

令和7年度 秋期
データベーススペシャリスト試験
午後Ⅱ 問題

試験時間

14:30 ~ 16:30 (2 時間)

注意事項

1. 試験開始及び終了は、監督員の時計が基準です。監督員の指示に従ってください。
2. 試験開始の合図があるまで、問題冊子を開いて中を見てはいけません。
3. 答案用紙への受験番号などの記入は、試験開始の合図があってから始めてください。
4. 問題は、次の表に従って解答してください。

問題番号	問 1, 問 2
選択方法	1 問選択

5. 答案用紙の記入に当たっては、次の指示に従ってください。
 - (1) B 又は HB の黒鉛筆又はシャープペンシルを使用してください。
 - (2) 受験番号欄に受験番号を、生年月日欄に受験票の生年月日を記入してください。
正しく記入されていない場合は、採点されないことがあります。生年月日欄については、受験票の生年月日を訂正した場合でも、訂正前の生年月日を記入してください。
 - (3) 選択した問題については、次の例に従って、選択欄の問題番号を○印で囲んでください。○印がない場合は、採点されません。2 問とも○印で囲んだ場合は、はじめの 1 問について採点します。
 - (4) 解答は、問題番号ごとに指定された枠内に記入してください。
 - (5) 解答は、丁寧な字ではっきりと書いてください。読みにくい場合は、減点の対象になります。

〔問 2 を選択した場合の例〕

選択欄	
1 問選択	問 1
	問 2

注意事項は問題冊子の裏表紙に続きます。
こちら側から裏返して、必ず読んでください。

問題文中で共通に使用される表記ルール

概念データモデル、関係スキーマ、関係データベースのテーブル（表）構造の表記ルールを次に示す。各問題文中に注記がない限り、この表記ルールが適用されているものとする。

1. 概念データモデルの表記ルール

(1) エンティティタイプとリレーションシップの表記ルールを、図1に示す。

- ① エンティティタイプは、長方形で表し、長方形の中にエンティティタイプ名を記入する。
- ② リレーションシップは、エンティティタイプ間に引かれた線で表す。
 - “1対1”のリレーションシップを表す線は、矢を付けない。
 - “1対多”のリレーションシップを表す線は、“多”側の端に矢を付ける。
 - “多対多”のリレーションシップを表す線は、両端に矢を付ける。

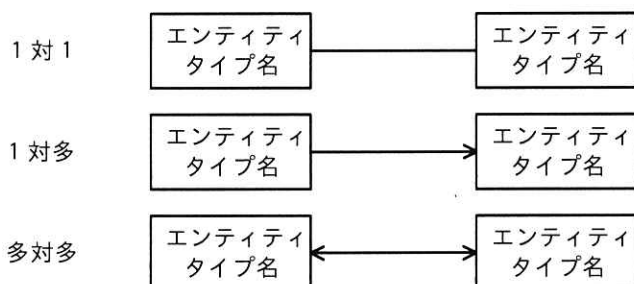
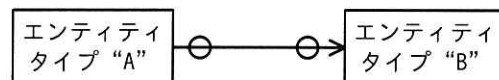


図1 エンティティタイプとリレーションシップの表記ルール

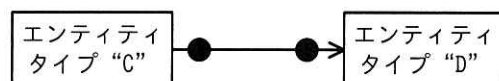
(2) リレーションシップを表す線で結ばれたエンティティタイプ間において、対応関係にゼロを含むか否かを区別して表現する場合の表記ルールを、図2に示す。

- ① 一方のエンティティタイプのインスタンスから見て、他方のエンティティタイプに対応するインスタンスが存在しないことがある場合は、リレーションシップを表す線の対応先側に“○”を付ける。
- ② 一方のエンティティタイプのインスタンスから見て、他方のエンティティタイプに対応するインスタンスが必ず存在する場合は、リレーションシップを表す線の対応先側に“●”を付ける。

“A” から見た “B” も，“B” から見た “A” も、インスタンスが存在しないことがある場合



“C” から見た “D” も，“D” から見た “C” も、インスタンスが必ず存在する場合



“E” から見た “F” は必ずインスタンスが存在するが，“F” から見た “E” はインスタンスが存在しないことがある場合

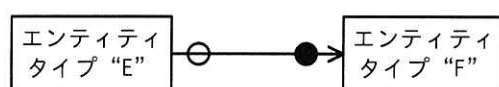
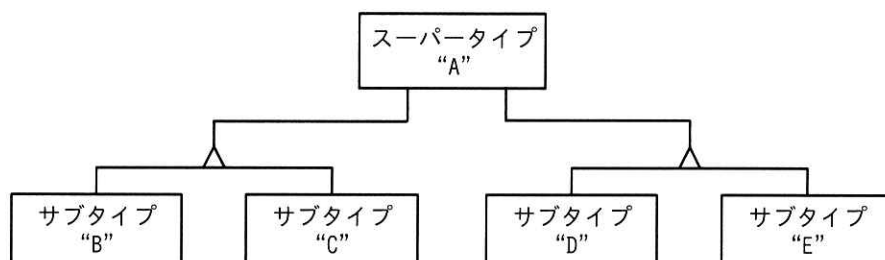


図2 対応関係にゼロを含むか否かを区別して表現する場合の表記ルール

(3) スーパータイプとサブタイプの間のリレーションシップの表記ルールを、図3に示す。

- ① サブタイプの切り口の単位に“△”を記入し、スーパータイプから“△”に1本の線を引く。
- ② 一つのスーパータイプにサブタイプの切り口が複数ある場合は、切り口の単位ごとに“△”を記入し、スーパータイプからそれぞれの“△”に別の線を引く。
- ③ 切り口を表す“△”から、その切り口で分類されるサブタイプのそれぞれに線を引く。



スーパータイプ“A”に二つの切り口があり、それぞれの切り口にサブタイプ“B”と“C”及び“D”と“E”がある例

図3 スーパータイプとサブタイプの間のリレーションシップの表記ルール

(4) エンティティタイプの属性の表記ルールを、図4に示す。

- ① エンティティタイプの長方形内を上下2段に分割し、上段にエンティティタイプ名、下段に属性名の並びを記入する。¹⁾
- ② 主キーを表す場合は、主キーを構成する属性名又は属性名の組に実線の下線を付ける。
- ③ 外部キーを表す場合は、外部キーを構成する属性名又は属性名の組に破線の下線を付ける。ただし、主キーを構成する属性の組の一部が外部キーを構成する場合は、

破線の下線を付けない。

エンティティタイプ名
<u>属性名 1</u> , <u>属性名 2</u> , … …, 属性名 n

図 4 エンティティタイプの属性の表記ルール

2. 関係スキーマの表記ルール及び関係データベースのテーブル（表）構造の表記ルール

(1) 関係スキーマの表記ルールを、図 5 に示す。

関係名 (属性名 1, 属性名 2, 属性名 3, …, 属性名 n)

図 5 関係スキーマの表記ルール

- ① 関係を、関係名とその右側の括弧でくくった属性名の並びで表す。¹⁾ これを関係スキーマと呼ぶ。
 - ② 主キーを表す場合は、主キーを構成する属性名又は属性名の組に実線の下線を付ける。
 - ③ 外部キーを表す場合は、外部キーを構成する属性名又は属性名の組に破線の下線を付ける。ただし、主キーを構成する属性の組の一部が外部キーを構成する場合は、破線の下線を付けない。
- (2) 関係データベースのテーブル（表）構造の表記ルールを、図 6 に示す。

テーブル名 (列名 1, 列名 2, 列名 3, …, 列名 n)

図 6 関係データベースのテーブル（表）構造の表記ルール

関係データベースのテーブル（表）構造の表記ルールは、(1) の ① ～ ③ で“関係名”を“テーブル名”に、“属性名”を“列名”に置き換えたものである。

注 ¹⁾ 属性名と属性名の間は“,”で区切る。

[メモ用紙]

問1 コード決済業者の決済業務及びポイント管理業務におけるデータベースの実装・運用に関する次の記述を読んで、設問に答えよ。

A社は、バーコードを用いたキャッシュレスの決済サービス、及び商品購入などに応じて一定のポイントを付与するポイントサービスを提供している。A社では、ポイントサービス利用件数の急増を受けて、システムの変更を検討している。

