ストレージ複製管理のためのアクセス履歴とデータライフサイクル情報利用

Replication Control based on Explicit Lifecycle Information and Access History

小林 大♣ 田口 亮♢ 横田 治夫♠

Dai KOBAYASHI Ryo TAGUCHI
Haruo YOKOTA

情報爆発時代の高度なデータ管理要求を実現するため，ストレージシステムは大規模化，複雑化しており，その管理コストの上昇が問題となっている．特にデータを扱う利用者，アプリケーション，システム管理者が，要求仕様を記述する際，システムの内部動作の記述を強制されるのは非常にコストが高い．一方で，ストレージシステムが自律動作を行う上で，利用履歴のみから性能要求を満たすための自律管理に必要な情報を推定・取得することは困難である．

本稿では，近年注目されているデータのライフサイクル情報に着目し，ユーザあるいはアプリケーションソフトウェアからメタデータとして提供される，コンテンツに関するライフサイクル情報と，ストレージシステム側で取得している利用履歴を元に，ストレージ QoS が可能な自律ストレージシステムについて考察する．また，本稿では明示的ライフサイクル情報を，要求性能維持のための複製データ配置管理に応用したコンセプト適用事例により有用性を明確化する．

Neglecting increasing storage management cost becomes hindering stable development of the information society in information explosion era. In particular, the cost to describe complex storage management policy is burden to the users and administrators because current policy must include storage system-level behavior, while there is just so much the systems can do using only access pattern analysis. We focus on lifecycle information of each content attached by users or applications for information lifecycle management. In this paper, we consider storage QoS management using content lifecycle information generated by combining application-given abstract lifecycle information in the metadata of each content with access trend information in storage management modules. We illustrate the efficiency of this concept with an example that is replica location management using lifecycle information for keep the quality of access latency.

1. はじめに

近年，ストレージ上のデータ量の爆発的な増加に伴い，効率的なデータ管理に対する要求が高まっている．

スループット，レイテンシ等の性能保障やデータ信頼性・可用性の維持のため様々な管理機能を併用し，ユーザから求められるサービス品質 (QoS) を保証する必要がある．現在の大規模なストレージシステムでは，多様な管理機能を把握しポリシーを記述することは難しく，ストレージ管理コスト上昇の大きな要因となっている．

管理コスト軽減のため，ストレージシステムに対し計算資源を付加し，自律的な管理を行う自律ストレージシステムが様々提案されている．その上で自律的な管理の精度を向上させるため，システム内資源の利用傾向から自律管理動作を起動させる仕組みが様々提案されている．しかし，利用傾向から逸脱したアクセスのあった場合の QoS 管理は依然難しい問題である．

一方，近年注目されているのが情報ライフサイクルマネジメント（Information Lifecycle Management:ILM）[1] である．これは，データが生成されてから，活用され，破棄される一連の流れの中での利用状態の変化に着目し，機密情報や法的制限を受けたデータの保存・破棄や，各データの利用頻度の変化を利用したストレージ管理を行うものである．

また，我々は以前より，多様なコンテンツごとの粒度の細かい情報ライフサイクル管理を行うための基盤として，ECA ルールによって分散ストレージのデータ管理を記述するアーキテクチャを提案している [2]．これまで，多量に付与された管理ルールを効率よく起動，処理するルール評価システムを実現しているが，多種多量のルールを効率よく記述する方法はいまだ重要な課題である．

そこで我々は，データ利用側から与えられるライフサイクル情報をストレージシステムが収集し，利用傾向情報と併せることで，コストが低く精度が高いストレージ管理ルールの生成を実現することを考える．

本稿では，アプリケーションからのライフサイクル情報収集方式として，コンテンツ登録時にコンテンツ管理ルールとして，生成・利用・破棄といったデータの各利用状態への遷移タイミングを用いることを提案する．そして，与えられたコンテンツ管理ルールとストレージシステム側で保持している各コンテンツの利用傾向と併せ各コンテンツのライフサイクル情報として統合することで，ストレージ管理ルールを発行する枠組みを提案する．

また，その適用具体事例として，負荷分散のための複製データ生成破棄管理を考える．生成・利用・破棄の各状態への遷移タイミングがコンテンツ管理ルールとしてアプリケーション側からストレージシステムへ与えられた場合に，応答性能・格納容量共に効率の良い負荷分散管理が実現できることを実験により示す．

