

問1 データベースの設計、実装に関する次の記述を読んで、設問1～3に答えよ。

B銀行では、支店の窓口において、預金の入出金、税金・公共料金の収納、振込などの金融サービス業務（以下、窓口業務という）を行っている。現在、行員が使用する窓口業務専用の端末（以下、窓口端末という）で記録されたログを、障害調査、監査などに利用している。今後は、蓄積したログを更に活用して、事務手続、アプリケーションソフトウェア（以下、APという）の改善に役立てるために、ログ分析システムを構築することにした。

[システムの現状]

1. 現行システムの構成

- (1) 支店は60支店あり、店番で識別している。各支店には、複数の窓口端末がある。窓口端末は、PC、モニタ、スキャナ、プリンタ、現金処理機などを組み合わせた端末で、支店ごとに一意な機番で識別している。
- (2) データセンタが1拠点ある。データセンタには、窓口端末サーバ、ホストコンピュータがある。窓口端末サーバでは、全支店の窓口端末の監視、ログ収集、APの配布などを行う運用管理システムが稼働しており、ホストコンピュータでは、業務処理を行う勘定系システムが稼働している。
- (3) 支店内、データセンタ内の機器はLANで接続されており、LAN間はWANで接続されている。
- (4) 窓口端末のPCでは、窓口端末用のAP（以下、端末APという）が稼働している。端末APは、業務ロジックをもたず、勘定系システムと連動しながら、画面の入出力及び接続機器の制御を行っている。
- (5) 端末APでは、預金入金、預金出金、振替などの取引ごとに取引メニューが用意されており、取引は、取引番号で識別している。取引メニューに対応する手続用の画面が複数ある。画面は、全体で一意な画面番号で識別している。

2. ログ収集・利用状況

- (1) 窓口端末には、ログを自動的に記録する機能が備わっている。窓口端末のPCは、ログをログファイルに保存し、約15分ごとに窓口端末サーバに転送している。

(2) ログには、店番、機番、タイムスタンプ（以下、TS という）、及びログテキストを記録する。ログテキストは、取引番号、画面番号、画面ごとに固有な項目とその値を “[項目 ID:項目名=値:値の説明]”（値の説明の表示は任意）の形式で編集した文字列を連結したものである。表 1 にログの例を示す。

表 1 ログの例（一部省略）

店番	機番	TS	ログテキスト
1234	2501	2019-03-15 11:24:11.120	[201:取引番号=3322:預金出金][301:画面番号=0777:口座出金依頼][155:処理区分=001:取引開始][101:行員番号=1122334:情報花子][017:科目=2:普通][005:口座店番=543:○○支店][006:口座番号=00578212:]…
1234	2501	2019-03-15 11:24:20.098	[201:取引番号=3322:預金出金][301:画面番号=0001:依頼書等受領確認][155:処理区分=055:残高確認依頼][315:受領結果=1:OK]…
1234	2501	2019-03-15 11:24:27.211	[201:取引番号=3322:預金出金][301:画面番号=0028:通帳受領確認][018:出金依頼金額=10000:][155:処理区分=055:残高確認依頼]…

- (3) 窓口端末サーバでは、RDBMS が稼働していて、窓口端末の PC から収集したログファイルのデータを、約 1 時間ごとに “ログ収集” テーブルに格納している。
- (4) 利用者である行員（以下、利用者という）は、店番、機番、及び TS の範囲を指定してテーブルの行を選択し、TS 及びログテキストを射影する問合せを行つて、障害調査、監査の際に利用している。

〔新たなログ分析システムの構築〕

1. ログ分析システムの主な機能

- (1) 蓄積した 60 か月分のログを対象に、障害調査、監査、事務手続改善、AP 改善に用いるデータを抽出するログ分析処理の機能を提供する。表 2 に、ログ分析処理の例を示す。
- (2) ログ分析システムの利用時間帯は、平日の 8:00～22:00 である。毎月の最終営業日の 22:00 から、翌営業日の 8:00 までの間に、過去データの削除、再編成などの処理（以下、月末処理という）を行う。

