

## 分散型マルチメディアシステム Symphony における QoS 指定可能なシナリオ記述言語

藤川 和利<sup>†</sup> 城島 貴弘<sup>††</sup> 寺西 裕一<sup>††1</sup> 松浦 敏雄<sup>††2</sup> 下條 真司<sup>††</sup>

<sup>†</sup> 奈良先端科学技術大学院大学 情報科学研究科

<sup>††</sup> 大阪大学 基礎工学部 情報工学科

遠隔会議システム等の分散型のマルチメディアアプリケーションでは、ネットワークの負荷等の環境条件に影響を受け、情報の品質が劣化し、必要な情報がやりとりされない場合がある。このような問題に対応するため、転送される情報の品質を意図的に変更する QoS 制御技術の研究が行なわれてきている。マルチメディアのプレゼンテーションの場合、必要な情報を提供するためには、場面に応じて QoS 制御法を指定する必要があるが、従来アプリケーションにおいて指定をどのように行なうかは、十分に検討がなされていなかった。そこで、本研究では、マルチメディアプレゼンテーションのための新たなシナリオ記述言語“MSDL-I”を提案する。

## Scenario Description Language with QoS Specification in A Distributed Multimedia System “Symphony”

Kazutoshi FUJIKAWA<sup>†</sup> Takahiro SHIROSHIMA<sup>††</sup> Yuuichi TERANISHI<sup>††3</sup>  
Toshio MATSUURA<sup>††4</sup> Shinji SHIMOJO<sup>††</sup>

<sup>†</sup> Graduate School of Information Science  
Nara Institute of Science and Technology

<sup>††</sup> Department of Information and Computer Sciences  
Faculty of Engineering Science, Osaka University

Current distributed multimedia application may lose its intention, if it can't keep a certain level of quality of contents by the environmental condition such as available network bandwidth. Recently, QoS control mechanisms have proposed to cope with the above problem by changing quality intentionally. In multimedia presentation, it is required that proper QoS control method is specified along with each scene. However, it is not sufficiently discussed how to deal with such specification. In this paper, we propose a new scenario description language for multimedia presentation, called “MSDL-I” which can specify QoS control method along with each scene.

<sup>3</sup>Currently, Nippon Telegraph and Telephone Corporation

<sup>4</sup>Currently, Faculty of Human Life Science, Osaka City University

<sup>1</sup>現在, 日本電信電話株式会社

<sup>2</sup>現在, 大阪市立大学 生活科学部

## 1 はじめに

近年、ATM[10]に代表されるような高速通信ネットワークの実用化により、マルチメディア情報の転送が容易になってきている。また、WWW(World Wide Web) [11] や gopher[1] 等の情報提供・検索システムの普及により、マルチメディア情報をネットワーク上に提供したり、提供されているマルチメディア情報を探索・検索して取得することが可能となってきた。こうした背景のもと、ネットワーク上に分散したマルチメディア情報を組み合わせて提供する分散型マルチメディアプレゼンテーションシステムが注目されている。

プレゼンテーションシステムにおいては、利用者の環境条件にかかわらず、情報提供者の作成した通りのプレゼンテーションを行いたいという要求がある。しかし、利用可能なネットワーク帯域やCPU、メモリ、特殊デバイスの有無等の計算機資源の制限により、音が途切れるなど十分な音質を実現できなかったり、圧縮された映像の伸長処理が間に合わず、映像の表示間隔の遅れや揺らぎ(ジッタ)が生じてしまい、記述通りのプレゼンテーションが行えない、あるいは、十分な表示品質を得られない場合がある。

これまでに、これら実行環境の制約に対処するための方法として、実行環境に応じて処理可能な程度にまで映像のフレームレートを落したり、解像度等メディアの品質の調節を行なうメディアスケーリング技術 [3, 6, 9] や、ネットワークの帯域の確保やOSのスケジューリングの調整を行なうことでマルチメディア情報の品質を保証する QoS(Quality of Service) 保証技術 [8] の研究がなされている。ここでは、メディアスケーリング及び QoS の保証を合わせて QoS 制御と呼ぶ。