〔業務・現行システムの概要〕

1. サービスの概要

(1) 決済サービス

- ① 決済サービスを利用する人（以下、決済会員という）は、決済会員登録を行い、金融機関などから一定の金額を決済サービスに入金した上で、物品購入などの支払に充てる。入金及び支払に伴う残高を決済残高という。
- ② 決済サービスに加盟する事業者（以下、決済加盟店という）は、決済サービスへの加盟申請、審査を経て決済加盟店登録を受ける。
- ③ 決済会員は、スマートフォンなどの携帯端末、決済加盟店の店舗に設置されたPC、POSなどの端末（これらをまとめて、以下、端末という）を使用して、バーコードを読み取ることで支払を行う。同じ決済会員が同時に複数の支払を行うことはない。
- ④ 決済は、決済会員の決済残高から支払金額を差し引くことで完了する。
- ⑤ A社は、月に1回まとめて決済金額を決済加盟店に支払う。

(2) ポイントサービス

- ① ポイントサービスを利用する人（以下、ポイント会員という）は、ポイント会員登録を行ってポイントサービスを利用する。
- ② ポイントサービスに加盟する事業者（以下、ポイント加盟店という）は、ポイントサービスへの加盟申請、審査を経てポイント加盟店登録を受ける。
- ③ ポイント会員は、ポイント加盟店での支払、くじ引き、イベント参加などによってポイントを獲得する。
- ④ 獲得したポイントには、有効期限の年月が決まっている。獲得したポイントはポイント残高として貯めておき、支払に充てることができる。

- ⑤ 獲得したポイントは、すぐにポイント残高に反映されることもあれば、翌日以降になる場合もある。

(3) サービス間の連携

決済金額の一部をポイントで支払い、残りを決済残高から支払う場合など、両方のサービスを連携した決済を行うことがある。

2. システム構成

現行システムのシステム構成を図1に示す。決済サービス、ポイントサービスそれぞれに専用のサーバ及びRDBMSを用いている。サーバ上ではサービスに用いるアプリケーションプログラム（以下、APという）が稼働している。両サービスを連携した処理は、決済APを介して決済DB、ポイントDBの両方にアクセスする。

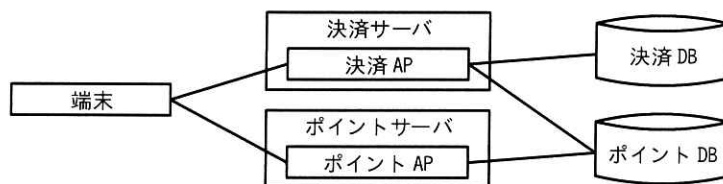


図1 現行システムのシステム構成

3. テーブル構造

決済サービス、ポイントサービスのテーブル構造をそれぞれ図2、3に、主な属性の意味・制約を表1に示す。なお、属性名の#は番号、Cはコード、Fはフラグを略した記号である。

決済加盟店（決済加盟店#，ポイント加盟店#，名称，支払先口座，…）
 店舗（決済加盟店#，店舗#，決済方式，所在地，…）
 決済会員（決済会員#，ポイント会員#，表示名，最終入金#，認証，…）
 決済残高（決済会員#，残高）
 入金履歴（決済会員#，入金#，入金日，入金金額，入金方法）
 決済履歴（履歴#，決済加盟店#，店舗#，決済会員#，決済日，決済金額，使用ポイント数）
 加盟店支払（決済加盟店#，支払年月，締め日，支払予定日，支払予定金額）

図2 決済サービスのテーブル構造（一部省略）

ポイント会員ランク (ポイント会員ランク C, 名称, 達成ポイント数)
ポイント獲得方法 (ポイント獲得方法 C, 名称)
ポイント付与率 (ポイント獲得方法 C, ポイント会員ランク C, 付与倍率)
ポイント加盟店 (ポイント加盟店#, 名称, 所在地, …)
ポイント会員 (ポイント会員#, ポイント会員ランク C, 氏名, 生年月日, 住所, 認証, …)
ポイント残高 (ポイント会員#, 有効期限年月, 残数, 有効 F)
会員ランク履歴 (ポイント会員#, 変更日, ポイント会員ランク C)
ポイント獲得履歴 (履歴#, ポイント会員#, ポイント加盟店#, ポイント獲得方法 C, 獲得日, 有効期限年月, 獲得ポイント数, 残高反映 F)
ポイント使用履歴 (履歴#, ポイント会員#, ポイント加盟店#, 使用日, 使用ポイント数)

図 3 ポイントサービスのテーブル構造 (一部省略)

表 1 主な属性の意味・制約

属性名	意味・制約
決済方式	バーコードの読取りを利用者の端末, 店舗の端末のどちらで行うかを表す区分
入金#	決済会員ごとに決済残高への入金の都度払い出す連番
入金方法	入金時に使用した銀行口座, クレジットカードなどの情報
ポイント獲得方法 C	ポイント加盟店での支払, くじ引き, イベント参加などポイント獲得の方法を表すコード
有効期限年月, 残数	ポイントが有効な期間は, 獲得方法によって異なる。“ポイント獲得履歴”テーブルには獲得したポイントの有効期限年月を設定し, “ポイント残高”テーブルには有効期限年月ごとにポイントの残数を記録する。
履歴#	“決済履歴”, “ポイント獲得履歴”, “ポイント使用履歴”テーブルの履歴#には, それぞれに一意な番号を自動的に設定する。
有効 F	有効期限年月が現在の年月以降の場合は TRUE, それよりも前のものは FALSE を設定する。
残高反映 F	獲得ポイント数を“ポイント残高”テーブルの残数に加算済みの場合は TRUE, 未加算の場合は FALSE を設定する。

4. 主な処理

各サービスの主な処理の内容を表 2 に, 表 2 中の S1, S2, P2, P3 の処理概要と SQL 文を表 3~6 に示す。各処理は一つ以上のステップから成る。表 2~6 中の二重引用符で囲んだ名前は, 図 2, 3 中のテーブル名である。

表 2 主な処理の内容

サービス	処理 ID	名称	内容
決済サービス	S1	決済残高入金処理	決済会員の指示によって金融機関からの入金実行を記録し、“決済残高”の残高に反映する。
	S2	決済残高消費処理	決済会員の決済指示を受けて、“決済残高”の残高を消費して、内容を“決済履歴”に記録する。
ポイントサービス	P1	ポイント獲得処理	ポイント会員がポイントを獲得する都度、獲得したポイント数を“ポイント獲得履歴”に記録する。
	P2	獲得ポイント反映処理	任意のタイミングで、ポイント会員のポイント獲得履歴を“ポイント残高”の有効期限年月該当行に反映する。
	P3	ポイント消費処理	ポイント使用指示を受けて、“ポイント残高”の残数を消費して、内容を“ポイント使用履歴”に記録する。

表 3 S1 の処理概要と SQL 文

ステップ	処理概要（上段）と SQL 文（下段）
S1-1	決済会員の“決済残高”の残高に入金金額を加算して更新する。
	UPDATE 決済残高 SET 残高 = 残高 + :hv3 WHERE 決済会員# = :hv1
S1-2	“決済会員”の最終入金#を更新する。
	UPDATE 決済会員 SET 最終入金# = 最終入金# + 1 WHERE 決済会員# = :hv1
S1-3	“入金履歴”の入金#に更新後の最終入金#を設定して行を追加する。
	INSERT INTO 入金履歴 (決済会員#, 入金#, 入金日, 入金金額, 入金方法) SELECT 決済会員#, 最終入金#, :hv2, :hv3, :hv4 FROM 決済会員 WHERE 決済会員# = :hv1

注記 ホスト変数 hv1～hv4 には、決済会員#, 入金日, 入金金額, 入金方法をそれぞれ設定する。

表 4 S2 の処理概要と SQL 文

ステップ	処理概要（上段）と SQL 文（下段）
S2-1	決済会員の“決済残高”の残高が決済金額以上であることを確認し、残高が不足していれば、戻り値に残高不足エラーを設定して処理を終了する。
	SELECT 残高 FROM 決済残高 WHERE 決済会員# = :hv1 AND 残高 >= :hv5 FOR UPDATE
S2-2	決済会員の“決済残高”の残高から決済金額を減算して更新する。
	UPDATE 決済残高 SET 残高 = 残高 - :hv5 WHERE 決済会員# = :hv1
S2-3	“決済履歴”の各列に対応するホスト変数の値を設定して行を追加する。
	INSERT INTO 決済履歴 (決済加盟店#, 店舗#, 決済会員#, 決済日, 決済金額, 使用ポイント数) VALUES(:hv2, :hv3, :hv1, :hv4, :hv5, :hv6)