表2 ログ分析処理の例

処理名	内容
処理 1	利用者が指定した年月に一致する 1 か月分のログを対象に、店番、店名、機番、TS、行員番号、行員氏名、取引番号、取引種別コードを出力する。利用者の権限レベル（‘1’、‘2’、‘3’）によって参照可能なログが異なり、‘1’ の場合は、当該利用者がその時点で所属する支店かつ当該利用者のログだけを、‘2’ の場合は、当該利用者がその時点で所属する支店のログだけを、‘3’ の場合は、全ログを対象にする。
処理 2	全ログを対象に、店番、機番ごとに、TS 順に連続する二つのログの画面番号の組を前画面番号、後画面番号としてログ間の経過時間（ミリ秒単位の整数）の平均値を求め、前画面番号、後画面番号、平均経過時間を、平均経過時間の降順に出力する。同じ画面番号が連続する場合は、後画面番号と前画面番号は同じになる。
処理 3	利用者が指定した店番、機番、及び TS の範囲（開始年月～終了年月）に該当するログの TS、取引番号、取引名を、操作の分岐に従ってツリー形式で表示する。また、利用者が TS を一つ選択すると、店番、機番、TS に対応するログテキストを表示する。
処理 4	利用者が指定した一つの店番について、年月が前年 4 月から前年 9 月までの 6 か月分のログを対象に、年月、行員番号、取引種別コードごとに伝票金額を集計して、店番、年月、行員番号、取引種別コード、取引種別名、合計伝票金額の一覧を出力する。
処理 5	利用者が指定した前月以前の年月に一致する 1 か月分のログを対象に、店番、端末種別コードごとに明細件数を集計して、店番、店名、端末種別コード、端末種別名、合計明細件数を出力する。
処理 6	利用者が指定した一つの画面番号について、年月が前々年 4 月から前年 3 月までの 12 か月分のログを対象に、店名、取引種別名ごとに、明細件数を集計して、画面番号、タイトル、店名、取引種別名、合計明細件数を出力する。

2. ログ分析システムの構成

- (1) 窓口端末サーバで稼働中の RDBMS をログ分析システムに利用する。
- (2) 分析用データの作成・参照・削除に用いる AP（以下、分析 AP という）を新たに開発し、窓口端末及び窓口端末サーバ上で稼働させる。

3. ログ分析システムのテーブル

次の方針でデータベースにデータを格納することにし、図 1 のログ分析システムのテーブル構造、表 3 の主なテーブルのデータ量見積りを作成した。

- (1) 支店から収集したログは、現状どおり“ログ収集”テーブルに格納する。格納処理は、ログファイルごとに、並行して実行する。新たに発生したログを追加するだけで、更新は行わない。
- (2) “ログ基本”テーブルには、“ログ収集”テーブルの各行に対応する行を格納し、一意な文字列であるログ ID で識別する。検索、集計のキーとなる項目を列として定義し、ログテキストから取り出した値を設定する。年月には、TS の年

月を設定する。

- (3) ログテキスト内に記録されている複数の操作を、操作ごとに分割して“ログ明細”テーブルに格納する。
- (4) 取引の開始から終了までの手続は、複数の操作に分岐していくので、ログは木構造をもつ。分析 AP によってログを解析し、取引開始ログをルートノードに、他のログをノードとする木構造の関連を“ログ関連”テーブルに格納する。
- (5) マスタ情報（支店、端末種別、窓口端末、取引種別、取引、画面、行員、行員所属）を各テーブルに格納する。マスタ情報は、データ量が少なく、更新は月1回なので、月末処理で変更を反映する。
- (6) “ログ収集”，“ログ基本”，“ログ明細”，“ログ関連”テーブルは、月末処理において、TS が翌月の 60 か月前の月以前の行を削除して、59 か月分のログが保存された状態にする。月が変わった後のログを合わせて 60 か月分を保有する。

支店（店番, 店名, 所在地）

端末種別（端末種別コード, 端末種別名）

窓口端末（店番, 機番, 端末種別コード, 設置場所）

取引種別（取引種別コード, 取引種別名）

取引（取引番号, 取引種別コード, 取引名）

画面（画面番号, タイトル）

行員（行員番号, 行員氏名, …）

行員所属（行員番号, 適用開始日, 所属店番, 権限レベル, 適用終了日）

ログ収集（店番, 機番, TS, ログテキスト）

ログ基本（ログ ID, 店番, 機番, TS, 行員番号, 取引番号, 画面番号, 年月, 処理区分, 口座店番, 科目, 口座番号, 明細件数）

ログ明細（ログ ID, 明細番号, 店番, 機番, TS, 年月, 開始時刻, 終了時刻, 伝票金額, ホスト送信データ, ホスト受信データ, 終了状態, …）

ログ関連（[REDACTED]）

注記 各テーブルの主キーには、索引が定義されている。

網掛け部分は、表示していない。

図1 ログ分析システムのテーブル構造（一部省略）

表3 主なテーブルのデータ量見積り

テーブル名	見積行数	ページ長（バイト）	平均行長（バイト）	ページ数
ログ収集	540M	4,000	900	135M
ログ基本	540M	4,000	400	60M
ログ明細	1,620M	4,000	200	90M