本稿の構成を以下に示す．2. で従来のストレージ管理機構の問題点を述べ，3. で我々が考えるストレージ管理機構のコンセプトを述べる．4. で並列ストレージの複製管理への応用を述べる．並列ストレージの複製管理への応用について，シミュレーションプログラムを用いた実験の結果を 5. で示す．最後に 6. でまとめと今後の課題について述べる．

2. 従来のストレージ管理

従来のストレージ管理の枠組みを図 1(a) に示す．

ユーザ，アプリケーションが利用するデータに対しサービス品質要件を要求したい場合，自身あるいはその代行となるシステム管理者が，利用するデータに求められるコンテンツ管理を実現するようなストレージ管理ルールを記述する必要があった．一方システム自身は格納データへのアクセスリクエストを監視し解析することで，ユーザから求められるコンテンツの性質とは独立に，個別にストレージ管理ルールを発行している．これは，ストレージシステムの構成・性質を熟知した上で適切なルールを記述する必要があり，ストレージ管理の負担が大きい．

ストレージ管理，特に QoS 制御を行う上で重要となるのがアクセス傾向の変化である．各アクセスに対して効率の良い資源配分を行うために，あらかじめ将来のワークロードを予測する必要がある．現在はデータ利用傾向に対しさまざまな解析手段を試みることで予測を行っているが，急激な傾向の変化への対応や，格納されるコンテンツに対する利用傾向の多種多様化を考慮可能な情報の利用は予測率向上に大きく寄与する．

♣ 学生会員 東京工業大学 大学院情報理工学研究科 計算工学専攻・日本学術振興会特別研究員 DC daik@de.cs.titech.ac.jp

♢ NHK 放送技術研究所 taguchi.r-cs@nhk.or.jp

♠ 正会員 東京工業大学 学術国際情報センター yokota@cs.titech.ac.jp

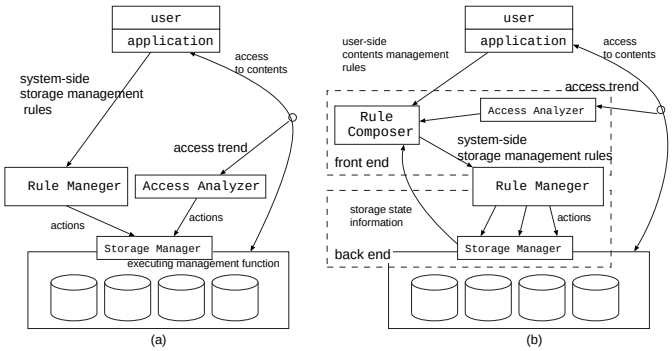

図 1: (a) 従来のストレージ管理 (b) 利用履歴とアプリケーションからの情報を統合したストレージ管理のコンセプト

Fig.1 Storage Management Architecture (a) current (b) proposal

3. 提案するストレージ管理機構

本節では，従来のストレージ管理機構の枠組みに対し我々の考えるストレージ管理機能発行の枠組みを述べ，比較を行う．

3.1 ライフサイクル情報を利用したコンテンツ管理ルール

近年注目されているのが情報ライフサイクルを利用した機密情報や法的制限を受けたデータの保存・破棄やストレージ管理である．ライフサイクルとはデータが生成されてから，活用され，破棄される一連の流れの中での利用状態の変化であり，これは将来のワークロード変化に関して，利用傾向解析とは直交した情報が得られる．

そこで本稿では，各コンテンツごとの明示的なライフサイクル情報を，コンテンツ管理ルールとして格納データに添付し利用することを考える．これによりコンテンツごとの特徴を活かした細粒度のコンテンツライフサイクル管理 [2] を実現するストレージ管理ルールを生成可能となる．

3.2 明示的情報と利用傾向との統合によるストレージ管理

従来の管理機構に対し，我々のアプローチを以下に示す．

従来のデータ利用者から直接ストレージ管理ルールを記述する複雑性を隠蔽するため，従来の管理機構をバックエンドとし，より抽象的な情報（コンテンツ管理ルール）からストレージ管理ルールを生成するフロントエンドを挿入したのが我々の考えるストレージ管理の枠組みである (図 1(b))．

コンセプトの根底にある仮定は (1) 「データの利用方法はアプリケーションやユーザが良く知っている」こと，(2)「データの格納方法はストレージシステムが良く知っている」である．そこで，ユーザやアプリケーションから，データの利用に関するコンテンツ管理ルールを入力させ，利用傾向解析によりその情報を洗練した後，現在のシステム状況に応じた，具体的なストレージ管理動作を含むストレージ管理ルールを発行する．