注記 ホスト変数 hv1～hv6 には、決済会員#, 決済加盟店#, 店舗#, 決済日, 決済金額, 使用ポイント数をそれぞれ設定する。

表 5 P2 の処理概要と SQL 文（未完成）

ステップ	処理概要（上段）と SQL 文（下段）
P2-1	指定されたポイント会員#について“ポイント獲得履歴”からポイント残高未反映の行の獲得ポイント数を有効期限年月ごとにまとめて，“ポイント残高”の該当行があれば残数に獲得ポイント数を加算して更新し，行がなければ残数に獲得ポイント数を設定して行を追加する。 <pre> MERGE INTO ポイント残高 A USING (SELECT ポイント会員#, 有効期限年月, a AS 加算ポイント数 FROM ポイント獲得履歴 WHERE ポイント会員# = :hv1 AND b GROUP BY ポイント会員#, 有効期限年月) B ON (A.ポイント会員# = B.ポイント会員# AND A.有効期限年月 = B.有効期限年月) WHEN MATCHED THEN UPDATE SET c WHEN NOT MATCHED THEN INSERT (ポイント会員#, 有効期限年月, 残数, 有効 F) VALUES(d, TRUE) </pre>
P2-2	“ポイント獲得履歴”の該当行の残高反映 F の値を更新する。 <pre> UPDATE ポイント獲得履歴 SET 残高反映 F = TRUE WHERE ポイント会員# = :hv1 AND 残高反映 F IS FALSE </pre>

注記 ホスト変数 hv1 にはポイント会員#を設定する。

表 6 P3 の処理概要と SQL 文

ステップ	処理概要（上段）と SQL 文（下段）
P3-1	指定されたポイント会員#について“ポイント残高”から有効期限内の残数を集計して求めた使用可能ポイント数が使用ポイント数に満たない場合は，戻り値にポイント不足エラーを設定して処理を終了する。 <pre> SELECT SUM(残数) AS 使用可能ポイント数 FROM ポイント残高 WHERE ポイント会員# = :hv1 AND 有効 F IS TRUE </pre>
P3-2	①の SQL 文に基づくカーソルによって，“ポイント残高”からポイントが有効な行を有効期限年月の昇順に読み込み，行ごとに残数を使用ポイント数に合わせて消し込む。ホスト変数 lv1 に消し込むポイント数（以下，消込ポイント数という）を，lv2 に有効期限年月を設定して②の更新を行い，使用ポイント数を全て消し込むまで繰り返す。 <pre> ① SELECT 有効期限年月, 残数 FROM ポイント残高 WHERE ポイント会員# = :hv1 AND 有効 F IS TRUE ORDER BY 有効期限年月 ASC ② UPDATE ポイント残高 SET 残数 = 残数 - :lv1 WHERE ポイント会員# = :hv1 AND 有効期限年月 = :lv2 </pre>
P3-3	“ポイント使用履歴”の各列に対応するホスト変数の値を設定して行を追加する。 <pre> INSERT INTO ポイント使用履歴 (ポイント会員#, ポイント加盟店#, 使用日, 使用ポイント数) VALUES(:hv1, :hv2, :hv3, :hv4) </pre>

注記 ホスト変数 hv1～hv4 には，ポイント会員#，ポイント加盟店#，使用日，使用ポイント数をそれぞれ設定する。

5. 現行システムで使用する RDBMS の排他制御機能

(1) トランザクションの ISOLATION レベルごとの排他制御は，次のとおりである。

- ① READ COMMITTED では、行の参照時に対象行の共有ロックを取得し、参照終了時に解放する。行の更新時に対象行の専有ロックを取得し、トランザクション終了時に解放する。
 - ② REPEATABLE READ では、行の参照時に対象行の共有ロックを、行の更新時に対象行の専有ロックを取得し、トランザクション終了時に解放する。
 - ③ SERIALIZABLE は、行の参照時又は更新時に表単位の専有ロックを取得し、トランザクション終了時に解放する。
- (2) READ COMMITTED 及び REPEATABLE READ では、SELECT 文に FOR UPDATE 句を指定すると、行の参照時に対象行の専有ロックを取得し、トランザクション終了時に解放する。

[現行システムの決済処理]

決済処理では、表 2 中の処理 S2 及び P3 を一つのトランザクションとして実行するために、2 相コミットプロトコルを用いている。

1. 決済処理の流れ

決済処理の正常実行時のシーケンス図を図 4 に示す。図中の①～⑬は実行の順番を、時点 A、B は実行の流れにおける特定の時点を表している。

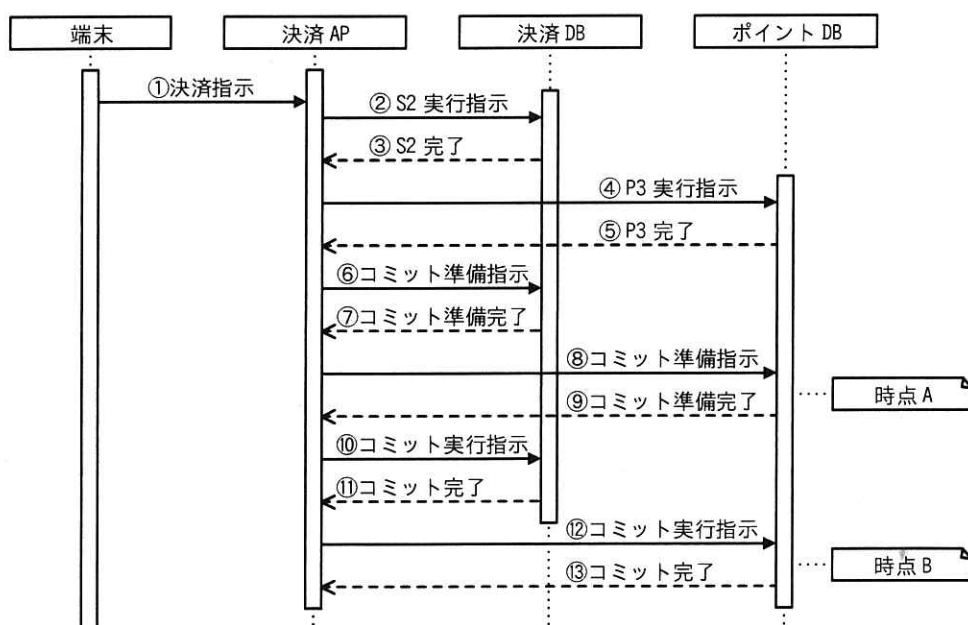


図 4 決済処理の正常実行時のシーケンス図（未完成）

2. 障害からの回復

決済処理の実行中に障害によってポイント DB が異常終了する状況を想定した障害・回復の事例を表 7 にまとめた。表中の時点 A, B は、図 4 中の時点 A, B に対応している。

表 7 決済処理中の障害・回復の事例（未完成）

事例	内容
事例 1	決済処理中の時点 A でポイント DB が障害によって異常終了した。障害除去後に再始動したところ、ポイント DB にはコミット準備のログがあり、トランザクションは未コミットの状態となった。決済 AP は、ポイント DB に状態を問い合わせ、決済 DB 及びポイント DB に e を行った。
事例 2	決済処理中の時点 A でポイント DB が障害によって異常終了した。障害除去後に再始動したところ、ポイント DB にはコミット準備のログがなく、トランザクションはロールバックされた。決済 AP は、ポイント DB に状態を問い合わせ、決済 DB に f を行った。
事例 3	決済処理中の時点 B でポイント DB が障害によって異常終了した。障害除去後に再始動したところ、ポイント DB はロールフォワードによってデータベースを回復した。決済 AP は、ポイント DB に状態を問い合わせ、 g した。
事例 4	決済処理中の時点 A でポイント DB が障害によって異常終了した。⑦復旧までの時間が長くなることで、業務上の不都合が生じたので、決済 DB において仕掛り中のトランザクションをヒューリスティックな決定によって全てコミットした。障害除去後の再始動において、ポイント DB は未コミットのトランザクションをロールバックした。

