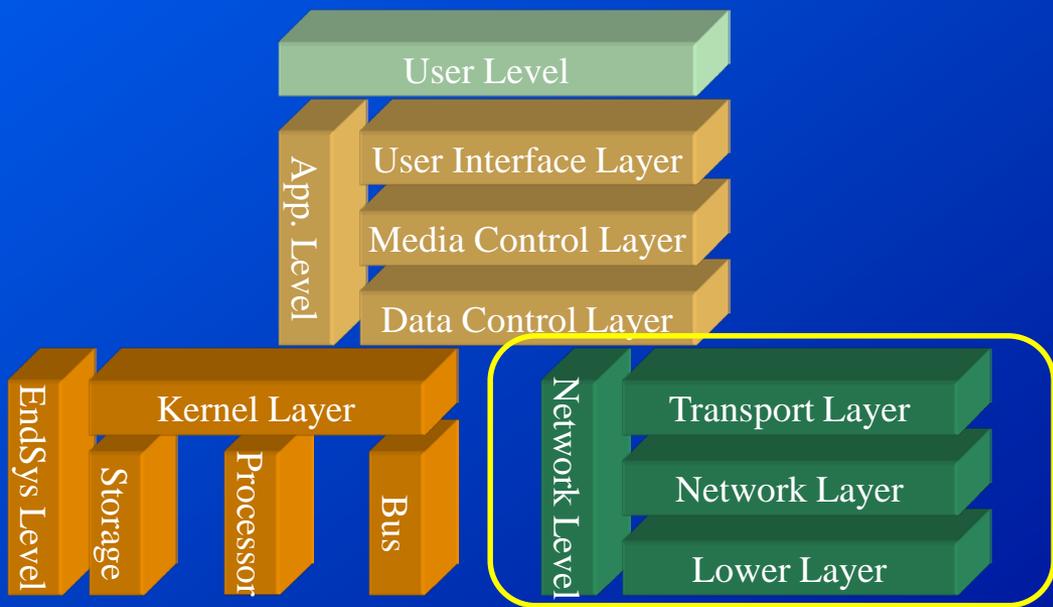
- 起動時間
 - ディスク配置最適化
- 応答時間, 表示時間
 - タスクスケジューリング
 - メモリアクセス高速化
- 標本化レート, 符号化手法, 解像度, フレームレート
 - CPU処理能力割当
 - メモリ容量, バス容量割当
 - デッドライン制御