注記 表中の単位 M は 100 万を表す。

[RDBMS の主な仕様]

ログ分析システムに利用する RDBMS の主な仕様は次のとおりである。

1. ページ

RDBMS とストレージ間の入出力単位をページという。同じページに異なるテーブルの行が格納されることはない。

2. オプティマイザの仕様

(1) LIKE 述語の検索パターンが ‘ABC%’ のように前方一致の場合は索引探索を選択し, ‘%ABC%’, ‘%ABC’ のように部分一致, 後方一致の場合は表探索を選択する。

(2) WHERE 句の述語が関数を含む場合, 表探索を選択する。

3. 再帰的な問合せの構文のサポート

WITH 句に RECURSIVE を指定した再帰的な問合せの構文をサポートする。再帰的な問合せの構文を用いると, 例えば, 階層構造の組織について, ある組織を起点として上位又は下位の組織を, 階層に沿って連続的に検索することができる。

4. ウィンドウ関数のサポート

ウィンドウ関数をサポートする。例えば, 選択する行ごとに, その直前の行の列値を参照したい場合には, LAG 関数が使用できる。

5. テーブルの物理分割

(1) テーブルごとに一つ又は複数の列を区分キーとし, 区分キーの値に基づいて物理的な格納領域を分ける。これを物理分割という。

(2) 区分方法には, ハッシュとレンジの二つがある。ハッシュは, 区分キー値を基に RDBMS 内部で生成するハッシュ値によって, 一定数の区分に行を分配する方法である。レンジは, 区分キー値によって決められる区分に行を分配する方法で, 分配する条件を, 値の範囲又は値のリストで指定する。

(3) 物理分割されたテーブルには, 区分キーの値に基づいて分割された索引（以下, ローカル索引という）を定義できる。ローカル索引のキー列には, 区分キーを構成する列（以下, 区分キー列という）が全て含まれていなければならぬ。

(4) テーブルを検索する SQL 文の WHERE 句の述語に区分キー列を指定すると, 区分キー列で特定した区分だけを探索する。また, WHERE 句の述語に, ローカ

ル索引の先頭列を指定すると、ローカル索引によって区分内を探索することができる。

- (5) 問合せの実行時に、一つのテーブルの複数の区分を並行して同時に探索する。同一サーバ上では、問合せごとの同時並行探索数の上限は 20 である。
- (6) 指定した区分を削除するコマンドがある。区分内の格納行数が多い場合、コマンドによる区分の削除は、DELETE 文よりも高速である。

6. クラスタ構成のサポート

- (1) シェアードナッシング方式のクラスタ構成をサポートする。クラスタは複数のノードで構成され、各ノードには、当該ノードだけがアクセス可能なディスク装置をもつ。
- (2) 各ノードへのデータの配置方法には、次に示す分散と複製があり、テーブルごとにいずれかを指定する。
 - ・ 分散による配置方法は、一つ又は複数の列を分散キーとして指定し、分散キーの値に基づいて RDBMS 内部で生成するハッシュ値によって各ノードにデータを分散する。分散キーに指定する列は、主キーを構成する全て又は一部の列である必要がある。
 - ・ 複製による配置方法は、全ノードにテーブルの複製を保持する。
- (3) データベースへの要求は、いずれか一つのノードで受け付ける。要求を受け付けたノードは、要求を解析し、自ノードに配置されているデータへの処理は自ノードで処理を行う。自ノードに配置されていないデータへの処理は、当該データが配置されている他ノードに処理を依頼し、結果を受け取る。特に、テーブル間の結合では、他ノードに処理を依頼するので、自ノード内で処理する場合と比べて、ノード間通信のオーバーヘッドが発生する。

[問合せの検討]