〔システム変更案の検討〕

ポイントサービスでは、EC サイトを運営する加盟店の増加によって、サービス利用件数が急増し、ポイント DB の応答性能、可用性の低下が問題になっている。その対策として、次のシステム変更案を検討している。

1. システムの構成変更

検討中のシステム構成を図 5 に示す。業務機能別に会員サービス、決済サービス、ポイントサービスに分け、サービスごとに専用のサーバとデータベースを配置する。各サービスには独立した疎結合の AP 及びデータベースを用いて、サービス内部の実装は隠蔽する。サービスに連携するアプリケーションプログラミングインタフェース（以下、API という）だけを公開し、API を呼び出すごとに単一のトランザクションとして処理する。端末からのアクセス時には、会員サービスを入り口にして他のサービスに連携する。

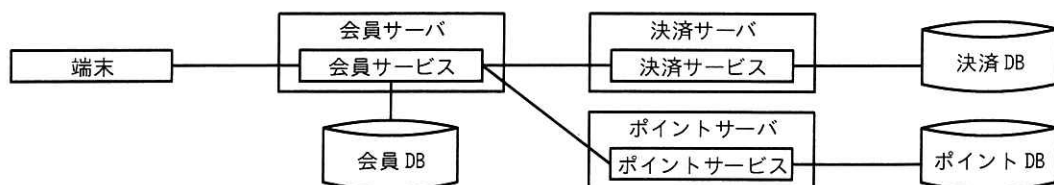


図5 検討中のシステム構成

(1) 会員サービス

- ① 決済会員，ポイント会員の会員情報の登録，端末からの接続認証など会員に関わる機能をもつ AP，及び後述する複数サービスが関与する処理の実行制御を担う AP を配置する。
- ② 会員 DB には，現行システムと同じ RDBMS を使用する。“決済会員”，“ポイント会員ランク”及び“ポイント会員”テーブルを配置し，登録，認証に使用する。会員情報の登録，変更は他のサービスにも連携する。

(2) 決済サービス

- ① 決済 AP は，ポイントサービスと連携する処理を無効にすることを除いて変更せず，API を決済 AP に連携することで，特定の機能を呼び出して実行する方式に変更する。
- ② 決済 DB は，現行 RDBMS のデータベースをそのまま使用する。

(3) ポイントサービス

- ① ポイント AP は応答性能，可用性を高めるために再構築する。API をポイント AP に連携することで，特定の機能を呼び出して実行する方式に変更する。
- ② ポイント DB を NoSQL に分類されるデータベースに変更し，図3中の全てのテーブルのデータを移行する。新しいポイント DB は，次の特徴をもつ。
 - ・ワイドカラムデータストアをもち，複数ノードにデータを分散して配置することで，高速性，高可用性を実現している。
 - ・RDBMS におけるテーブルと同様に，列及び主キーを定義できる。外部キーは定義できない。CRUD 操作命令を行う言語をサポートしていて，表3～6の SQL 文と同様の操作を記述できる。
 - ・複数の CRUD 操作命令をまとめて一つのトランザクションとして実行する機能はなく，CRUD 操作命令の終了時点でデータへの操作が確定する。

- ・ 2 相コミットを実現する機能はない。

2. 処理方式の設計

現行の 2 相コミットに替えて、次の処理方式を検討している。検討に当たって、ポイント DB に格納するデータの設計には、RDBMS のテーブル構造を用いる。

(1) 処理方式

- ① 複数サービスが関与する処理の実行制御を担う AP（以下、オーケストレーターという）を会員サーバに配置する。
- ② 端末からの要求ごとにオーケストレーターを割り当て、他サービスへの要求送信と結果受信とを同期的に行うことで、処理を順に実行していく。
- ③ オーケストレーターは、ワークフローの機能を持ち、登録した処理を状態に応じて実行していく。タイムアウトなどの例外によって処理が異常終了した場合には、処理が成功するまで一定回数のリトライを行う。
- ④ リトライを経ても処理が失敗した場合、オーケストレーターは、それ以前に終了した処理について、ワークフローの逆方向に取消処理を順に実行する。取消処理では、データの変更を伴う場合に、そのデータ変更を取り消すトランザクション（以下、補償トランザクションという）を実行する。

(2) 実行制御テーブルの追加

会員 DB に“決済指示”テーブルを追加し、オーケストレーターによるトランザクションの実行状態の記録、実行の制御に用いる。追加するテーブルのテーブル構造を図 6 に示す。“決済指示”テーブルの決済指示#は、決済指示ごとに一意な番号を自動採番し、トランザクションの識別子として使用する。オーケストレーターは、処理実行の指示及び補償トランザクション実行の指示に合わせて、“決済指示”テーブルに対応する行の全列の値を他サービスに連携する。

決済指示（決済指示#、加盟店#、店舗#、決済会員#、ポイント会員#、決済金額、使用ポイント数、決済指示日時、処理 ID、ステップ、モード、状態）

注記 1 処理 ID、ステップには仕掛り中の処理について表 2～6 の処理 ID、ステップのいずれかを設定する。モードには通常モード、取消モードのいずれか、状態には進行中、完了、失敗のいずれかを表す値を設定する。

注記 2 外部キーは会員 DB に配置するテーブルとの間にだけ設定する。

図 6 追加するテーブルのテーブル構造

(3) 決済処理の流れ

決済処理を次のように実行することにした。

- ① 決済サービスでは、S2 を一つの API として実行する。
- ② ポイントサービスでは、P3 のステップ P3-1～P3-3 のそれぞれに対応する API を用意し、会員サービスは API を順番に実行する。
- ③ 会員サービスでは、他のサービスへの実行指示に対する処理結果を“決済指示”テーブルの状態に記録していく。

検討中の決済処理の正常実行時のシーケンス図を図 7 に示す。図 7 中の①～⑮は実行の順番を、時点 C～E は実行の流れにおける特定の時点を表している。

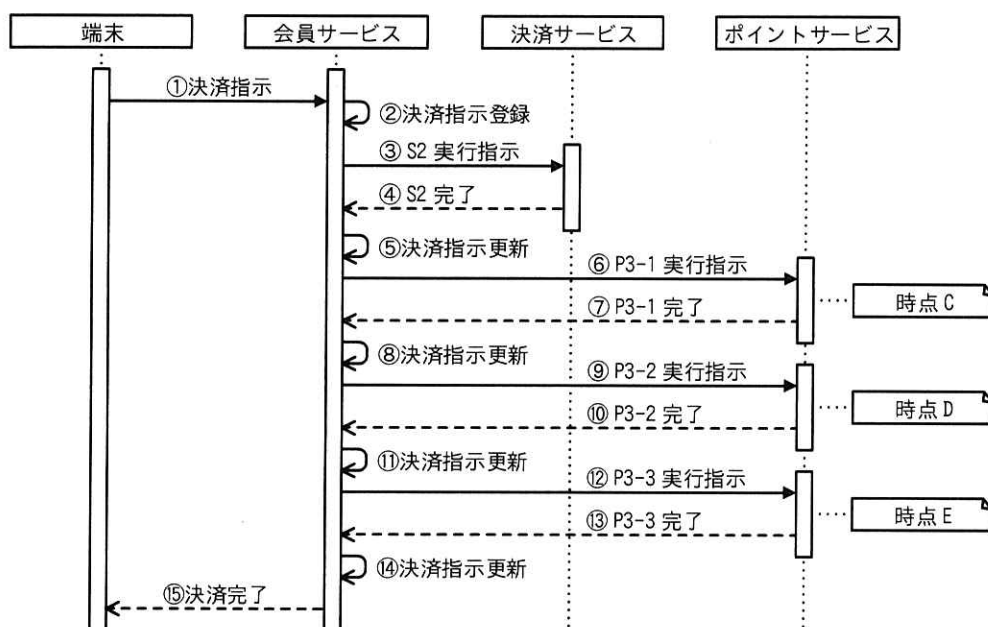


図 7 検討中の決済処理の正常実行時のシーケンス図

3. テーブル構造及び処理の変更

(1) リトライへの対応

図 7 中の実行指示に対応する処理の中には、リトライによる実行指示を受けたときに、重複更新の発生が懸念されるものがある。これを避けるためのテーブル構造及び処理の変更を行う。①決済サービスでは、対策として図 2 中の一つのテーブルに列を一つ追加し、処理 S2 の先頭にステップを一つ追加する。ポイントサービスでも同様の対策を行う。