QoSマッピング



ネットワークレベルの役割

- ネットワーク資源の管理
- QoSに応じたプロトコル, サービスレベル, パラメータの選択, 設定
- ネットワーク品質の揺らぎの吸収



ユーザQoSとアプリケーションQoS

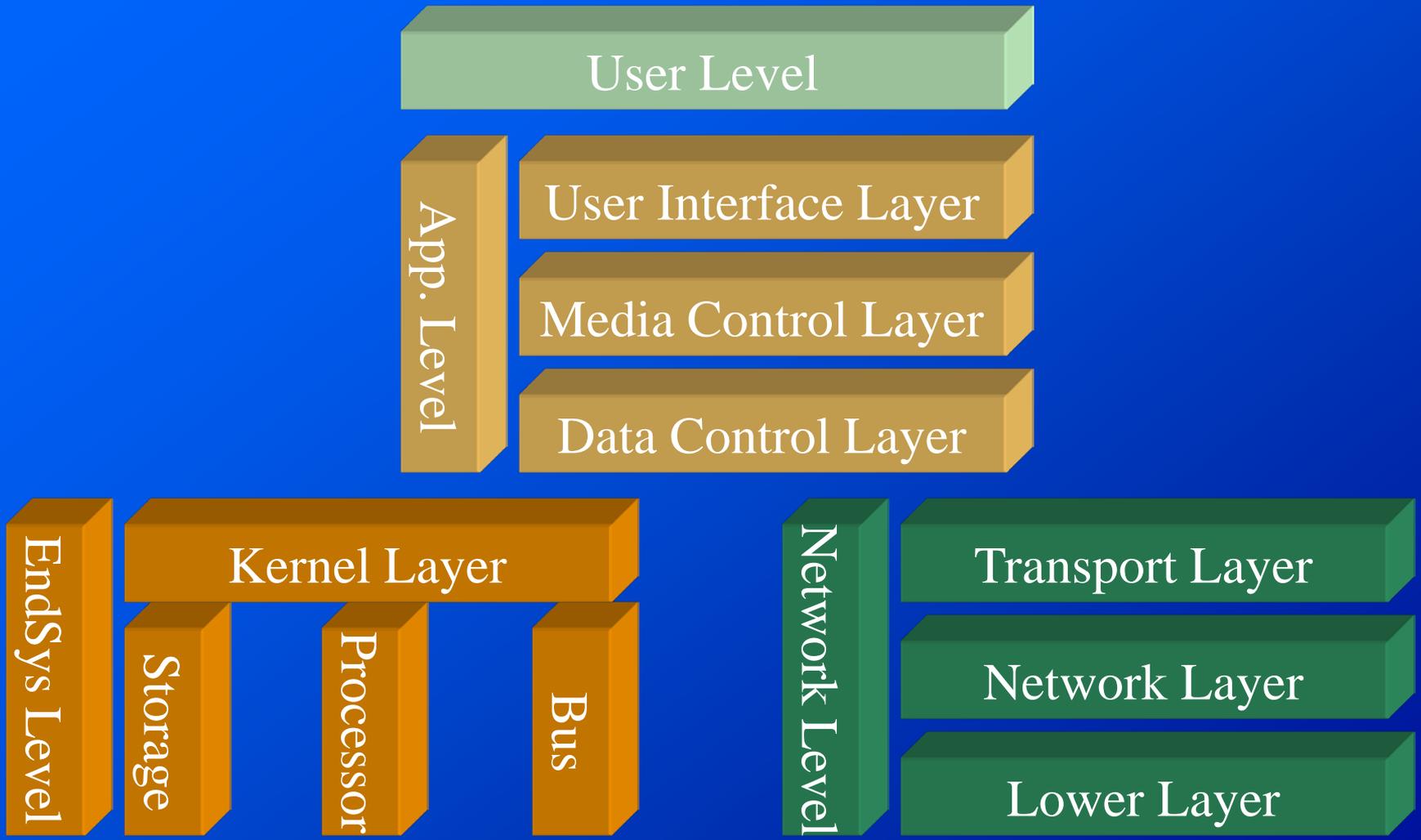
マッピングの例

| | アプリケーションQoS | ネットワークQoS |
|---------|-------------------------------------|----------------------|
| データ | 応答時間, 転送時間など | 棄却率, 転送遅延など |
| WWW | 応答時間など | 伝搬遅延, 転送遅延など |
| マルチメディア | 標本化レート, 符号化手法, 解像度, フレームレートなど | 帯域, 転送遅延, 遅延揺らぎなど |

アプリケーションQoSを考慮した ネットワークQoS保証, 制御技術

- 応答時間
 - QoSルーティング
- 転送時間
 - パケットスケジューリング
- 標本化レート, 符号化手法, 解像度, フレームレート
 - 帯域割当
 - 実時間通信プロトコル