ビデオ・オン・デマンド・システム QtPlay[8] やテレビ会議システム nv 等では、マルチメディア情報の表示の際、必要に応じて利用者が QoS 制御の指定を行なっている。これを“ユーザ制御モデル”と呼ぶ。しかし、プレゼンテーションシステムでは、利用者が実行時に各場面毎に必要なと思われる QoS 制御を指定することは困難である。また、利用者が QoS 制御を行なうと、情報提供者側のプレゼンテーションに対する意図が正しく反映できない可能性がある。一方、MODE[2]、HeiTS[3]、AMF[4] 等では、QoS 制御方法を提案しているが、その指定は固定、もしくはシステムが自動で選択して行なう。これを“システム制御モデル”と呼ぶ。この方法では、内容・場面に応じて QoS 制御を行なうといった要求

に対応できない。

プレゼンテーションにおいて、シナリオ記述者(オーサと呼ぶ)の意図を損なわないようにするためには、オーサがシナリオ記述として QoS 制御を指定することが有効と考えられる。この場合、利用者が QoS 制御を実行時に設定する必要がなくなる。

そこで、本研究では、QoS 制御の指定可能なシナリオ記述言語“MSDL-I”を提案する。また、MSDL-Iの有効性を検証するために、QoS 制御機構をもつ分散型マルチメディアプレゼンテーションシステム Symphony をMSDL-Iの処理系として試作した [7, 12, 13]。

以下、2章では、ここで提案するシナリオによる QoS 制御の有効性とその問題点について論じる。3章では、提案するシナリオモデルに基づいて定義したシナリオ記述言語MSDL-Iについて述べる。4章では、MSDL-I 処理系である Symphony システムについて説明する。

## 2 シナリオによる QoS 制御

本章では、マルチメディアプレゼンテーションにおいて QoS 制御の指定をどのようにに扱うべきかを考察し、ここで提案するシナリオによる QoS 制御指定を扱うモデルの有効性、及びその問題点について論じる。

### 2.1 QoS 制御に関する要求

ここでは、マルチメディア情報を用いたプレゼンテーションでは、どのような QoS 制御が要求されるかを挙げる。

マルチメディア情報の品質を決定する要素は、メディア内 QoS 要素とメディア間 QoS 要素の二つに分類できる。メディア内 QoS 要素を表1に、メディア間 QoS 要素を表2に挙げる。

QoS 要素をどのように保証すべきか、いかに調整すべきかといった QoS 制御の要求は、表示するマルチメディア情報の内容によって異なる。また、同じ情報でもその情報が利用される状況に応じて要求される QoS 制御が異なる場合がある。さらに、要求される QoS の要素が実現できない場合の制御として、映像をアニメーションに切替えたり、音声テキストに変えたりといった具合に、“代替メディア”を用いることが考えられる。

以上のことより、マルチメディアプレゼンテーションでは、

- メディア内、メディア間の QoS 要素をどれだ

表 1: メディア内 QoS 要素

要素	意味
遅延時間	シナリオで指定された表示されるべき時刻と、実際に表示された時刻の差を表す。
時間解像度	映像の場合はフレームレートを表す。
空間解像度	映像では画面の解像度や画面表示の大きさを表し、音声ではサンプルビット数を表す。
ジッタ	映像やアニメーション等における個々のフレームの表示間隔のばらつきを示す。

表 2: メディア間 QoS 要素

要素	意味
遅延時間	メディア内 QoS 要素における遅延時間と同じ意味である。
同期のずれ	同期が要求される複数の情報間での表示のずれを表す。
同期の方法	同期がずれる場合に、どのように制御して同期を保つかを指定する。