1. 問合せの傾向分析・索引設計

表 2 の処理 1~6 の参照テーブルを表 4 にまとめて、問合せの傾向を分析した。また、図 1 中のマスタ情報を格納するテーブルについて、処理 1~6 で結合に用いられる主キー以外の列に索引を定義することにし、その対象テーブル名・列名を表 5 にまとめた。

表4 処理1～6の参照テーブル（未完成）

テーブル名 処理名	支店	端末種別	窓口端末	取引種別	取引	画面	行員	行員所属	ログ収集	ログ基本	ログ明細	ログ関連
処理1	○				○		○	○		○		
処理2										○		
処理3					○				○	○		○
処理4				○	○					○	○	
処理5										○		
処理6										○		

注記 ○：テーブルが処理で参照されることを表す。

表5 主キー以外の索引定義対象テーブル名・列名（未完成）

テーブル名	索引を定義する列名

2. 処理1の問合せ検討

表2の処理1の問合せに用いる図2のSQL文を作成した。

```

WITH TEMP AS (SELECT
CASE WHEN 権限レベル [a] THEN 所属店番 ELSE NULL END AS 検索店番,
CASE WHEN 権限レベル [b] THEN 行員番号 ELSE NULL END AS 検索行員番号
FROM 行員所属
WHERE 行員番号 = :hv1 AND CURRENT_DATE >= 適用開始日
      AND (適用終了日 IS NULL OR CURRENT_DATE < 適用終了日))
SELECT A.店番, B.店名, A.機番, A.TS, A.行員番号, C.行員氏名, A.取引種別コード
FROM ログ基本 A, 支店 B, 行員 C, 取引 D, TEMP E
WHERE A.年月 = :hv2
      AND A.店番 = B.店番
      AND A.行員番号 = C.行員番号
      AND A.取引番号 = D.取引番号
      AND (E.検索店番 [c] OR E.検索店番 = A.店番)
      AND (E.検索行員番号 [c] OR E.検索行員番号 = A.行員番号)

```

注記 hv1 は利用者の行員番号のホスト変数を、 hv2 は利用者が指定した年月のホスト変数を表す。

図2 処理1の問合せに用いるSQL文（未完成）

3. 処理 2 の問合せ検討

表 2 の処理 2 の問合せに用いる SQL 文の例を、図 3～5 に作成した。ここで、
GET_LAPSE(TS1, TS2)は、TS1 から TS2 までの経過時間を、ミリ秒単位の整数で
返却するユーザ定義関数である。また、図 5 中の ア は、図 3, 4 の SQL
文と同じ結果を返すように、“ログ基本” テーブルに追加した列の列名である。

なお、図 3～5 の結果行には、分析対象でない前画面番号と後画面番号の組が含
まれるが、これらは、分析 AP 上で別途除外されるものとする。

```
SELECT B.画面番号 AS 前画面番号, A.画面番号 AS 後画面番号,  
    AVG(GET_LAPSE(B.TS, A.TS)) AS 平均経過時間  
FROM ログ基本 A  
    LEFT JOIN ログ基本 B ON A.店番 = B.店番 AND A.機番 = B.機番  
        AND B.TS = (SELECT MAX(Z.TS) FROM ログ基本 Z  
            WHERE A.店番 = Z.店番 AND A.機番 = Z.機番 AND Z.TS < A.TS)  
WHERE B.画面番号 IS NOT NULL  
GROUP BY B.画面番号, A.画面番号  
d
```

図 3 処理 2 の問合せに用いる SQL 文の例 1（未完成）

```
SELECT 前画面番号, 後画面番号, AVG(経過時間) AS 平均経過時間 FROM  
(SELECT LAG(画面番号) OVER (PARTITION BY 店番, 機番 ORDER BY TS) AS 前画面番号,  
    画面番号 AS 後画面番号,  
    GET_LAPSE(LAG(TS) OVER (PARTITION BY 店番, 機番 ORDER BY TS), TS) AS 経過時間  
    FROM ログ基本) TEMP  
WHERE 前画面番号 IS NOT NULL  
GROUP BY 後画面番号, 前画面番号  
d
```

図 4 処理 2 の問合せに用いる SQL 文の例 2（未完成）

```
SELECT B.画面番号 AS 前画面番号, A.画面番号 AS 後画面番号,  
    AVG(GET_LAPSE(B.TS, A.TS)) AS 平均経過時間  
FROM ログ基本 A  
    LEFT JOIN ログ基本 B ON A.店番 = B.店番 AND A.機番 = B.機番  
        AND (A.ア - 1) = B.ア  
WHERE B.画面番号 IS NOT NULL  
GROUP BY A.画面番号, B.画面番号  
d
```