(2) ポイント DB の整合性確保

表 6 の処理 P3 は、ステップごとにデータ操作が確定するので、ステップ実行の合間に他のトランザクションから同じ会員のデータ更新があると、ポイント DB 内のデータに不整合が生じるおそれがある。これを防ぐために“ポイント会員”テーブルに更新中 F の列を追加して TRUE 又は FALSE を設定した上で、㉔ P2 及び P3 に、それぞれ同じ内容の処理を行うステップを追加する。

4. 補償トランザクションの設計

ここまでの検討結果を踏まえて、決済処理の補償トランザクションを表 8 にまとめた。表 8 中の処理 ID は表 2 の処理 ID、表 8 中のステップは表 4、6 のステップに対応し、補償トランザクションは、図 7 中の実行指示と同じ単位で実行するものとする。

表 8 決済処理の補償トランザクション（未完成）

処理 ID	ステップ	補償要否	補償可否	説明 (補償不要の理由／補償不可の理由／補償処理の概要)
S2	S2-1	不要	-	参照だけでデータの変更がない。
	S2-2	要	可	h
	S2-3	要	可	i
P3	P3-1	不要	-	参照だけでデータの変更がない。
	P3-2	要	不可	㉔使用ポイント数を有効期限ごとの残数から消費した消込ポイント数が分からない。
	P3-3	要	可	

注記 1 補償要否には変更の取消しが必要ならば“要”を、それ以外は“不要”を記入する。

注記 2 補償可否には、現行システムのテーブル構造を前提に変更の取消しが可能ならば“可”を、可能でなければ“不可”を、補償が不要であれば“-”を記入する。

注記 3 説明には、補償不要の場合は不要の理由を、補償不可の場合は取消しができない理由を、それ以外は、補償処理の概要を記入する。

注記 4 網掛け部分は表示していない。

5. 障害からの回復

表 8 の補償要否が“要”のステップは、全て補償可否が“可”となるように対策を行った上で、決済処理実行中にポイントサービスが障害によって停止する状況を想定した障害・回復の事例を表 9 にまとめた。表中の時点 C～E は、図 7 中の時点 C～E に対応する。

表 9 決済処理中の障害・回復の事例

事例	内容
事例 5	ポイントサービスが時点 C で障害によって停止した。オーケストレーターは、P3-1 実行指示を一定回数リトライ後、S2 の取消実行指示を行った。
事例 6	ポイントサービスが時点 D で障害によって停止した。オーケストレーターは、P3-2 実行指示を一定回数のリトライ後、P3-1、S2 の取消実行指示を順に行った。
事例 7	ポイントサービスが時点 E で障害によって停止した。オーケストレーターは、P3-3 実行指示を一定回数のリトライ後、P3-2 から S2 までの取消実行指示を順に行った。

設問 1 「業務・現行システムの概要」について答えよ。

- (1) 表 5 中の a ～ d に入れる適切な字句を答えよ。
- (2) 一貫性の観点から、処理実行時の ISOLATION レベルについて答えよ。
 - (a) 表 3 の処理 S1 の ISOLATION レベルは READ COMMITTED が適切である。
REPEATABLE READ である必要がない理由を 30 字以内で具体的に答えよ。
 - (b) 表 6 の処理 P3 の ISOLATION レベルは SERIALIZABLE が適切である。
REPEATABLE READ では不都合がある理由を 30 字以内で具体的に答えよ。

設問 2 「現行システムの決済処理」について答えよ。

- (1) 図 4 について答えよ。
 - (a) “①決済指示” に対応する “決済完了” の応答が欠けている。トランザクションが正常に終了するとみなした時点で決済 AP から端末に応答を返すとしたら、③～⑬のどのメッセージの受信後か。番号を一つ答えよ。
 - (b) ポイント DB が④受信後の P3 実行において、ステップ P3-1 でポイント不足エラーを返した場合、⑥及び⑧で同じメッセージを送信してトランザクションを終了する。送信するメッセージの内容を答えよ。
 - (c) 時点 A と時点 B とでは、トランザクションの状態が異なる。状態の相違点を 45 字以内で具体的に答えよ。
- (2) 表 7 について答えよ。
 - (a) 事例 1～3 の e ～ g に入れる適切な字句を次の中から選択して答えよ。

コミット準備指示, コミット実行指示, ロールバック実行指示,
トランザクションを終了

(b) 事例 4 の下線㉗について、発生したと考えられる業務上の不都合を 30 字以内で具体的に答えよ。

(c) 事例 4 では、最終的にデータの状態はどのようなになるか。40 字以内で具体的に答えよ。

設問 3 「システム変更案の検討」について答えよ。

(1) “3. テーブル構造及び処理の変更”について答えよ。

(a) 本文中の下線㉘について、列を追加する図 2 中のテーブル名、追加する列名を答えよ。また、追加するステップの適切な処理概要を 40 字以内で答えよ。

(b) 本文中の下線㉙について、処理 P2 及び P3 の先頭に共通して追加するステップの適切な処理概要を 60 字以内で、末尾に共通して追加するステップの適切な処理概要を 20 字以内で答えよ。

(2) “4. 補償トランザクションの設計”について答えよ。

(a) 表 8 中の h , i に入れる適切な補償処理の概要をそれぞれ 40 字以内で答えよ。

(b) 表 8 中の下線㉚への対策として、ポイント DB に図 8 の“ポイント消込履歴”テーブルを追加して補償を可能にする。図 8 中の j に入れる一つ又は複数の適切な列名を答えよ。なお、主キーの構成列には巻頭の表記ルールに従って実線の下線を付けよ。

ポイント消込履歴 (j , 消込ポイント数)
--

図 8 “ポイント消込履歴”テーブルのテーブル構造（未完成）

(3) “5. 障害からの回復”について答えよ。

表 9 中の事例 5～7 のうち、ポイント DB 内のデータが一時的に不整合になる事例の一つ答えよ。また、不整合の内容を 50 字以内で具体的に答えよ。

問2 宅配ピザチェーンにおける店舗への資材配送業務の概念データモデリングに関する次の記述を読んで、設問に答えよ。

B社は、宅配ピザチェーンである。B社では、店舗資材配送システムの再構築を進めており、業務分析を行い、概念データモデル及び関係スキーマを設計した。

〔現状の業務分析の結果〕

1. 社内外の組織に関連する資源

- (1) 全国を対象に、人口分布及び道路状況を考慮して幾つかの郵便番号の指す区域を束ねて地区として設定している。地区は地区コードで識別する。
- (2) 首都圏及び地方中核都市のベッドタウンに重点的に店舗を展開しており、約500店がある。店舗は店舗コードで識別し、いずれかの地区に立地する。
- (3) 物流センター（以下、DCという）は全国に12あり、DCコードで識別する。
 - ① DCは、幾つかの地区を束ねて、それらにある店舗を配送対象にしている。
 - ② DCは、店舗への資材の配送及び調達先からの資材の受入れを行う。
 - ③ DC内の倉庫には80～120程度の区画があり、DCごとに区画番号で識別する。区画のほとんどは冷蔵保管の機能で、残りの一部が常温保管と冷凍保管の機能となっている。区画がいずれの機能かは、機能区分で分類する。
 - ④ 区画は、幾つかの棚に仕切られており、棚が幾つあるかの棚数をもつ。
 - ⑤ 棚は、区画ごとに棚番号で識別する。
- (4) DCから店舗へ配送するトラックのルートを整備している。ルートはDCごとのルート番号で識別し、車両番号をもつ。
 - ① ルートの店舗への納入は13～16時の間に終わるように設定している。
 - ② ルートごとに店舗への納入順を決めている。
- (5) 資材の調達先をベンダーと呼び、ベンダーコードで識別する。

2. 商品及び資材に関連する資源

- (1) 店舗で販売する商品は、商品コードで識別し、商品名をもつ。
 - ① 商品の種類には、ピザ、追加トッピング、サイドメニュー、ドリンクがあり、商品種類区分で分類する。
 - ② 商品には、ピザであればマルゲリータ、ハワイアンなど、サイドメニューであればフライドチキン、サラダなどがある。