け実現すべきか

- 必要な QoS が得られない場合、代替メディアによる対処を含め、どう制御していくのが望ましいか

といった QoS 制御に関する指定が必要である。

## 2.2 シナリオによる QoS 制御の有効性

本研究では、プレゼンテーションに有効なモデルとして、シナリオのオーサが QoS 制御を指定するモデル“シナリオ制御モデル”を提案する。このモデルに基づくシナリオでは、プレゼンテーションの各場面に応じた QoS 制御が記述されている。

シナリオ制御モデルでは、シナリオ記述に従って処理系が QoS 制御を行なうため、利用者が場面の進行に伴って QoS 制御を設定し直すといった操作は必要ない。

## 2.3 シナリオによる QoS 制御の問題点

QoS 制御の指定をシナリオにおいて可能とする場合、以下の問題点が挙げられる。

- 記述の混在  
QoS 制御はプレゼンテーションの進行に伴って行なう必要がある。従って、プレゼンテーションの進行指定のための記述と QoS 制御の指定のための記述を用意し、それらを合わせてプレゼンテーションを構成することが考えられる。シナリオ記述には、プレゼンテーションの進行が容易に把握できることが望まれるが、この場合プレゼンテーションの進行記述と QoS 制御の記述が混在して、シナリオ本来の進行が不明瞭になってしまう。
- 実行環境への依存  
希望する QoS を実現できない場合、対処方法を手続き的に直接指定することが考えられる。この時、デバイスの有無や能力、ネットワークの負荷などは、各プレゼンテーション実行環境に応じて異なるため、適切な対処を記述するためには、全ての状況を把握した上でシナリオを記述していく必要がある。しかし、こうしたシステムに依存した要素の記述はシナリオを複雑化・肥大化させる原因となる。
- 複数動作の扱い  
メディア間の QoS 要素の制御は、実際にプレゼンテーションにおいて同時に表示される複数の素材情報の集合や一連の場面に対してなされる必要がある。こうした複数の動作に対する QoS 制御の指定をいかに簡潔に扱うかも問題である。

## 3 シナリオ記述言語 MSDL-I

前章で挙げた要求・問題点を考慮して、本章ではシナリオ記述言語 MSDL-I<sup>1</sup>を提案する。以下では、MSDL-I の基本方針について説明し、さらに、記述例を挙げる。

### 3.1 設計方針

シナリオの進行と QoS 制御の記述が混在を避けるため、MSDL-I では、構成要素をプレゼンテーションの素材情報を定義する“オブジェクト記述”とシナリオの進行を表現する“シナリオ進行記述”の二

<sup>1</sup>Multimedia Scenario Description Language version 1 の略

つに分け、オブジェクト記述において、QoS 制御の指定を含んだオブジェクトとして素材情報を定義し、シナリオ進行記述ではオブジェクトの表示順序だけを指定する。

また、一つの素材情報として、QoS 制御の異なる複数の候補 (オブジェクト) を指定することを可能にする。これにより、細かい条件を記述することなく、実行環境に応じて、プレゼンテーションシステムが適切な候補を選ぶことができる。

さらに、一つのシーン内の複数の素材情報に対して、まとめて QoS 制御の指定が行なえるよう、素材情報の組み合わせであるシーンを一つの素材情報として扱えるようにする。

MSDL-I では、以上のような指定を行なうため、単独のマルチメディア情報に対して QoS 制御を指定できる Scaled 型オブジェクト、代替メディアの候補を指定できる Alternative 型オブジェクト、他のシナリオを指定できる Composite 型オブジェクトの 3 種類のオブジェクトが用意されている。

オブジェクトには、ネットワーク上に分散するマルチメディア情報を指定するため、属性の一つとして、Location があり、値として URL<sup>2</sup> を指定する。Scaled 型オブジェクトでは、QoS の指定を行なうために、以下に説明するような属性が用意されている [7]。