我々の提案するストレージ管理機構では，ユーザやアプリケーションは，ストレージ情報を意識しないコンテンツ管理ルールを自律ストレージシステム中の Rule Composer コンポーネントへ入力する．ここで，コンテンツ管理ルールには従来のコンテンツに対する性能保障記述に加え，各コンテンツごとの ILM をまた付加する．一方で，Access Analyzer は格納されたデータに対する利用傾向情報を採取し，統計・解析により得られた結果を同じく Rule Composer へ入力する．Rule Composer は上記の情報と，構成や使用状況等を含むシステムの現在状態に関する情報から，適切なストレージ管理ルールを作成し，Rule Manager へ発行する．

4. 複製を用いた負荷均衡化への適用

与えられたライフサイクル情報を利用した効率の良いストレージ管理ルールの生成を考える．ここでは複製を用いた応答性能保障における複製データの生成・破棄を起動するストレージ管理ルールを考える．

複数のディスクノードで構成されたネットワークストレージシステムにおいて，一部のストレージノードへアクセス負荷が集中することは性能低下を招き，応答性能保障を難しくする [3]．よって，システム構成ストレージノード間での負荷均衡化が重要である．データの複製を複数のノードに配置すること（レプリケーション）でアクセスリクエストをノード間に分散することが出来る．

4.1 これまでの問題

コンテンツごとの細かい複製配置指定をするために，各コンテンツごとにストレージ管理ルールを添付する管理手法 [2] を用いる．この場合，コンテンツ利用者の要望から次のようなストレージ管理ルールを発行し，ルールマネージャへ登録する:

```
rule[copy file1 to disk3 on July 13th]
rule[delete file1 on disk3 on July 14th]
```

しかし，ネットワークストレージでは各ストレージノードは半導体メモリによるキャッシュを備えていることが多く，レプリケーションによって同一のデータが複数のノード上のキャッシュに残ってしまいキャッシュヒット率を低下させるため，単純なレプリケーションは期待される性能維持効果を得られない [4]．そのためシステム構成に関する知識に乏しいアプリケーションらが直接ストレージ管理ルールを生成するのは難しい．

我々の以前の研究結果 [5] より，複製による負荷均衡化では低・中頻度利用データへのアクセスを複製間で回送するのが，各ノード上のキャッシュ利用効率を最大化するためには望ましいとの結果が得られている．しかし，多くの利用例では低・中頻度利用データは格納データの大部分を占める [6] ため，信頼性に必要な冗長性を超えた数の複製を常時作成することはディスク領域を大きく圧迫するため，好ましくない．

複製データの生成・破棄において重要なのは，必要な時点 (条件 1) のみ， 低・中頻度利用データ (条件 2) の複製を，作成することである．

4.2 ライフサイクルに基づく生成破棄管理

提案するストレージ管理の場合，ユーザやアプリケーションはコンテンツ管理ルールとしてそのコンテンツの性能要件や，ILM 情報を作成し，Rule Composer へ登録する．次のルールはその一例である：

```
rule[file1 is used from July 13th to 14th]
rule[file1 requires 30Mbps when used]
```

Rule Composer では与えられたストレージ管理ルールと，ストレージシステム構成情報，各コンテンツの利用状況から，実際に複製を生成・破棄するストレージ管理ルールを作成し，Rule Manager に登録する．負荷均衡化のための複製管理では 4.1 で述べた特徴を考慮した管理ルールを生成する必要がある．よって以下では，コンテンツ管理ルールとして与えられるライフサイクル情報より条件 1 のための必要な時点を取得し，与えられたライフサイクルと各コンテンツの利用頻度情報から条件 2 のための相対的な利用頻度を推測することで複製生成・破棄を行う手法について考察する．

4.2.1 コンテンツ生成

ユーザ・アプリケーションは自身の扱うデータの利用率上昇タイミング及び利用率収束タイミングに関する知識を備えていると仮定する．ユーザ・アプリケーションよりデータ生成時に，コンテンツ管理ルールとして，利用率上昇および利用率収束を予測した情報（ライフサイクル情報）を格納する．ストレージシステムはアクセス要求に対しデータを提供すると同時に利用傾向を監視し，メタデータあるいは管理機構内に保存する．そして，次節のような複製生成・破棄ストレージ管理ルールを生成する．

4.2.2 複製生成

あるコンテンツ A について，ライフサイクル情報である利用開始時点として与えられたタイミングで，ストレージシステムは A に関するアクセス傾向を取り出し，A の利用頻度クラスを決定する．決定された頻度が低または中であった場合に，ストレージシステムは A の複製を他のディスクに作成するストレージ管理ルールを発行する．