- ③ ピザ及び追加トッピングには、S、M、Lの3サイズがある。
 - ④ ドリンクには、MサイズとLサイズの2サイズがある。
 - ⑤ 全てのサイドメニューはMサイズの1サイズだけである。複数のサイドメニューの組合せ及び入れる個数の違うものは、別の商品としている。
 - ⑥ ピザのようにサイズのある商品もあるが、商品はサイズ別には登録しない。商品には、その商品に幾つサイズがあるかのサイズ数をもつ。
 - ⑦ ハーフ＆ハーフなどの組合せ商品は、マルゲリータ＆ハワイアンのように商品名を付けてあらかじめ全ての組合せを登録する。
 - ⑧ 商品の販売には、定番商品、期間限定商品、新商品があり、商品販売区分で分類する。
 - ・ 定番商品は、その商品の前年日販数をもつ。
 - ・ 期間限定商品は、販売期間の開始と終了の年月日をもつ。
 - ・ 新商品は、3か月の評価期間の売れ行きで判断し、定番化するか否か決めるもので、販売開始の年月日と定番化判断の結果をもつ。
- (2) 資材は資材コードで識別し、資材名、調達先ベンダーコードなどをもつ。
- ① 資材には、食材と梱包材がある。梱包材は、調理したピザやサイドメニューの梱包に用いる。資材がいずれであるかは食材梱包材区分で分類する。
 - ② 資材が、冷蔵保管、常温保管、冷凍保管のいずれの対象かを保管温度帯区分で分類する。
 - ③ 定番商品に用いる食材及び梱包材を共通資材と呼ぶ。共通資材には、食材であればピザ台、チーズ、サラミ、ハム、カット野菜、食用油などがあり、梱包材であれば商品共通の箱やポリ袋などがある。3か月平均単価をもつ。
 - ④ 期間限定商品と新商品に固有で用いる食材及び梱包材を企画資材と呼ぶ。企画期間単価をもつ。
 - ⑤ 資材が、共通資材であるか企画資材であるかは、共通企画区分で分類する。
 - ⑥ 資材には計量単位を設定する。枚、グラム、個、ミリリットルなどがある。
- (3) 資材をベンダーから購入する荷姿を購入荷姿と呼ぶ。主に容量が50リットル程度の段ボール箱である。
- (4) DC から資材を店舗に配送する荷姿を配送荷姿と呼ぶ。配送の最小単位であり、購入荷姿の段ボール箱を開梱した小箱、袋、ボトルなどである。DC では、この

単位で保管するので保管荷姿とも呼ぶ。

(5) 資材には、購入荷姿の入数と配送荷姿の入数をもつ。DC から店舗への配送は、配送荷姿の入数の整数倍で行う。

(6) 資材を調達するベンダーは、資材ごとに1社に決めている。

3. 資源の組合せで定める資源及び在庫

(1) 商品とサイズの組合せで、資材ごとの使用量をレシピとして登録している。
使用量は調理時のロスを考慮している。

(2) DC ごとに資材の在庫をもつ。

4. 商品企画のやり方

(1) 商品企画の対象は期間限定商品と新商品である。期間限定商品と新商品で商品企画のやり方は同じである。

(2) 期間限定商品のレシピ及び販売期間はB社の本部が定める。

(3) 本部は、新商品の評価期間後に定番商品にするか廃番にするかを定める。新商品を定番商品にする場合、新たに定番商品として登録する。

5. DC からベンダーへの発注のやり方

定量発注と計画発注がある。どちらも1日に高々1回行う。

(1) 定量発注のやり方

① 共通資材を対象として、定量発注を実施している。

② B社が採用している定量発注は、DCごと資材ごとに発注ロットサイズ、発注リードタイムを調達条件として取り決めておき、DCの实在庫数が、DCごと資材ごとに設定した基準在庫数を下回ったときに、発注ロットサイズ分の発注を日次で行う方式である。発注ロットサイズは購入荷姿の入数の整数倍にする。調達条件は必要に応じてベンダーと協議する。

③ 定量発注では、確かに在庫を確認して発注したものが否かを表す在庫確認サイン及び発注年月日を記録する。

(2) 計画発注のやり方

① 企画資材を対象として、計画発注を実施している。

② 計画発注では、全販売期間にわたって週次の調達量をあらかじめ発注する。
発注年月日、発注数及び希望納期年月日を記録する。

6. DC での在庫保管のやり方

(1) 倉庫内の資材の保管は、固定ロケーション方式を採用している。

① 倉庫の区画に対して一つ又は複数の資材を割り当てるが、一つの資材は一つの区画だけに収めている。DC ごと資材ごとに保管区画を決めている。

② 資材は一つ又は複数の棚に割り当てるが、一つの棚には一つの資材だけを割り当てる。

(2) 同じ資材は、先に入庫したものから先に出庫する先入先出を行う。

① 資材に割り当てた棚には、次に入庫する棚を示す後続棚番号をもたせ、同じ資材に割り当てた棚の中で後続棚番号が一巡するように設定する。棚には、同じことが目視で判別できるように、棚番号、後続棚番号、割り当てている資材コードと資材名を記載したプレートを掲示する。

② 最後に入庫した棚番号を最終入庫棚番号として記録する。最後に出庫した棚番号を最終出庫棚番号として記録する。最終入庫棚番号と最終出庫棚番号は、DC の資材在庫にもつ。これによって、次の入庫又は出庫をどの棚番号から行えばよいかを知ることができる。

7. 入荷及び入庫のやり方

(1) 入荷では、ベンダーの納品を受けて資材ごとに現物を確認し、入荷年月日、検品サイン、どの発注に基づく入荷かの対応を記録する。

① 定量発注による入荷では、入荷時分を記録する。

② 計画発注による入荷はベンダーの都合で分納され得る。入荷数を記録する。

(2) 入荷した段ボール箱を開梱して保管荷姿の現品を入庫する。入庫は入荷の単位に行い、入庫時分を記録する。

(3) 保管荷姿の現品には、資材の情報を記したバーコードラベルを貼付する。

8. DC での出庫のやり方

(1) DC では毎営業日の 8～11 時に出庫を行う。出庫は摘取方式で行う。摘取方式とは、店舗ごとの出庫要求に従って倉庫を回って資材を出庫し、店舗ごとに用意したかご台車に積むやり方である。

(2) 出庫に先立って出庫要求をまとめる。出庫要求では、共通資材の補充要求、企画資材の計画要求、店舗の追加要求の 3 種類の資材要求を作り、資材要求から店舗ごと資材ごとに対象の配送年月日の出庫要求を導く。3 種類の資材要求

は要求区分で分類する。

- (3) 共通資材の補充要求は、各店舗の販売実績から求めることのできる共通資材ごとの使用量を必要量として、配送の 1 営業日前に次の手順で求めたものである。共通資材の補充要求は、算出時分をもつ。
 - ① 店舗の営業終了時に、各店舗から当日の販売実績として、販売年月日、店舗コード、商品コード、販売サイズ、販売数を連携してもらう。
 - ② 連携された販売実績とレシビを基に、全ての資材の使用量を求める。
 - ③ ②の結果のうち共通資材だけを対象に、店舗ごと資材ごとに集計して共通資材の使用量を求め、配送荷姿の入数で除して四捨五入することで配送荷姿での必要数とし、資材要求数を導く。
- (4) 企画資材の計画要求は、販売期間及び各店舗の販売実績を勘案して求めた営業日ごと企画資材ごとの必要量を DC が事前に登録したものである。登録年月日を記録する。
- (5) 店舗の追加要求は、店舗責任者が事前に登録してある共通資材の補充要求と企画資材の計画要求を前提に判断し、全ての資材について追加に必要な資材を要求する。登録年月日時分を記録する。
- (6) 出庫は、導いた出庫要求に基づいて行う。店舗ごとに出庫すべき資材を全て出庫し終わったら出庫完了時分を記録し、内訳の出庫実績数を記録する。

9. DC での出荷のやり方

- (1) 店舗へのお荷を毎営業日に 1 回、13 時から実施する。
- (2) 出荷では、配送車が出発するごとに配送年月日、出荷時分を記録する。
- (3) 出荷の店舗ごとの内訳では対象店舗を記録する。
- (4) 出荷の店舗ごと資材ごとの内訳として出荷店舗資材明細を作り、どの出庫の内訳を対応させたかとお荷実績数を記録する。