TemporalCrop は、オブジェクトそのものもつ時間解像度がどれだけ間引かれることを許すかを示し、SpatialCrop は、オブジェクトそのものもつ空間解像度がどれだけ低下することを許すかを示す。これらは、元の解像度に対してどれだけの割合で品質を下げた再生するかを表す。つまり、0~100% の範囲で値をとり、0 ならば品質を全く下げないことを意味する。Latency は、情報が実際に表示されるまでの準備にかかる時間を表し、Continuity は、情報の表示間隔の時間の差を意味する。SyncLag は、許される複数の情報間での表示のずれの時間を表す。これら 3 つは、値として秒数をとる。SyncType では、設定された QoS 要素の値内での調整方法を表現する。これには実時間再生を優先する Realtime、表示の連続性を優先する Motion、音声の再生を優先する Sound の 3 つがある [5]。

また、オブジェクトを同時に表示する場合、すべてが同じ優先度をもつとは限らない。MSDL-I では、こうした場合に対応するために、オブジェクトどう

<sup>2</sup>URL は、RFC1738 で規定されており、WWW 等でも用いられている一般的な識別子である。

```

1 Title: "サンプルシナリオ",
2 Object: %% オブジェクト記述
3 { alternative (Name: "picture",
4   Candidate: obj (
5     Location: "cmt://.../v2.mpg",
6     TemporalCrop: 50,
7     SpatialCrop: 0,
8     SyncType: Realtime),
9   Candidate: obj (
10    Location: "http://.../v2.mpg",
11    TemporalCrop: 80,
12    SpatialCrop: 20,
13    SyncType: Realtime)),
14  alternative (Name: "explanation",
15    Candidate: obj (
16      Location: "cmt://.../v2.au",
17      TemporalCrop: 40,
18      Latency: 100,
19      SyncType: Sound),
20    Candidate: obj (
21      Location: "http://.../subtitle.msd",
22      SyncType: Realtime))),
23 Preference: explanation > picture,
24 Behavior: %% シナリオ進行記述
25 [[picture <-put(At: 300@300)];
26 [explanation <-put(At: 400@400)];
27 [picture<-play() | explanation<-play() ]

```

図 1: 記述例

しの優先順位を指定可能としている。

### 3.2 記述例

以上のように定義した MSDL-I を使った簡単な例を図 1 に示す<sup>3</sup>。

この例では、あるニュース番組の内容を表現している。2 行目から 22 行目にかけて、二つの代替メディアをもつオブジェクト (Alternative 型) を定義している。picture という名前のオブジェクトと explanation という名前のオブジェクトである。

picture は、MPEG 形式の映像である。この例では、同じ MPEG データで QoS 要素の値の異なる候補を二つ設定することで、一つの MPEG データの QoS 制御の方法が指定されている。すなわち、第一候補の QoS 要素を守るよう実行できない場合

<sup>3</sup>例では紙面の都合上、URL の内容を一部省略している。

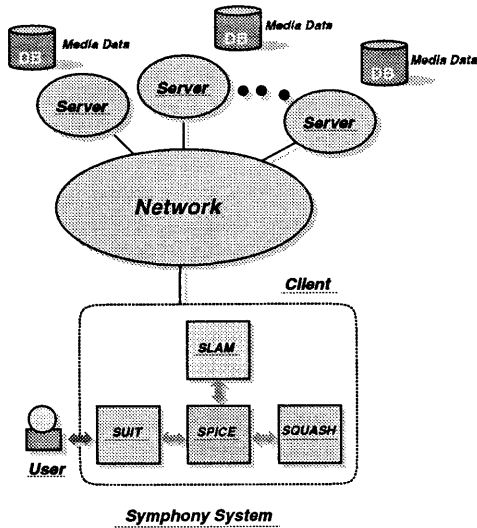


図 2: Symphony システムの構成

には第二候補の値まで調整してよいことを示している。explanation は、音声情報を第一候補としてもち、同じ内容のアニメーションによる字幕を第二候補として持つ。

23 行目に、picture と explanation の優先順位が指定されている。この例では explanation のほうが優先順位が高く、picture の優先順位が低く設定されている。