図 5 処理 2 の問合せに用いる SQL 文の例 3（未完成）

[“ログ関連” テーブルの検討]

1. テーブル構造の案

木構造をもつデータを取り扱う場合、テーブル構造によって、行の追加・削除の効率、問合せに用いる SQL の構文、問合せの性能、データの制限が異なる。そこで、表 6 の “ログ関連” テーブルのテーブル構造案 1~3 を検討することにした。検討に当たって、木構造をもつログの例を用いて案を比較することにし、各案におけるテーブルの行を表 7~9 に、案 3 の左端番号・右端番号付与の例を図 6 にまとめた。

表 6 “ログ関連” テーブルのテーブル構造案

案	テーブル構造	説明
案 1	ログ関連 (<u>ログ ID</u> , 親 <u>ログ ID</u>)	各ノードの上位ノードのログ ID を、親ログ ID に設定する。ルートノードの親ログ ID は NULL にする。親ログ ID には、索引を定義する。
案 2	ログ関連 (<u>ログ ID</u> , 経路)	ルートノードから各ノードに至る全ノードのログ ID を、接続文字 (“/”) で連結した文字列値を経路に設定する。経路には、索引を定義する。
案 3	ログ関連 (<u>ログ ID</u> , 左端番号, 右端番号)	左端番号及び右端番号は、両方を合わせて連続する一意な番号であり、あるノードの左端番号と右端番号は、その下位の全ノードに対して、“上位ノードの左端番号 < 下位ノードの左端番号”かつ“下位ノードの右端番号 < 上位ノードの右端番号”となるように設定する。左端番号、右端番号にはそれぞれ索引を定義する。

表 7 案 1 の行

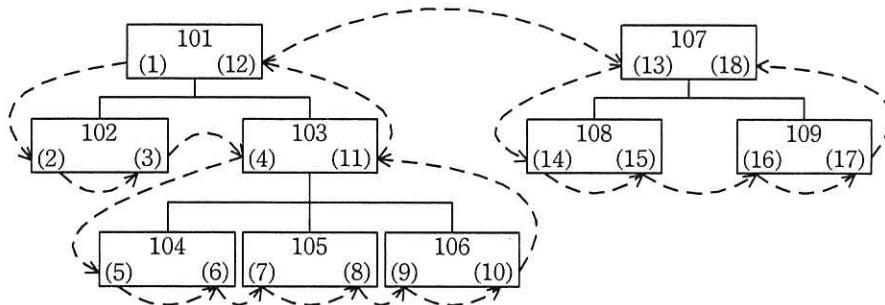
ログ ID	親ログ ID
101	NULL
102	101
103	101
104	103
105	103
106	103
107	NULL
108	107
109	107

表 8 案 2 の行

ログ ID	経路
101	101
102	101/102
103	101/103
104	101/103/104
105	101/103/105
106	101/103/106
107	107
108	107/108
109	107/109

表 9 案 3 の行

ログ ID	左端番号	右端番号
101	1	12
102	2	3
103	4	11
104	5	6
105	7	8
106	9	10
107	13	18
108	14	15
109	16	17



凡例 : ノードを表す。上部中央の数字はログ ID、左下の括弧内の数字は左端番号・右端番号
 : 番号付与の順序

図 6 案 3 の左端番号・右端番号付与の例

2. テーブル構造案の検討

表 6 の案ごとに、次の問合せ[A], [B]に用いる SQL 文の案を、表 10 にまとめた。

- ・問合せ[A] ログ ID = '101' 及びその下位の全てのログ ID を抽出する。
- ・問合せ[B] ログ ID = '105' 及びその上位の全てのログ ID を抽出する。