10. 店舗への納入のやり方

- (1) ルートの納入順に店舗へ納入を行う。納入は、かご台車のまま納品書を添えて店舗に引き渡し、店舗が確認することで完了する。
- (2) 納入は、出荷店舗明細と対応付け、納入時分と店舗の確認サインを記録する。
- (3) 店舗からは、前回の納入で用いたかご台車を回収する。かご台車の回収は特に記録しない。

〔再構築における店舗への出庫及び出荷のやり方の変更〕

1. 変更の背景

B 社は、創業以来 DC での出庫は摘取方式で行ってきたが、DC 当たりの店舗数が増えてきたので、作業効率を上げるために種まき方式に変更することにした。

2. 変更する出庫のやり方

- (1) 種まき方式とは、資材ごとに全ての店舗分をまとめて一度に出庫し、出庫した資材を各店舗向けのかご台車に種まきをするように仕分けるやり方である。
- (2) 共通資材の補充要求の求め方、企画資材の計画要求の登録の仕方、店舗の追加要求の仕方の3種類の資材要求数の導き方は変えず、資材要求から DC ごと資材ごとに対象配送年月日の出庫要求を導くように変更する。
- (3) 出庫要求とは別に、出荷の段取りとして資材要求から店舗ごと資材ごとに対象配送年月日の出荷予定明細を導き、出荷予定数をもつ。
- (4) 出庫は区画ごとに実施し、出庫明細で区画内の資材を出庫要求に基づいて出庫する。出庫では配送年月日と出庫完了時分を記録し、出庫明細で出庫要求との対応と出庫実績数を記録する。

3. 出庫のやり方の変更に伴う出荷のやり方の変更

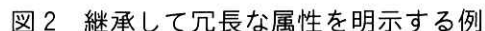
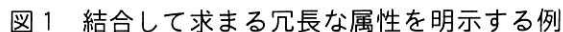
- (1) 出荷の店舗ごと資材ごとの内訳で、どの出荷予定明細に基づいたか、どの出庫明細と対応付けたかを記録する。
- (2) 出荷店舗資材明細に出荷実績数を記録する。

〔概念データモデルと関係スキーマの設計〕

1. 概念データモデル及び関係スキーマの設計方針

- (1) 関係スキーマは第3正規形にし、多対多のリレーションシップは用いない。
- (2) リレーションシップが1対1の場合、意味的に後からインスタンスが発生する側に外部キー属性を配置する。
- (3) 概念データモデルでは、リレーションシップについて、対応関係にゼロを含むか否かを表す“○”又は“●”は記述しない。
- (4) 実体の部分集合が認識できる場合、その部分集合の関係に固有の属性があるときは部分集合をサブタイプとして切り出す。
- (5) サブタイプが存在する場合、他のエンティティタイプとのリレーションシッ

- (6) 概念データモデル及び関係スキーマは、マスター及び在庫領域と、トランザクション領域を分けて作成し、マスターとトランザクションの間のリレーションシップは記述しない。
- (7) 関係スキーマでは、可読性を重視するので自然キー（ナチュラルキー）だけを用い、代理キー（サロゲートキー）は用いない。
- (8) 自然キーを参照する外部キーには冗長な属性が必要になる場合がある。この場合、属性名の前後を“[”と“]”で挟んで明示する。結合して求まる冗長な属性を明示する例を図 1 に示す。また、スーパータイプからサブタイプに継承した冗長な属性を明示する例を図 2 に示す。



(1) 「現状の業務分析の結果」を基に設計した現状のマスター及び在庫領域の概念データモデルを図 3 に、現状のトランザクション領域の概念データモデルを図 4 に、現状のマスター及び在庫領域の関係スキーマを図 5 に、現状のトランザクション領域の関係スキーマを図 6 に示す。

- (2) 〔再構築における店舗への在庫及び出荷のやり方の変更〕を基に，トランザクション領域の概念データモデルと関係スキーマの一部を変更した。図 4 に対して変更を検討する範囲を示した概念データモデルを図 7 に，変更した概念データモデルを図 8 に，変更した関係スキーマを図 9 に示す。

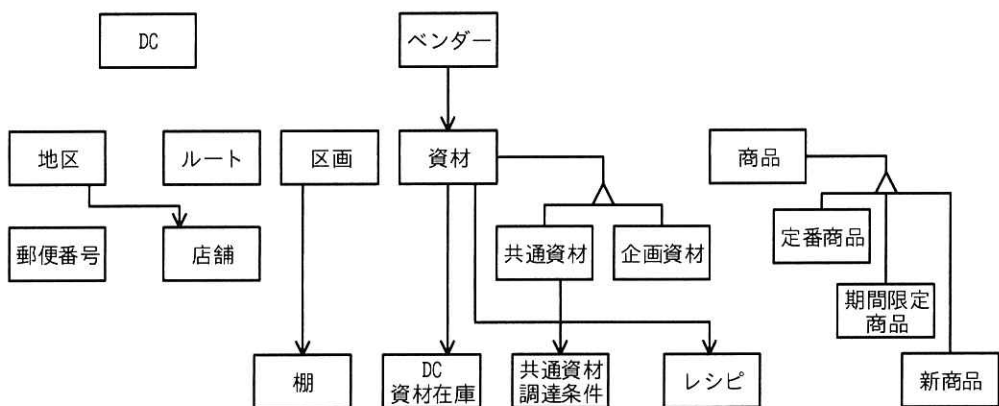


図3 現状のマスター及び在庫領域の概念データモデル（未完成）

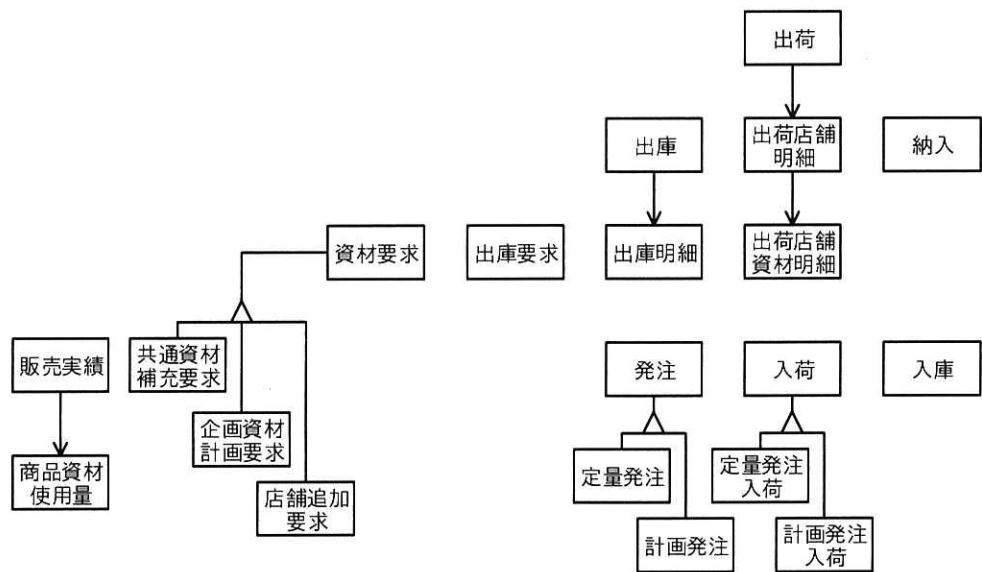


図4 現状のトランザクション領域の概念データモデル（未完成）

DC (DC コード, DC 名)
 地区 (地区コード, 地区名,)
 郵便番号 (郵便番号,)
 ルート ()
 店舗 (店舗コード, 店舗名,)
 区画 ()
 棚 ()
 ベンダー (ベンダーコード, ベンダー名)
 資材 (資材コード, 資材名, 調達先ベンダーコード, 食材梱包材区分, 保管温度帯区分, 共通企画区分,
 計量単位, 購入荷姿入数, 配送荷姿入数)
 共通資材 (資材コード, 3 か月平均単価)
 企画資材 (資材コード, 企画期間単価)
 共通資材調達条件 (資材コード,)
 商品 (商品コード, 商品種類区分, 商品名, サイズ数, 商品販売区分)
 定番商品 (商品コード, 前年日販数)
 期間限定商品 (商品コード, 販売開始年月日, 販売終了年月日)
 新商品 (商品コード, 販売開始年月日, 定番化判断結果)
 レシピ (資材コード,)
 DC 資材在庫 (DC コード, 資材コード,)