4.2.3 複製破棄

A に関する利用終了時点として与えられたタイミングで， 生成時と同様にストレージシステムは A に関するアクセス傾向を取り出す．過去のアクセスピークと比較し，現在の利用頻度が十分収束していると確認できた場合，ストレージシステムは A の複製を破棄するストレージ管理ルールを発行する．

5. 実験

前節により述べた例のうち，複製管理手法の挙動についてネットワーク接続された HDD 動作を模擬するシミュレーションプログラム上に実装しその結果を観察した．

5.1 実験概要

実験では，シミュレーションプログラム上に構成されたシステムに対し，web アクセスを模したアクセスパターンの READ リクエストを投入し，その最大レスポンスタイム（レイテンシ）を記録する．

測定対象とするシステムは，レプリケーションを行わないシステム (normal)，利用頻度に依らず全てのデータに対しライフサイクル情報を元にレプリケーションを行うシステム (rep_all)，そして，4.2 で述べたように各コンテンツの利用頻度とライフサイクル情報を併用し，低・中頻度利用データのみを対象としたストレージ管理ルールを発行するシステム (ilm+access) の三種類とした．

ここで，本来はユーザまたはアプリケーションによって付与されるライフサイクル情報については投入するワークロードから事前に以下のように機械的に作成したものを用いた．各コンテンツごとに 1 シミュレーション時間ごとのアクセス数から最もアクセスが集中するピーク負荷を算出した．つづいて，ピーク負荷値に対して 1/5 のトラフィックを記録する最初の時間を利用開始タイミング，1/2 のトラフィックを記録する最後の時間を利用終了タイミングとした．なお，ここで定義したライフサイクル情報を変化させた場合の実験を 5.5 に記す．

また，中頻度利用アクセスは，利用開始タイミングの 1 シミュレーション時間あたりの読み出しリクエストが 20 以下のものとした．これを満たすファイルは格納ファイル数の 97%を占める．

以上を異なる頻度のワークロード下で行い，それぞれの実験における最大レイテンシと，増加ファイル数を評価の尺度とした．

5.2 実験環境

待ち行列を利用したイベントドリブンのストレージシミュレーションプログラムを構築し実験に用いた．ディスクによるサービス時間については [7] を基にした表 1 にしめすパラメータと前回アクセス時のヘッド位置により算出した．シミュレーション内のシステム構成を表に示す．またノードネットワーク性能は 300Mbps とし，クライアント側ネットワーク性能は無制限とした．

シミュレーションで用いるシステム負荷は，FIFA WorldCup98 Official WEB サイトのアクセスログ [8] から抽出した．詳細を表 1 に示す．シミュレーション実行時間削減のためリクエスト数を 1/80 へ無作為に削減している．また高いシステム負荷を実現するために，ファイルサイズについて加工をし合計格納サイズを 2GByte としている．ワークロード傾向の詳細については [9] を参考にされたい．異なる負荷のワークロードを実現するため，シミュレーション時間中のアクセスリクエストの発行間隔を実際のログより縮小したワークロードにより実験を行った．この縮小度合いをワークロードの負荷大小を示す指標とする．

表 1: シミュレーション構成設定
Tab.1 Configuration of the simulation experiments

simulation parameter	value
Rotational Speed (RPM)	7200
# of surfaces	4
# of sectors per cylinder	2520 - 5184
# of zone	29
Full stroke seek time (msec)	14.7
Single track seek time (msec)	0.8
Head switch time (msec)	1.4
buffer size	2048KB
system component	**#**
disk node	4
client	2
workload parameter	**value**
time span (days)	2 (day 24 to 25)
# of request	約 200,000
read:write	10:0
total file size	約 2GB

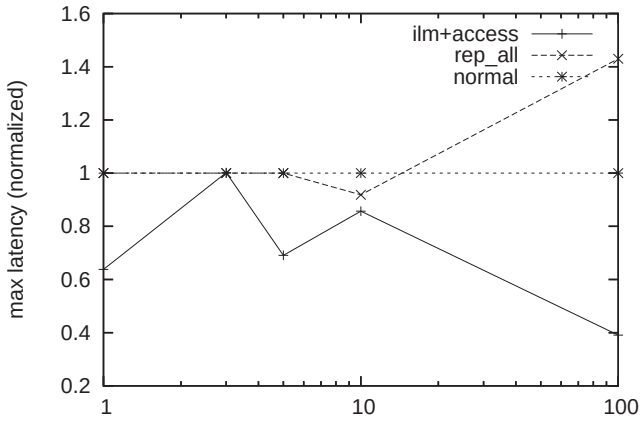

図 2: 最大レイテンシタイム比
Fig.2 Ratio of max latency time

5.3 複製配置による性能への影響の測定

実験結果を図 2 に示す．横軸は負荷の大きさを表す指標の値である．縦軸は，レプリケーションを行わない場合の最大レイテンシタイムを 1 とした場合の相対最大レイテンシタイムである．