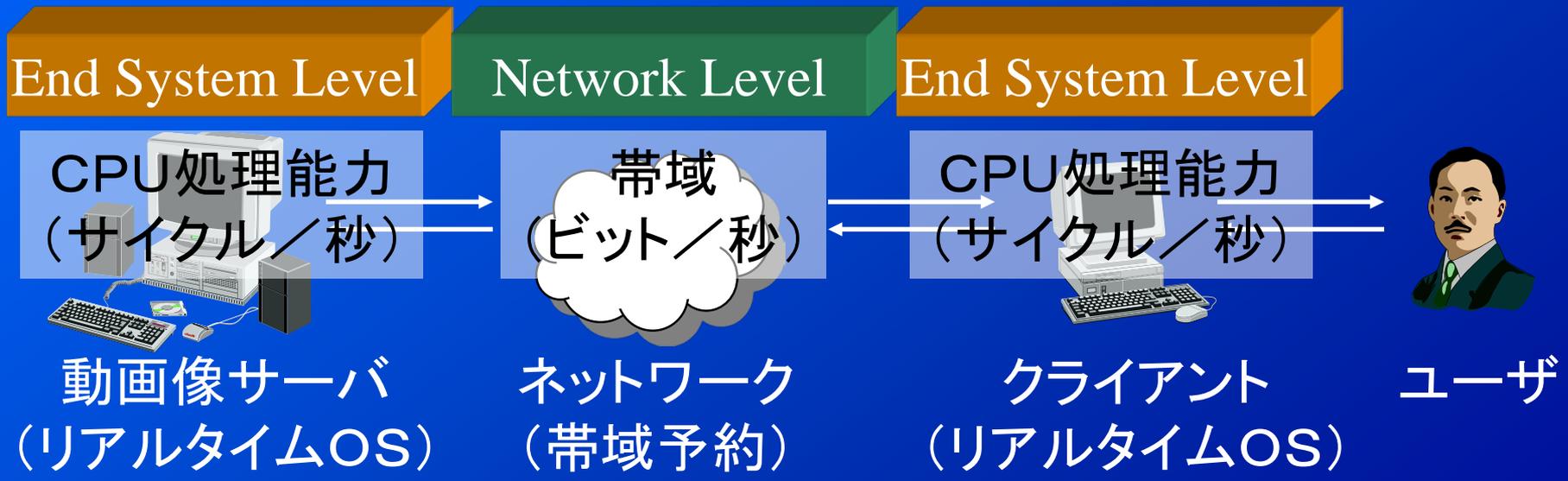
QoSマッピング



QoSマッピングによる 動画像通信におけるエンド間QoS制御

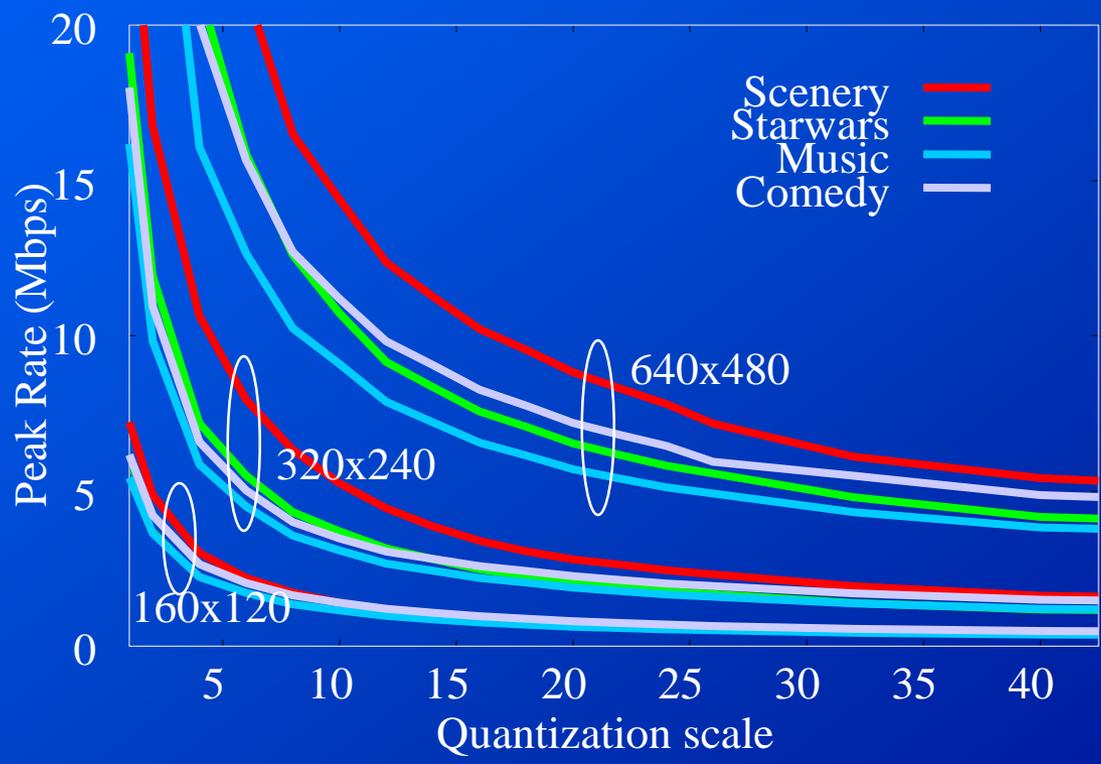


SNR解像度, 時間解像度, 空間解像度, GoP構成



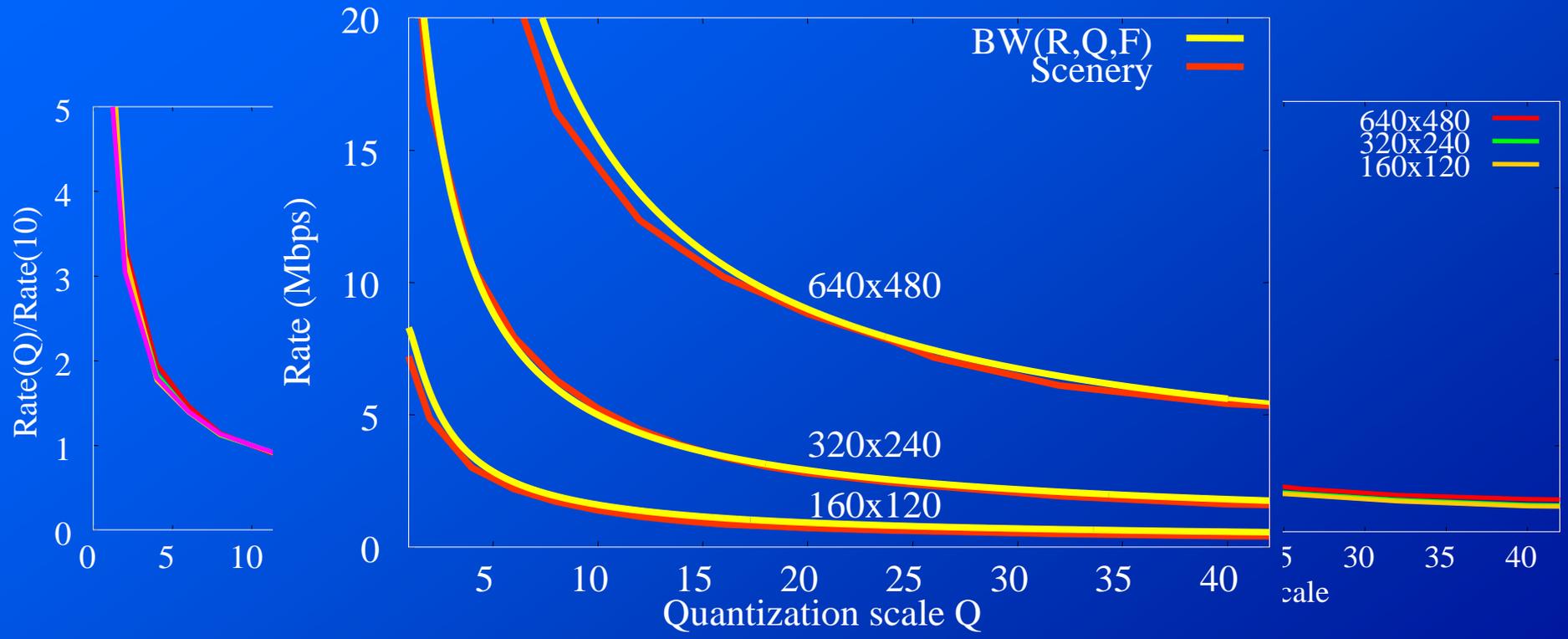
アプリケーションQoSとネットワークQoS

- 符号化パラメータと帯域の関係



アプリケーションQoSとネットワークQoSのマッピング(定式化)

- 定式化によりQoS制御のアルゴリズム化が可能



$$B \cong 3.1^{\log_4 \frac{R}{640 \times 480}} \left(\alpha + \frac{\beta}{Q} - \frac{\gamma}{Q^2} \right) \frac{F}{30} B_{base}$$