図 5 現状のマスター及び在庫領域の関係スキーマ (未完成)

販売実績 (販売年月日, 店舗コード, 商品コード, 販売サイズ, 販売数)
 商品資材使用量 (販売年月日, 店舗コード, 商品コード, 販売サイズ, 資材コード, 資材使用量)
 資材要求 (, 資材要求数)
 共通資材補充要求 (, 算出時分)
 企画資材計画要求 (, 登録年月日)
 店舗追加要求 (, 登録年月日時分)
 出庫要求 (, 出庫要求数)
 出庫 (, 出庫完了時分)
 出庫明細 (, 出庫実績数)
 出荷 (, 出荷時分)
 出荷店舗明細 ()
 出荷店舗資材明細 (, 出荷実績数)
 納入 (, 納入時分, 店舗確認サイン)
 発注 (, 共通企画区分)
 定量発注 (, 在庫確認サイン)
 計画発注 (, 発注数, 希望納期年月日)
 入荷 (DC コード, 入荷年月日, 資材コード, , 共通企画区分, 検品サイン)
 定量発注入荷 (DC コード, 入荷年月日, 資材コード, , 入荷時分)
 計画発注入荷 (DC コード, 入荷年月日, 資材コード, , 入荷数)
 在庫 (, 在庫時分)

図 6 現状のトランザクション領域の関係スキーマ (未完成)

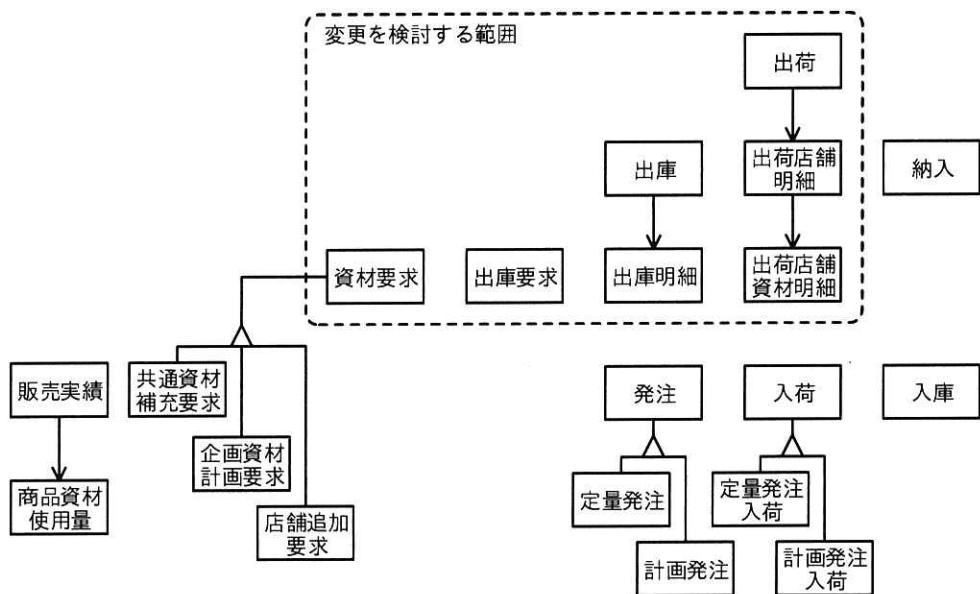


図7 変更を検討する範囲を示した概念データモデル（図4の再掲，未完成）



図8 変更した概念データモデル（未完成）

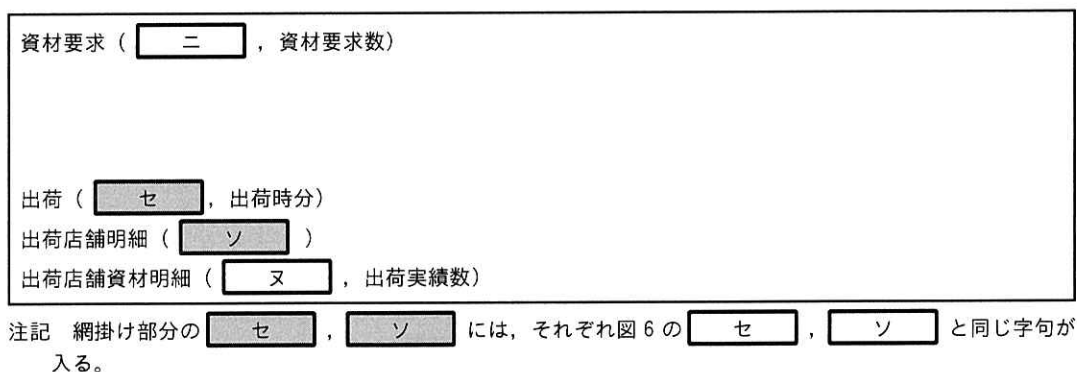


図9 変更した関係スキーマ（未完成）

解答に当たっては、巻頭の表記ルールに従うこと。また、エンティティタイプ名、関係名、属性名は、本文中の字句を用いて適切な名称とすること。関係スキーマに入れる属性名を答える場合、主キーを表す下線、外部キーを表す破線の下線についても答えること。

設問 1 「現状の業務分析の結果」を基にした概念データモデル及び関係スキーマについて、次の問いに答えよ。

なお、解答に当たっては、本文中に示した設計方針に従うこと。

- (1) 図 3 は、幾つかのリレーションシップが欠落している。欠落しているリレーションシップを補って図を完成させよ。
- (2) 図 4 は、幾つかのリレーションシップが欠落している。欠落しているリレーションシップを補って図を完成させよ。
- (3) 図 5 中の ア ～ ケ に入れる一つ又は複数の適切な属性名を補って関係スキーマを完成させよ。
- (4) 図 6 中の コ ～ ナ に入れる一つ又は複数の適切な属性名を補って関係スキーマを完成させよ。

設問 2 「再構築における店舗への在庫及び出荷のやり方の変更」に基づいて変更した概念データモデル及び関係スキーマについて、次の問いに答えよ。

- (1) 図 8 には、四つのエンティティタイプが欠落している。欠落したエンティティタイプ及び関連するリレーションシップを補って図を完成させよ。

なお、欠落しているエンティティタイプを補うときのエンティティタイプ名は、本文中の字句を参考に、現状と同じエンティティタイプ名の使用も含めて意味的にふさわしく命名せよ。

- (2) 図 9 において、(1)で補った四つのエンティティタイプに対応する関係を追加する必要がある。追加する関係について、関係スキーマを答えよ。

なお、(1)で補った四つのエンティティタイプに対応する関係の関係名は、(1)で補ったエンティティタイプ名を用いること。

- (3) 図 9 中の ニ，ヌ に入れる一つ又は複数の適切な属性名を補って関係スキーマを完成させよ。

6. 退室可能時間中に退室する場合は、手を挙げて監督員に合図し、答案用紙が回収されてから静かに退室してください。

退室可能時間	15:10 ～ 16:20
--------	---------------

7. 問題に関する質問にはお答えできません。文意どおり解釈してください。
8. 問題冊子の余白などは、適宜利用して構いません。ただし、問題冊子を切り離して利用することはできません。
9. 試験時間中、机上に置けるものは、次のものに限りです。
- なお、会場での貸出しは行っていません。
- 受験票、黒鉛筆及びシャープペンシル（B 又は HB）、鉛筆削り、消しゴム、定規、時計（時計型ウェアラブル端末は除く。アラームなど時計以外の機能は使用不可）、ハンカチ、ポケットティッシュ、目薬
- これら以外は机上に置けません。使用もできません。
10. 試験終了後、この問題冊子は持ち帰ることができます。
11. 答案用紙は、いかなる場合でも提出してください。回収時に提出しない場合は、採点されません。
12. 試験時間中にトイレへ行きたくなったり、気分が悪くなったりした場合は、手を挙げて監督員に合図してください。

試験問題に記載されている会社名又は製品名は、それぞれ各社又は各組織の商標又は登録商標です。

なお、試験問題では、TM 及び [®] を明記していません。