

## DIB 専用 DBMS:ASSIST/D を用いた分散処理対応 3D-6 OSI ディレクトリシステムエージェント (DSA) の評価

横田 英俊 西山 智 小花 貞夫 浅見 徹 鈴木 健二  
国際電信電話株式会社 研究所

### 1. はじめに

筆者らはこれまでに、高速な OSI ディレクトリ<sup>[1]</sup>を実現するために、ディレクトリ情報ベース (DIB) 専用 DBMS:ASSIST/D<sup>[2]</sup>の開発と、ハッシュを用いた高速名前解析処理方式<sup>[3]</sup>の提案を行った。今回、ASSIST/D をベースに OSI ディレクトリの分散処理機能を持つ DSA の実装を行ったので<sup>[4]</sup>、その評価について報告する。

### 2. 分散処理対応 DSA の実装概要

本 DSA の実装概要を以下に示す<sup>[4]</sup>。

- 拡張可能 DBMS 構築技法に基づき階層化、モジュール化されている ASSIST/D の最上位層の DSA モジュールを、分散処理機能を持つ DSA モジュールに変更して実装した。また、ハッシュを用いた高速名前解析処理方式<sup>[3]</sup>を採用している。
- 3つの分散処理形態 (チェイン、マルチキャスト、リフェラル) の全てと、全ての分散処理知識のタイプ (上位参照、下位参照、不特定下位参照、クロス参照) を提供する。
- 他の DSA が格納する DIB 部分のコピーを持つ機能 (シャドウ機能) を提供し、シャドウを用いて検索操作の負荷分散を実現できる。

### 3. DSA の評価実験

#### 3.1 基本性能の評価

DSA の基本性能となるローカルなエントリに対する操作実行と他の DSA への操作中継 (チェイン) の性能を測定した。表 1 に示した測定条件のもとで、図 1 に示す DIB を使用し、測定対象の DSA にローカルに格納するエントリ件数をパラメータとして、Read 操作の操作応答時間を測定した。計算機 A 上の DUA から同じく計算機 A 上の DSA1 のローカルなエントリへの Read 操作応答時間を図 2 に示す。また、図 3 に計算機 A での Read 操作の中継を示す。中継時間の測定では計算機 A 上の DUA から計算機 A の DSA1 を経由して計算機 B の DSA2 で操作実行が行なわれた場合の操作応答時間から計算機 B でのローカルな操作応答時間を減じて求めた。図 2、図 3 には UNIX の gprof に

よって取得した DSA 内での主な内部処理が占める割合も併せて示す。図 2 において操作実行時間がグラフに現れないのは、本実装で使用した名前解析処理方式では名前解析時にエントリ自身を読み込む等、ほとんどの Read 操作に必要な処理を行うため、実質的な操作実行の処理がきわめて小さいことによる。

表 1: 測定条件

測定環境	計算機 A(DSA1): Sun SPARC2 計算機 B(DSA2): Sun SPARC10 計算機 C(DSA3): Sun SPARC2
下位プロトコル	DUA/DSA 間、DSA/DSA 間ともイーサネット上で TCP/IP を使用した (socket インターフェース利用)。
測定に用いた DIB	図 1 に示す。部分木の各段において均等に分岐する。
測定内容	ランダムに選んだエントリの全属性 (5 個、約 60 バイト) の読み出し (1,000 回)

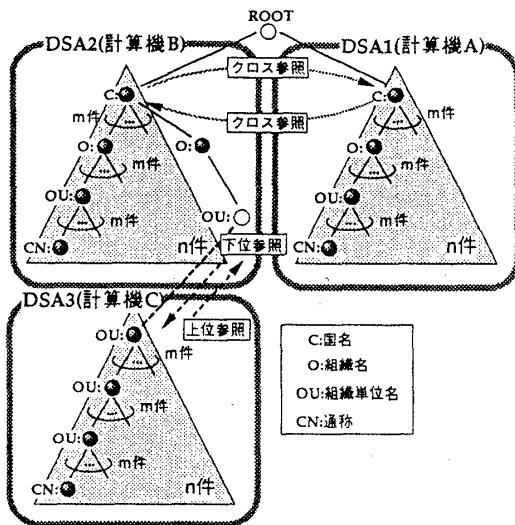


図 1: 測定に用いた DIB

#### 3.2 シャドウ機能の評価

シャドウ機能を評価するために、他 DSA からシャドウされたエントリに対して読み出しを行った場合と、シャドウ機能を使わずにチェインにより読み出しを行った場合の性能を測定した。まず、DSA1 に 1,000 件のエントリを格納し、その葉エントリ 26 個を同じく 1,000 件のエントリを格納している DSA3 にシャドウし、DSA3 に接続した DUA から DSA3 に対してこれらのエントリへの Read 操作の操作応答時間を測定した。次にシャドウ機能を使わずに、DSA3、DSA2 を経由した場合の

"Evaluation of OSI Directory System Agent with Support of Distributed Operations using ASSIST/D" by Hidetoshi YOKOTA, Satoshi NISHIYAMA, Sadao OBANA, Tohru ASAMI and Kenji SUZUKI  
KDD R & D Laboratories