図より，ライフサイクル情報と利用頻度情報を用いたレプリケーション (ilm+access) が多くの場合で最大レイテンシタイムを削減できていることがわかる．これは，最大レイテンシを記録する大きなアクセスの発生時に，他のアクセスをその他のディスクへ分散しており，かつ高頻度利用データの複製によりキャッシュ利用率を大きく低下させることがないためである．

5.4 格納ファイルサイズの増加

利用開始タイミングを各コンテンツに関して（あらかじめワークロード情報から計算して得た）ピーク時の 1/5 のトラフィックが初めて観測された時間，利用終了タイミングをピーク時の半分のトラフィックが最後に観測された時間としたとき，ファイルの頻度を考慮せずに複製を作成した場合 (rep_all) で最大で 53%格納データ量が増加した．

一方，利用開始タイミングの利用頻度が 20 [request/hour] 以

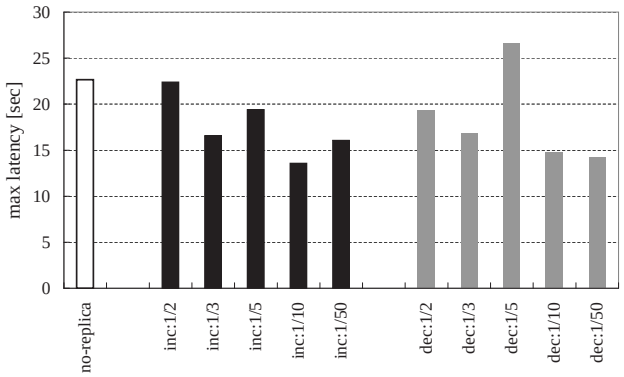

図 3: ライフサイクル情報精度の最大レイテンシタイムへの影響
Fig.3 Effects of accuracy of ILM info. on Max latency time

下，2 [request/hour] 以上のコンテンツを中・低頻度コンテンツであると定義したとき，時間に寄らず常に複製データを配置した時，最大で 66%格納データ量が増加した．

上記のライフサイクルタイミングと頻度を両方考慮し複製生成破棄を行った場合 (ilm+access)，格納データ量は最大で 38%増加した．

5.3 とあわせ，利用状況とライフサイクルを併用することで，性能と格納効率のいずれについても効率がよいレプリケーションが実行可能であることが得られた．

5.5 ライフサイクル情報精度の性能への影響

本実験ではライフサイクル情報はあらかじめ得られた正しいワークロードから算出しているが，現実の情報は予測できない誤りが生じることが多いと考えられる．ここでは，5.2 で述べたタイミング定義に用いたピーク時と比較したトラフィック割合の値を変化させ，その傾向を調べた．

まず利用終了タイミングを固定し，利用開始タイミングについて変化させた結果の最大レイテンシタイムを図 3(左）に表す．ここで，横軸の値が小さいほど早く複製が作成されている．図より，今回のワークロードに関してはピーク時の 1/2 を示すタイミングよりも十分早く複製が作成されていればタイミングの差異に依らず負荷分散による性能向上を実現できている．同様に利用開始タイミングを固定し，利用終了タイミングについて変化させた結果の最大レイテンシタイムを図 (右) に表す．横軸の値が小さいほど破棄タイミングが遅い場合を示す．終了タイミングは，遅ければ遅いほど最大レイテンシを低く抑えられることがわかる．しかし，複製削除タイミングを遅くすることはディスク利用効率の悪化を招く．

6. まとめと今後の課題

本稿では，ストレージ管理コスト削減を目指し，ユーザやアプリケーションから与えられたライフサイクル情報とデータ利用傾向情報を用いたストレージ管理機構について論じ，複製を用いた負荷分散への応用について述べた．ユーザ・アプリケーション側とストレージシステム側でそれぞれ高い精度で得られるデータのライフサイクル情報を統合し，ストレージ管理に利用することで，高い精度のストレージ管理がより低コストで実現できることが考えられる．具体的な適用例として，複製データを用いた負荷分散管理に対しライフサイクル情報を利用した複製生成破棄制御を行うことで， より性能の良いストレージシステムが実現できることを述べた．