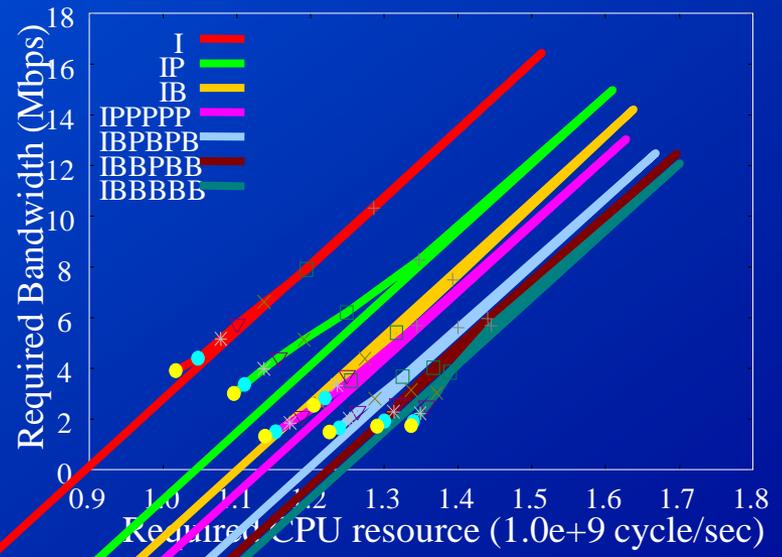
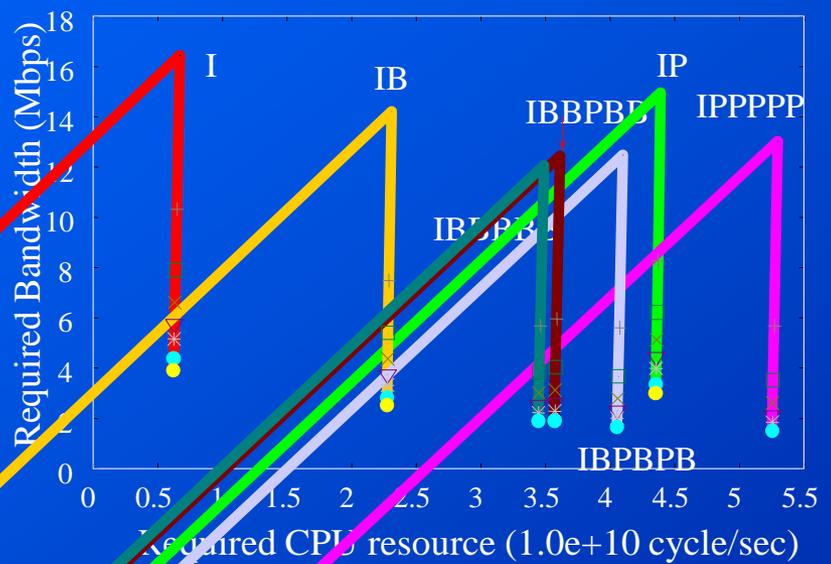
アプリケーションQoSと エンドシステムQoSのマッピング

- サーバCPU処理能力

$$S \cong S_G \frac{R}{640 \times 480} \frac{F}{30}$$

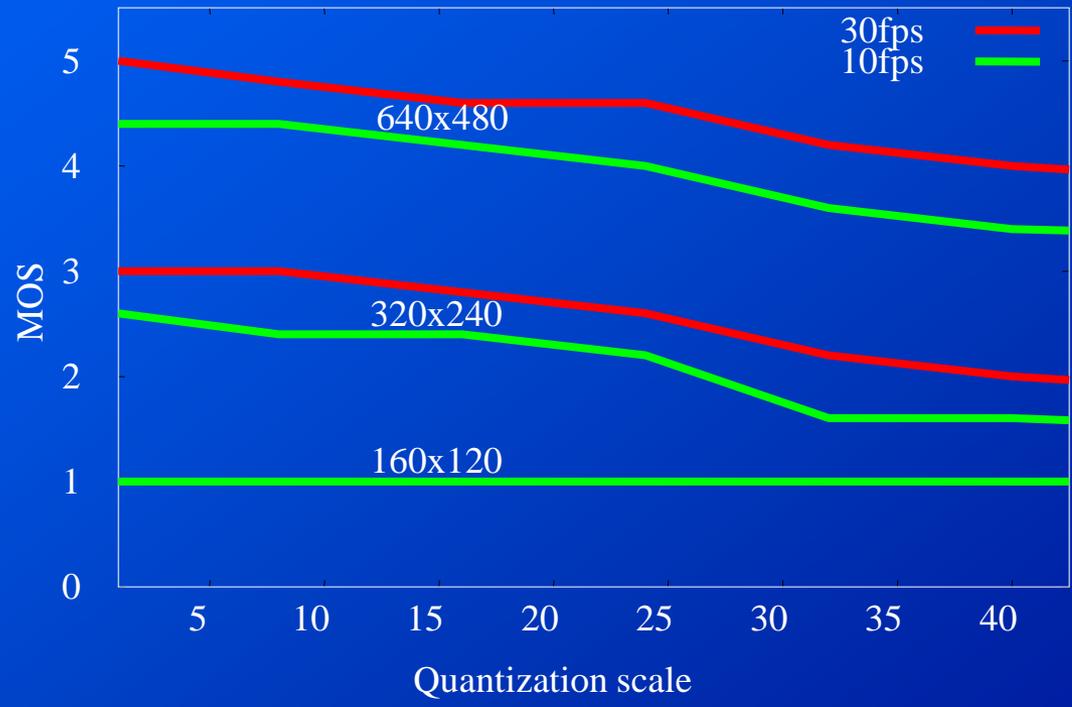
- クライアントCPU処理能力

$$C \cong 40B + \left(870 + \frac{N_P}{N} \delta + \frac{N_B}{N} \varepsilon \right) \frac{R}{640 \times 480} \frac{F}{30}$$