表 10 問合せに用いる SQL 文の案（未完成）

案	問合せ[A]	問合せ[B]
案 1	<pre>WITH RECURSIVE TEMP(ログ ID) AS (SELECT ログ ID FROM ログ関連 WHERE ログ ID = '101' UNION ALL SELECT A.ログ ID FROM ログ関連 A, TEMP B WHERE A.親ログ ID = B.ログ ID) SELECT ログ ID FROM TEMP</pre>	<pre>WITH RECURSIVE TEMP(ログ ID, 親ログ ID) AS (SELECT ログ ID, 親ログ ID FROM ログ関連 WHERE e UNION ALL SELECT A.ログ ID, A.親ログ ID FROM ログ関連 A, TEMP B WHERE A.ログ ID = f) SELECT ログ ID FROM TEMP</pre>
案 2	<pre>SELECT ログ ID FROM ログ関連 WHERE 経路 LIKE '101%'</pre>	<pre>SELECT B.ログ ID FROM ログ関連 A, ログ関連 B WHERE A.ログ ID = '105' AND POSITION(B.経路 IN A.経路) = 1</pre>
案 3	<pre>SELECT B.ログ ID FROM ログ関連 A, ログ関連 B WHERE A.ログ ID = '101' AND A.左端番号 <= B.左端番号 AND A.右端番号 >= B.右端番号</pre>	<pre>SELECT B.ログ ID FROM ログ関連 A, ログ関連 B WHERE A.ログ ID = '105' AND g AND h</pre>

注記 POSITION(S1 IN S2)は文字列 S2 内で、文字列 S1 が最初に出現する文字位置を、S2 の最初の文字位置を 1 とする整数で返す関数。S1 が出現しない場合は 0 を返す。

3. テーブル構造案の評価

行の追加・削除の効率の観点では、例えば、ログ ID = ‘199’ の新たなログを ‘101’ の下位、‘103’ の上位に追加し、‘103’ の階層の深さが一つ下がる場合、‘199’ の行追加以外の更新行数は、案 1 では 1 行、案 2 では i 行、案 3 では j 行である。案 k の負荷が最も高いが、本業務では、l ので、変更時の負荷は問題にならない。

問合せの難易度の観点では、処理 3 の問合せに SQL 文を用いる場合、案 1 では、m の構文を用いるが、案 2, 3 では、選択、射影、n の関係演算を行う構文を用いればよい。案 2 と案 3 の問合せの性能を比較すると、RDBMS のオプティマイザの仕様から、表 10 の問合せ[A]ではほぼ同等であるが、問合せ[B]では案 2 が劣る。

これらの評価から、案 3 による実装が最適と判断されるが、左端番号、右端番号の列を整数型で定義する場合、整数型の上限を超えないように留意する必要がある。

[テーブルの物理分割・クラスタ構成の検討]

ログデータを格納するテーブルは、格納行数が多いので、データの追加、参照、削除の性能が懸念される。そこで、テーブルの物理分割を行うことにした。

1. “ログ収集” テーブルの物理分割

表 11 の “ログ収集” テーブルの物理分割案 A～C を作成し、性能評価を行った。各案の性能を相対的に評価し、性能が低い案から順に 0, 1, 2 の点数を付ける。評価が同じ案は、同じ点数とする。

表 11 “ログ収集” テーブルの物理分割案・性能評価

案	区分方法（区分キー） 区分への分割方法	ローカル索引	性能評価		
			追加	参照	削除
案 A	ハッシュ（店番、機番、TS） ハッシュ値ごとに 60 区分に分割	{店番、機番、TS}	1	0	0
案 B	レンジ（店番） 店番ごとに 60 区分に分割	{店番、機番、TS}	1	0	0
案 C	レンジ（TS） TS の年月ごとに 60 区分に分割	{TS, 店番, 機番}	0	0	1

注記 { } 内は、索引の列を定義順に記述したリストである。

2. “ログ基本” テーブルの物理分割

(1) 物理分割案とその評価

“ログ収集” テーブルと同様に、表 12 の “ログ基本” テーブルの物理分割案 D～F を作成し、性能評価を行った。追加、削除の性能評価は “ログ収集” テーブルと同様であったが、参照の性能評価は処理によって異なるので、更に処理ごとに性能評価を行うことにした。

表 12 “ログ基本” テーブルの物理分割案・性能評価

案	区分方法（区分キー） 区分への分割方法	ローカル索引	性能評価		
			追加	参照	削除
案 D	ハッシュ（ログ ID） ハッシュ値ごとに 60 区分に分割	{ログ ID} {店番, 機番, 年月, ログ ID}	1		0
案 E	レンジ（店番） 店番ごとに 60 区分に分割	{店番, ログ ID}	1		0
案 F	レンジ（年月） 年月ごとに 60 区分に分割	{年月, ログ ID}	0		1