24 行目～27 行目が、このシナリオの進行、すなわち、空間的にどう配置するか、時間的にどう変化させるかが記述されている。この記述では、picture と explanation が画面上に置かれ、それらの内容が並列に再生されるという内容となっている。

このシナリオの場合、十分なネットワーク帯域・CPU 能力・デバイスがある環境では、高品質の映像と音声再生される。しかし、十分な実行環境ではない場合、映像の品質が落ち、音声はアニメーションに替わり、紙芝居のような映像と時間的に変化する字幕による同じ内容のプレゼンテーションとなる。

## 4 MSDL-I 処理系の実現

MSDL-I の処理系を Symphony のシナリオ解析・実行系として実現する。本章では、まず Symphony の概要を述べ Symphony の動作手順を説明する。

### 4.1 Symphony の概要

Symphony の構成を図 2 に示す。Symphony において映像・音声といった時系列メディアを転送する

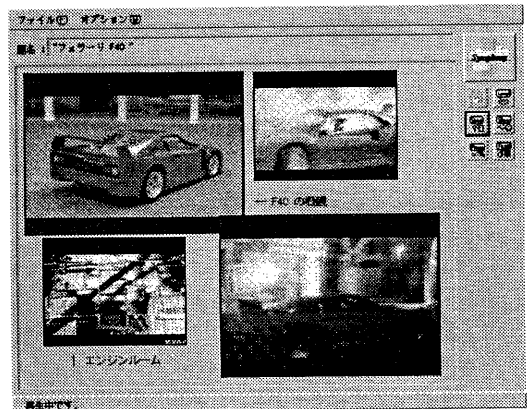


図 3: Symphony の画面例

サーバは、メディアスケール機能を持つサーバとして独自に作成したものである。利用者は、クライアントからサーバ上にあるシナリオを指定し、クライアント上でプレゼンテーションを得る。

Symphony のクライアントは、ユーザインタフェース部 (SUIT)、プレゼンテーション制御部 (SPICE)、言語解析部 (SLAM)、QoS 制御部 (SQUASH) の 4 つの基本要素から構成される。

Symphony のプロトタイプシステムは、プラットフォームとして SPARCstation 10、OS として Solaris2.3 を用い、X ウィンドウ Toolkit ライブラリである Motif のアプリケーションとして実現している (図 3)。各機能要素の実装は、並列的な処理を制御するために、マルチスレッド機構を利用している。

### 4.2 Symphony の動作

Symphony では、まず、表示や転送を開始する前にクライアントの実行環境において、シナリオに指定された QoS 要素の値の通り、オブジェクトが実行可能かどうかを判別する。この時、オブジェクト毎に SPICE から SQUASH に対して QoS 見積り要求が出される。SQUASH では、サーバに記述されている QoS 要素の値、クライアントの環境情報、過去の表示実績をもとに、実現される QoS 要素の値を見積もる。

SPICE は、得られた QoS 要素の値の見積もりと各オブジェクトに指定された QoS 要素の値とを比較し、指定された QoS 要素の値でオブジェクトの再生を実現可能かどうかを検証する。Alternative 型

の場合、指定された QoS 要素の値を実現できない場合は、次の候補に対して検証を行なう。

サーバ上にある素材情報を用いたプレゼンテーションでは、表示するために実際にサーバからクライアント上へ情報を転送しなければならない。この時、シナリオに含まれる全てのオブジェクトの情報の転送を終了してから各オブジェクトの表示を開始したのでは、遅延時間が大きくなるため、シナリオに指定された遅延時間に従って、転送が終了した情報を表示する。転送が終了していない素材情報は、プレゼンテーション中にシナリオ進行の進み具合に従って、順次前もって転送を行なう。

プレゼンテーション中では、SQUASH が指定された QoS 要素の値を守るようモニタしながらオブジェクトの転送、同期処理を進める。表示中に QoS 要素の値を守れない状態になると、SPICE に対してそれを知らせる。SPICE は、オブジェクトが Alternative 型であれば次の候補への変換操作を行なうが、そうでない場合、利用者にその旨を伝えるよう SUIT に命令して、転送・表示を続行する。