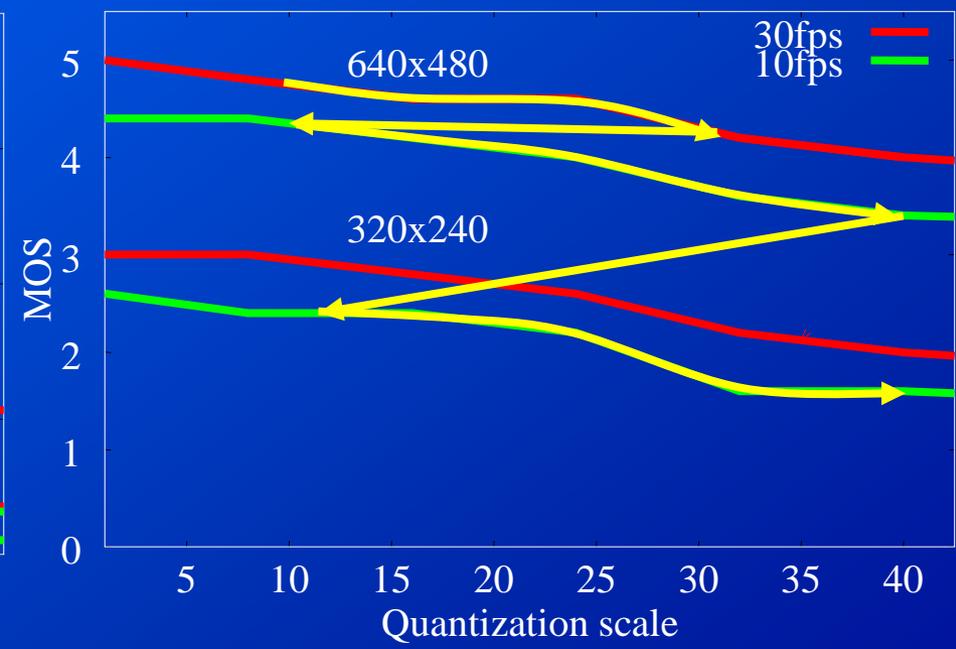
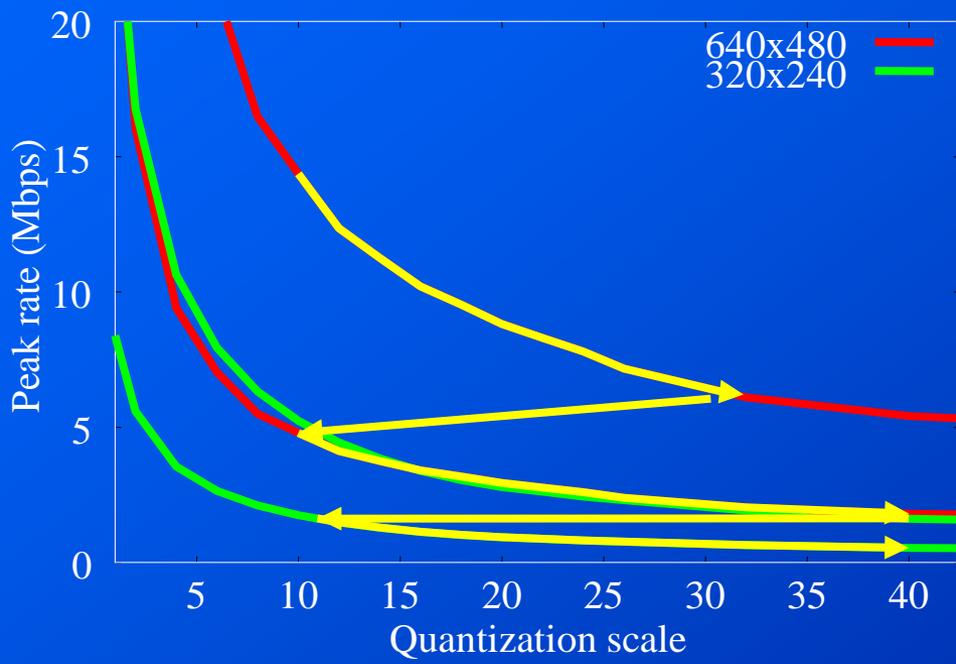
ユーザQoSとアプリケーションQoSの マッピング