注記 { }内は、索引の列を定義順に記述したリストである。

網掛け部分は表示していない。

(2) 探索対象ページ数試算による性能評価

表 2 の処理 4～6 における物理分割案 D～F について、次の①～④を前提条件として、探索対象の最大区分数（以下、探索区分数という）、テーブルからの読み込みページ数（以下、探索ページ数という）を試算し、表 13 を作成した。

- ① “ログ基本” テーブルへのアクセスだけを対象とする。どの処理でも、検索条件に合致する行は、30,000 行あるものとする。
- ② 月末営業日の処理を想定し、60 か月分のログが全て蓄積されていて、ログは各支店、各年月に均一に分布しているものとする。
- ③ どの区分方法でも、ページ総数は同じで、どのページにも最大行数のデータが格納され、ページは各区分に均等に配置されているものとする。また、テーブルの行が複数ページに分割されて格納されることはないものとする。
- ④ バッファヒット率は、索引のページでは 100%，テーブルのページでは 0% とする。

表 13 表 2 の処理 4~6 における探索区分数・探索ページ数試算（未完成）

前提／試算項目		案	案 D	案 E	案 F
前提	区分数		60	60	60
	テーブルのページ数		60M	60M	60M
	1区分当たりのページ数		1M	1M	1M
処理別 の試算	処理 4	探索区分数	60	1	6
		探索ページ数	30,000	1M	6M
	処理 5	探索区分数	<input type="text" value="o"/>	60	1
		探索ページ数	<input type="text" value="p"/>	60M	30,000
	処理 6	探索区分数	60	60	<input type="text" value="q"/>
		探索ページ数	60M	60M	<input type="text" value="r"/>

注記 表中の単位 M は 100 万を表す。

3. クラスタ構成の検討

テーブルの物理分割だけでは、将来的なデータ量の増加に対応できないので、RDBMS がサポートしているクラスタ構成の検討を行った。クラスタ構成への変更に当たり、各テーブルのデータの配置方法に分散を指定し、主キー列を分散キーとして設定することにした。

クラスタ構成への変更後、表 2 の処理 2, 4 の性能を試算したところ、処理 2 は同時並行探索数の上限がなくなることで性能が改善されたが、処理 4 は性能が低下することが判明した。そこで、処理 4 の性能低下への対策として、“取引種別” テーブル、“取引” テーブル及び“ログ明細” テーブルのデータの配置方法を見直すこととした。

設問 1 [問合せの検討] について、(1)~(4)に答えよ。

- (1) 表 4 中の太枠内に “○” 印を記入し、表を完成させよ。
- (2) 表 5 中の空欄を埋め、表を完成させよ。
- (3) 図 2 中の ~ に入れる適切な字句を答えよ。また、図 3~5 中の に入れる適切な字句を答えよ。
- (4) 図 5 中の の列に事前に設定すべき値の内容を、20 字以内で具体的に述べよ。

設問2 [“ログ関連” テーブルの検討] について、(1)～(3)に答えよ。

- (1) 表 10 中の

e

 ~

h

 に入れる適切な字句を答えよ。
- (2) “3. テーブル構造案の評価” の本文中の

i

 ~

k

 に入れ
る適切な数字を答えよ。また、

l

 ~

n

 に入れる適切な字句
を、本文中の用語を用いて答えよ。
- (3) “3. テーブル構造案の評価” の本文中の下線部について、問合せ[B]では、
案2の性能が案3よりも劣る理由を、30字以内で具体的に述べよ。

設問3 [テーブルの物理分割・クラスタ構成の検討] について、(1)～(3)に答えよ。

- (1) 表 11 中の性能評価について、次の①、②をそれぞれ 30 字以内で具体的に
述べよ。
 - ① 追加の性能評価において、案 C が低い理由
 - ② 参照の性能評価において、どの案も同じ理由
- (2) 表 13 中の

o

 ~

r

 に入れる適切な数値を答えよ。また、
案 E について、処理 4, 5 の探索ページ数の試算値が最小となるローカル索引
を構成する列名を全て答えよ。
- (3) “3. クラスタ構成の検討” について、①～③に答えよ。
 - ① 処理 4 の性能が低下した理由を、50字以内で具体的に述べよ。
 - ② “取引種別” テーブル及び “取引” テーブルについて行うべき対応を、
20字以内で具体的に述べよ。
 - ③ “ログ明細” テーブルの分散キーを見直すことにした。どのように見直
せばよいか、20字以内で具体的に述べよ。