

# ネーミングシステムを用いた透過的サービス合成システム

齊藤 昭<sup>†</sup>， 南 正輝<sup>‡</sup>， 森川 博之<sup>‡</sup>， 青山 友紀<sup>†</sup>

<sup>†</sup> 東京大学大学院 情報理工学研究所

<sup>‡</sup> 東京大学大学院 新領域創成科学研究科

## [システムの目的]

本システムは Connectivity, Computing, Contents が遍在するユビキタスな環境において、機能オブジェクトをネーミングシステムによって統合することでサービスを合成するアーキテクチャである STONE(Service synthesizer On the NEt)のコンセプトを例証するために実装されている。本システムは、名前解決とデータ転送を統合することにより、名前に基づいたデータ配送を、透過性をサポートしつつ、コンテキストに応じた形態で行うことができる。

## [システム概要]

デモシステムの概要を図 1 に示す。デモシステムは、映像入力機能オブジェクトであるカメラと、映像出力機能オブジェクトであるモニタ、および、これらを統合するネーミングシステムであるサービスリゾルバ SR(Service Resolver)から成る。

カメラからキャプチャされた画像は SR へ転送され、SR は機能オブジェクトの名前に基づいて適切なモニタへと画像を転送する。モニタはインターフェースで集約された名前で指定され、SR が実際にどのモニタに画像を出力するかを選ぶ。この処理は動画のフレーム毎に行なわれ、動画を適切なモニタへと表示される。

SR はフレーム毎に出力先のモニタを選ぶために、ユーザが移動したときなど出力先を変更したい場合にもスムーズに対応することができる。

STONE においては、機能オブジェクトのインターフェースを名前として利用する。本デモシステムではモニタのみが SR に登録されており、その名前は、

Jpeg でエンコードされた Video を受け取るモニタを

求めていることから Input/Video/Jpeg のようになる。

実際にどのモニタを使用するかをユーザが指定することはなく、モニタを表す Input/Video/Jpeg という名前のみで指定する。サービスは機能オブジェクトの接続関係を記した SG(Service Graph)によって記述されるが SG には機能オブジェクトが「名前」で記述され、ユーザが SG を SR へ送信すると、SR は「名前」に対応する適切な機能オブジェクトを発見、接続し、サービスが合成される。すなわちこのデモシステムの場合ではカメラ側は、SG 中に自分の出力する動画を受け取るべきモニタを Input/Video/Jpeg という名前で記述する。SR は Input/Video/Jpeg にマッチする機能オブジェクトの中で適切なものを選び、データを送出する。(図 2)。ユーザが移動したときに出力先を変更するなど、コンテキスト適応な処理を行うためには、機能オブジェクト同士を直接つなぐのではなく、SR にデータを中継させる必要がある。これを STONE では On-the-fly モードと呼んでいる。On-the-fly モードでは機能オブジェクト同士を直接接続するのではないため障害発生時に別の機能オブジェクトを用いることで障害を隠蔽したり、上記のようにサービスのハンドオフを行ったりすることが可能となる。反面、SR がデータを中継する必要があるため、SR の負荷は非常に高くなってしまう。

On-the-fly モードにおいて、データは SG に付随して転送される。コンテキスト適応な処理を行うための情報の取り扱いに関してはアプリケーション依存となっており、このデモシステムの場合は位置情報などを用いている。

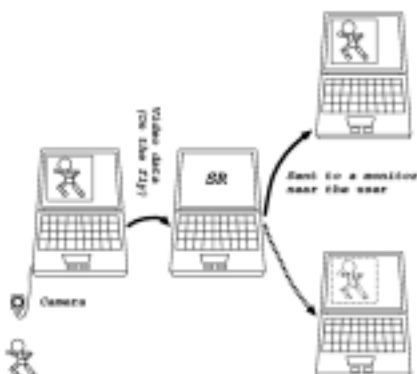


図 1: システム概要

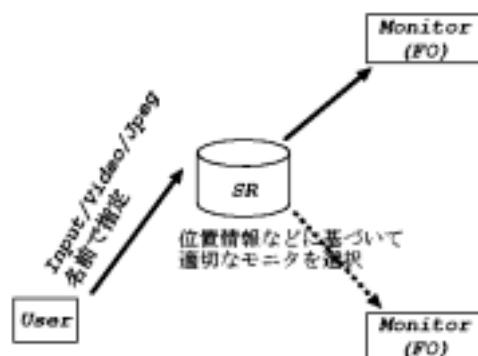


図 2: 名前に基づくデータ転送