- 動画像の主観評価結果と符号化パラメータの対応付け(マッピング)



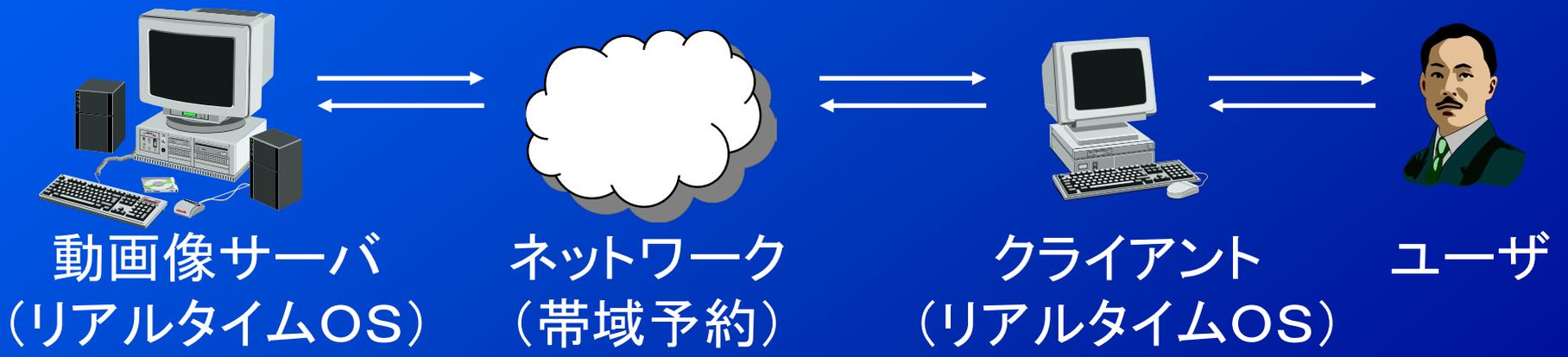
QoSマッピングを利用したQoS制御

- 利用可能な資源量(提供されるQoS)に応じたユーザレベルQoSを考慮した動的なQoS制御



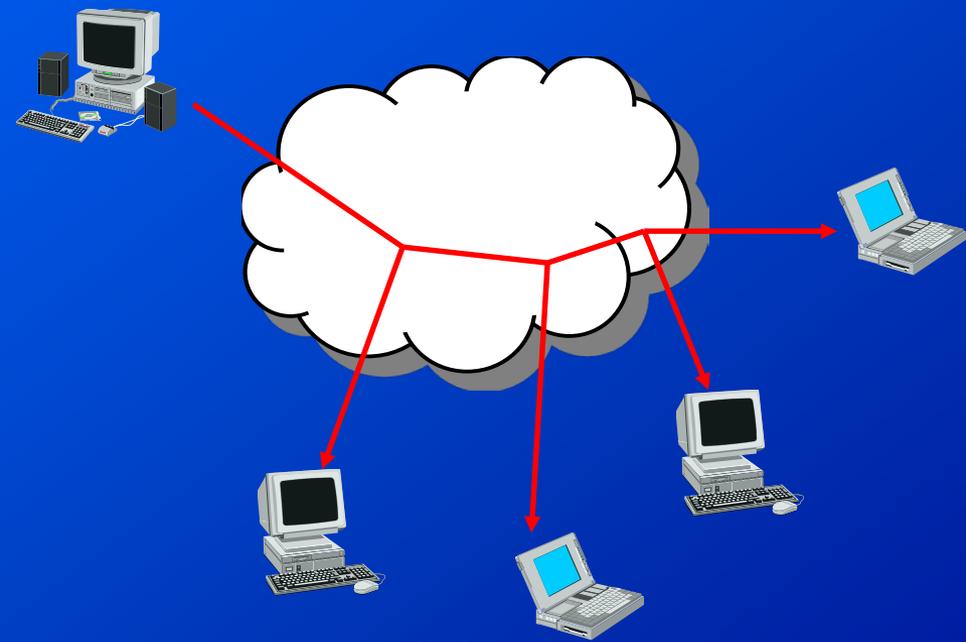
QoSマッピングを利用した 統合化資源割当制御

- アプリケーションQoSを介することによりエンドシステムQoSとネットワークQoSの関係が明らかになる
- 同じアプリケーションQoSを提供する場合にも...
 - エンドシステムが頑張ればネットワークが楽になる
 - ネットワークに余裕があればエンドシステムが楽になる