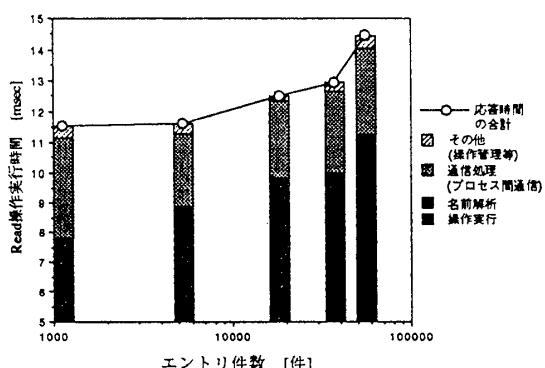


図 2: ローカルな Read 操作応答時間

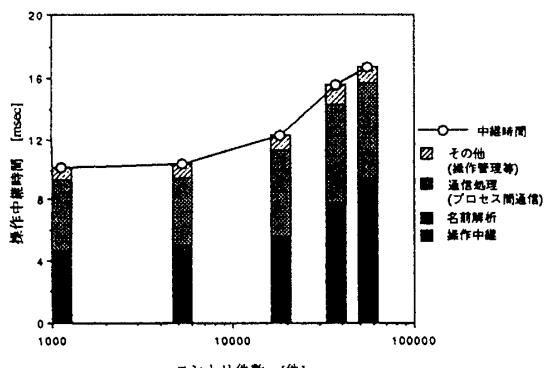


図 3: Read 操作中継時間

同一エントリに対する Read 操作の操作応答時間を測定した。また、更新操作についても、シャドウを行う場合とシャドウを行わない場合の両方について、シャドウ範囲のエントリに対する更新操作応答時間を測定した。これらの結果を表 2 に示す。

表 2: シャドウ機能の評価

Read 操作	シャドウ有り	11.6 [msec]
	シャドウ無し	30.7 [msec]
更新操作	シャドウ有り	45.7 [msec]
	シャドウ無し	14.8 [msec]

#### 4. 考察

##### 4.1 DSA の基本性能について

図 2、図 3 より、計算機 A 上で 18,000 件のエントリ格納時には、ローカルなエントリに対する Read 操作の操作応答時間は 12.5 ミリ秒、および Read 操作の中継時間は 12 ミリ秒であった。例えば、交換網の呼処理に必要なデータを提供する DB を構築する場合には、100 ミリ秒程度の応答時間が要求される<sup>[5]</sup>。評価結果より、本 DSA を用いることで 1~2 段程度の DSA の中継が入っても十分応答時間の要求を満たせることから、呼処理用の DB としても適用できる高速性を持つ。

内部の処理時間については、測定した格納エントリ件数の範囲では、全体に対する通信処理の割合が 1/3~1/2 程度を占めている。この結果より、ソケットによる

プロセス間通信処理がボトルネックになるまで高速化が図れたことが分かる。

#### 4.2 シャドウ機能の性能について

シャドウされたエントリも通常のエントリと同様の形態で格納されるため、検索操作に対しては表 2 より、ローカルなエントリと同程度の 11.6 ミリ秒という応答時間となっている。一方、シャドウ機能を用いない場合の Read 操作応答時間は、DSA3、DSA2 を経由して DSA1 のエントリを読むため、シャドウ機能を用いる場合のおおむね 3 倍になっている。また、更新操作については、シャドウ機能を用いるとシャドウ元 DSA ではシャドウログへの書き込みやログ内容の送信処理等のオーバヘッドにより 45.7 ミリ秒の操作応答時間を要する。さらにシャドウ先でも通常のシャドウ無しの場合に相当する処理が必要となる。1 つの DSA における更新操作の処理量を  $C_U$ 、検索操作の処理量を  $C_R$ 、全体に占める更新操作の比率を  $r$  とすれば、1 操作にかかる平均処理量は  $C_{AV} = C_Ur + C_R(1 - r)$  で与えられる。シャドウすることでシャドウ範囲に対する検索操作を 2 つの DSA で処理することができるため、シャドウを行った場合の処理能力は、1 つの DSA に対して検索操作のみを行う場合を 1 とすると  $\frac{2C_R}{C_{AV}}$  で与えられる。表 2 の値を DSA で必要な処理量とみなすと、この例では更新操作の比率  $r$  が 0.2 以下の場合に 1 以上の値が得られ、シャドウによる効率化が図れることが分かる。

#### 5. おわりに

本稿では、DIB 専用 DBMS:ASSIST/D をベースに開発した分散処理機能を持つ DSA の性能評価について報告した。DSA がエントリを 18,000 件格納した状態で、Sun SPARC2 上で、ローカルな Read 操作を 12.5 ミリ秒、Read 操作の他の DSA への中継を 12 ミリ秒で実行でき、交換網の呼処理に必要なデータを提供する DB としても適用できる高速性があることが示された。また、シャドウ機能による検索操作の効率化についても実証できた。最後に日頃御指導頂く KDD 研究所 浦野所長、真家次長に感謝します。

#### 参考文献

- [1] ITU-T Draft New Recommendation X.500 Series: The Directory (1992).
- [2] 西山 他：“拡張可能 DBMS 構築技法に基づく高速 OSI ディレクトリ専用 DBMS の設計と評価”、情処論文誌 Vol.34、No.6、(1993).
- [3] 西山 他：“ハッシュを用いた OSI ディレクトリの高速名前解析処理方式”、情処第 47 回全大 6F-2、(1993).
- [4] 西山 他：“DIB 専用 DBMS:ASSIST/D を用いた分散処理対応 OSI ディレクトリシステムエージェントの実装”、情処第 48 回全大 3D-5、(1994).
- [5] Bellcore: Advanced Intelligent Network(AIN) Release 1: Service Logic Program Framework Generic Requirements, Issue 1, (1991).