今後は，今回述べたコンセプトを具体的に実現するストレージ管理ルール生成アルゴリズムを構築することが重要である．本稿で示したライフサイクルによるストレージ管理ルール生成は， 特に利用傾向解析では難しい QoS 制御に適していると考えている．よって，大量のデータ移動を伴うことで，一時的な性能低下を引き起こす，データマイグレーションによる負荷均衡化手法のタイミング制御への適用による QoS が課題として挙げられる．また，与えられるライフサイクル情報は非常に曖昧であり現実の挙動との誤差はもちろん大きい．そのずれを，アクセス利用傾向解析により補正することで，より高度なストレージ管理が実現できると考えている．また，それらを実装したストレージ管理ルール・コンパイラの実現が課題として挙げられる．

[謝辞]

本研究の一部は，独立行政法人科学技術振興機構戦略的創造研究推進事業 CREST，文部科学省科学研究費補助金特別研究員奨励費，情報ストレージ研究推進機構 (SRCs)，文部科学省科学研究費補助金特定領域研究 (18049026) および東京工業大学 21 世紀 COE プログラム「大規模知識資源の体系化と活用基盤構築」の助成により行なわれた．

[文献]

[1] Mandis Beigi, Murthy V. Devarakonda, Rohit Jain, Marc Kaplan, David Pease, Jim Rubas, Sugata Ghosal, Rohit Jain, Upendra Sharma, and Akshat Verma. Policy-based information lifecycle management in a large-scale file system. In *Sixth IEEE International Workshop on Policies for Distributed Systems and Networks*, pp. 139–148, 2005.

[2] Kensuke Ota, Dai Kobayashi, Takashi Kobayashi, Ryo Taguchi, and Haruo Yokota. Treatment of rules in individual metadata of flexible contents management. In *International Special Workshop on Databases For Next Generation Researchers (SWOD 2006) in conjunction with ICDE 2006*, 2006.

[3] Huseyin Simitci. *Storage Network Performance Analysis*. Wiley Technology Publishing, 2003.

[4] Christopher R. Lumb, Richard Golding, and Gregory R. Ganger. DSPTF: Decentralized request distribution in brickbased storage systems. In *Proceedings of ASPLOS'04*, Boston, MA, October 2004.

[5] Dai Kobayashi, Akitsugu Watanabe, Ryo Taguchi, Toshihiro Uehara, and Haruo Yokota. An efficient access forwarding method based on caches on storage nodes. In *International Special Workshop on Databases For Next Generation Researchers (SWOD 2005) In Memoriam of Prof. Kambayashi*, pp. 188–191, 2005.

[6] Windsor W. Hsu and Alan Jay Smith. Characteristics of I/O traffic in personal computer and server workloads. *IBM Systems Journal*, Vol. 42, No. 2, pp. 347–372, February 2003.

[7] Hitachi Global Storage Technologies. *Deskstar T7K250 Hard Disk Drive Specification*. http://www.hitachigst.com, ver. 1.7 edition, 2006.

[8] Lawrence Berkeley National Laboratory. The internet traffic archive. http://ita.ee.lbl.gov/.

[9] Martin Arlitt and Tai Jin. Workload characterization of the 1998 world cup web site. Technical Report HPL-1999-35R1, Hewlett-Packart Laboratories, October 1999.

小林 大 Dai KOBAYASHI

平 15 東工大・工・情工卒．平 17 同大大学院・情報理工・計算工・修士課程了．同大大学院・情報理工・計算工・博士課程在学中．平 18 日本学術振興会特別研究員 DC. 日本データベース学会学生会員．

田口 亮 Ryo TAGUCHI

平 6 慶應義塾大大学院・理工・計測工・修士課程了．同年より NHK 放送技術研究所．映像情報メディア学会会員．

横田 治夫 Haruo YOKOTA

昭 55 東工大・工・電物卒．昭 57 同大大学院・情報・修士課程了．同年富士通 (株)．同年 6 月 (財) 新世代コンピュータ技術開発機構研究所．昭 61(株) 富士通研究所．平 4 北陸先端大・情報・助教授．平 10 東工大・情報理工・助教授．平 13 東工大・学術国際情報センター・教授．工博．日本データベース学会理事．電子情報通信学会，情報処理学会，人工知能学会，IEEE，ACM 各会員．