ヘテロニアス動画像マルチキャストへの 適用例

- それぞれ置かれる環境の異なるクライアントに、
利用可能な資源の範囲内で
できるだけ高品質な動画像を
マルチキャスト配信する



ヘテロジニアス動画像マルチキャスト

1. CPU資源量(エンドシステムQoS), アクセスリンク容量(ネットワークQoS)によるグループ分け
2. システム全体の効用(Utility)最適化

$$U = \sum_i^k U_i \rightarrow U_i = \frac{B_i}{P_i} \rightarrow B_i = q_i m_i$$

$$P_i = \zeta (P_i^B)^2 + \eta (P_i^S)^2 + \theta (P_i^C)^2$$

$$P_i^B = n_i \frac{B_i}{B_i^{free}} \quad P_i^S = \frac{S_i}{S_i^{free}} \quad P_i^C = \frac{1}{m_i} \sum_j \frac{C_i}{C_{ij}^{free}}$$

効用 = 利得 / コスト

利得 = ユーザQoS

コスト = エンドシステムQoS, ネットワークQoS

数値例

- 3つのマルチキャストグループ
- 同じエンドシステムQoS(サーバCPU処理能力), ネットワークQoS(帯域)を配分した場合

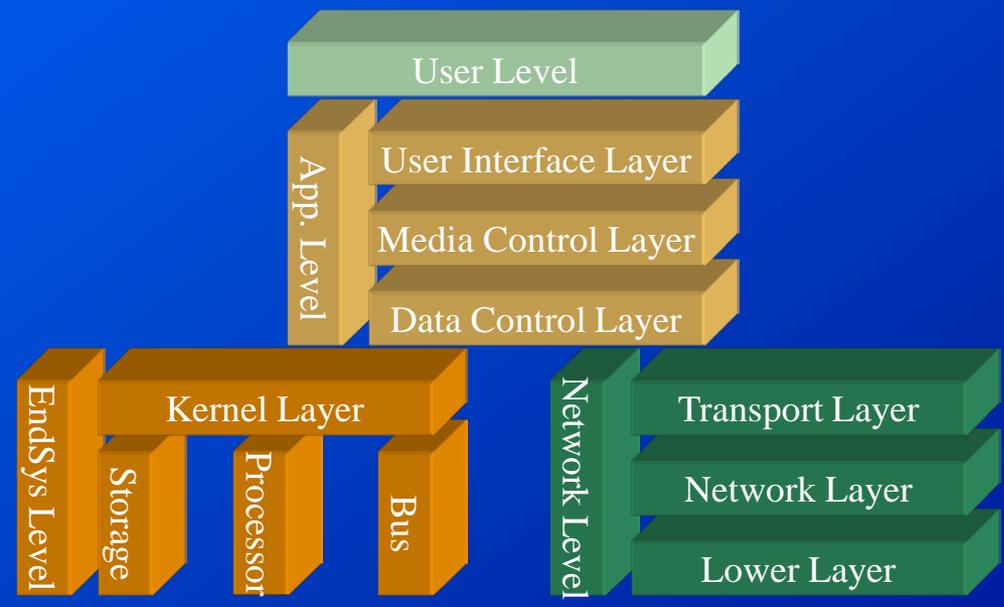
19.42 19.42 18.93 [dB]

- ユーザQoSを最大化する最適割当を行った場合

25.71 21.59 20.68 [dB]

さまざまなQoSマッピング

- レイヤ間のQoSマッピング
Intserv/ATM, Diffserv/ATM
- システム間のQoSマッピング
Intserv - Diffserv



まとめ

- ユーザ不在のシステム制御, サービス提供からの脱却
- QoSマッピングを利用した高度なQoS制御
- QoSマッピングは万能ではない