## 5 あとがき

本研究では、従来のシナリオの機能に加えて、QoS 制御の指定を扱えるシナリオ記述言語“MSDL-I”を提案し、その処理系を試作した。

シナリオ記述者は MSDL-I を用いることにより、シナリオの進行、及び QoS 制御を記述することができる。MSDL-I は、利用者が QoS 制御を意識すること無く、実行環境に応じたプレゼンテーションを得ることができる枠組を提供している。

今後の課題として、オーサリングツールの構築、シナリオ部品の再利用といった問題点について検討していくとともに、処理系の分散化といったシステム構築面での検討を進め、より効率良く処理を進めるマルチメディアプレゼンテーション環境としてシステムを実現していくことが挙げられる。

## 参考文献

- [1] M.McCahill. The Internet Gopher Protocol: A Distributed Server Information System. *Conneziuous — The Interoperability Report*, Vol. 6, No. 7, July 1992.
- [2] G.Blakowski, J.Hubel, and U.Langrehr. Tool support for the synchronization and presentation of distributed multimedia. *Computer Communications*, Vol. 15, No. 12, December 1992.
- [3] L.Delgrossi, C.Halstrick, D.Hehmann, R.G.Herrtwich, O.Krone, J.Sandvoss, and C.Vogt. Media Scaling for Audiovisual Communication with the Heidelberg Transport System. *Proceedings of ACM Multimedia '93*, pp. pp99-104, August 1993.
- [4] G.von Rossum, J.Jansen, S.Mullender, and D.C.A.Bulterman. CMIFed: A Presentation Environment for Portable Hypermedia Documents. *Proceedings of ACM Multimedia '93*, pp. pp183-188, August 1993. Anaheim.
- [5] 藤川和利, 下條真司, 松浦敏雄, 西尾章治郎, 宮原秀夫. 分散型ハイパメディアシステム Harmony における情報同期機構の実現. 電子情報通信学会論文誌, D-I, Vol. J76-D-I, pp. pp.473-483, September 1993.
- [6] 尾上裕子, 藤井敬三, 徳田英幸. マルチキャストサーバにおける動的 QoS 制御. 情報処理学会, マルチメディア通信と分散処理 66-1, pp. pp1-6, June 1994.
- [7] 寺西裕一, 島村栄, 松浦敏雄, 下條真司, 谷口健一. QoS を考慮したマルチメディアシナリオ記述言語. 情報処理学会, 情報メディア 18-1, pp. pp.1-8, September 1994.
- [8] 中島達夫, 手塚宏史, 藤田浩. 連続メディア処理のための動的 QoS 管理方式. 情報処理学会 第 49 回 (平成 6 年後期) 全国大会, 7R-02, 3, September 1994.
- [9] 手塚宏史, 中島達夫. ビデオデータの滑らかな表示に関する考察. 日本ソフトウェア科学会 第 11 回大会論文集, pp. pp.245-248, October 1994.
- [10] R.O.Onvural. *ASYNCRONOUS TRANSFER MODE NETWORKS: Performance Issues*. Artech House, 1994.
- [11] T.Berners-Lee. HyperText Transfer Protocol - HTTP/1.0. Internet Draft, December 1994. Expires: June 19, 1995.
- [12] K.Fujikawa, S.Shimojo, H.Shimamura, Y.Teranishi, T.Matsuura, and H.Miyahara. Application level qos modeling for a distributed multimedia system. In *to appear Proceedings of 1995 Pacific Workshop on Distributed Multimedia Systems*, March 1995. Honolulu HI. USA.
- [13] 呉世雄, 島村栄, 椿野真治, 藤川和利, 下條真司, 松浦敏雄, 宮原秀夫. 分散型マルチメディアシステム Symphony における品質保証機構. 情報処理学会, マルチメディア通信と分散処理 70-7, pp. pp39-44, May 1995.