

# Oracle Direct Seminar



# ORACLE®

## いまさら聞けない！？ Oracle Database設計

日本オラクル株式会社

**Oracle** Direct



# アジェンダ



- データベース設計の流れ
- 論理設計
  - データ・モデリング
    - E-R図
    - 正規化
- 物理設計
  - オブジェクトの設計
  - 領域の設計
  - アクセス・コントロールの設計

## 無償技術サービスOracle Direct Concierge

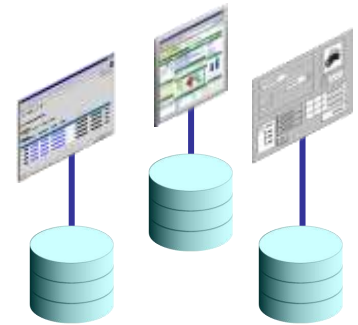
- SQL Serverからの移行アセスメント
  - MySQLからの移行相談
  - PostgreSQLからの移行相談
  - Accessからの移行アセスメント
- Oracle Database バージョンアップ支援
- Oracle Developer/2000 Webアップグレード相談
  - パフォーマンス・クリニック
  - Oracle Database 構成相談
  - Oracle Database 高可用性診断
    - システム連携アセスメント
    - システムセキュリティ診断
      - 簡易業務診断
    - メインフレーム資産活用

<http://www.oracle.com/lang/jp/direct/services.html>

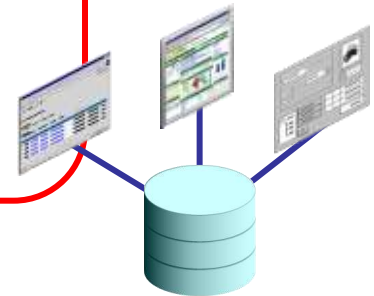
ORACLE

# 主なシステム開発方法

- プロセス中心アプローチ (Process Oriented Approach)
  - 業務処理の流れを元に、システムを設計していく方法
    - データが冗長化しやすく、データの整合性を維持する必要
    - アプリケーションを構築するたびにデータを用意する必要

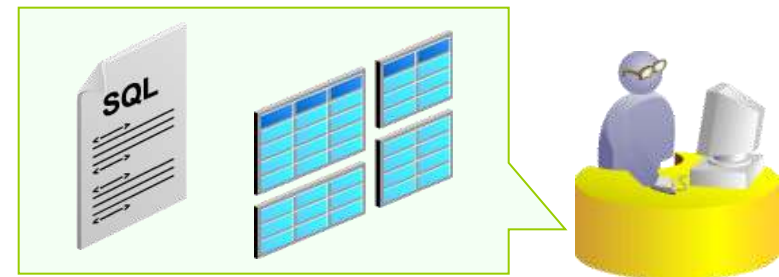
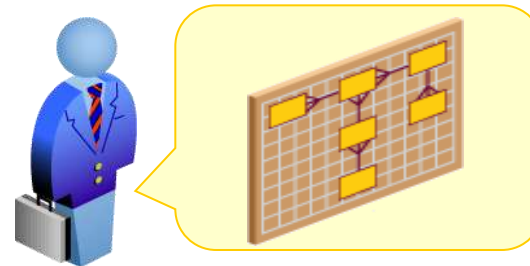


- データ中心アプローチ (Data Oriented Approach )
  - 業務上必要なデータを元にシステムを設計していく方法
    - データが各アプリケーションから独立して一元管理されているので整合性が保たれる
    - 処理が変わってもデータは変わらないため、メンテナンス性が向上



# データベース設計の流れ

- ビジネス要件の調査
  - 機能要件、非機能要件の調査
  - システム化対象範囲のデータの確認
- 論理設計
  - ビジネス活動をデータモデルで可視化
  - リレーショナル表の設計
- 物理設計
  - 使用環境を考慮した設計
  - 物理配置の決定
- データベースの構築



# ビジネス要件の調査

## 要件定義の確認

- 機能要件、非機能要件の調査
  - 機能要件
    - 業務に必要なシステムの動作や処理内容
  - 非機能要件
    - 処理性能
    - 信頼性
    - 拡張性
    - セキュリティ
- システム化対象範囲のデータの確認
  - 業務フローの作成
  - モデリング

業務処理をするために最低限必要な機能は？  
パフォーマンスやセキュリティを向上させ  
実業務に耐えうるシステムにするには？

業務をシステム化するには  
どのデータが必要？  
どのような順で処理をする？



# アジェンダ



- データベース設計の流れ
- 論理設計
  - データ・モデリング
    - E-R図
    - 正規化
- 物理設計
  - オブジェクトの設計
  - 領域の設計
  - アクセス・コントロールの設計

# データベースの論理設計

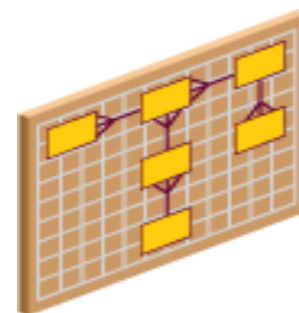
## 論理設計とは

- 論理設計とは
  - システム化の範囲となるデータを整理し、誰にでも分かる形式で明確化すること
  - 業務要件を論理モデルに変換し、最適なテーブル構成を設計すること
- 代表的なデータベースのモデリング手法
  - E-Rモデル
  - 正規化
  - テーブル・インスタンス・チャート
  - CRUD図

# E-Rモデル

## E-Rモデルとは

- ビジネス・データを表現し、相互の規則を明確にするモデリング手法
  - E: 企業が管理すべき対象
  - R: ビジネスルール
- E-R モデルの4つの要素
  - エンティティ
  - リレーションシップ
  - 属性(アトリビュート)
  - 一意識別子(Unique Identifier)





# E-Rモデル

## エンティティ

- 人や物、具体的または抽象的なものを含めて、情報を蓄積する対象となるもの
- エンティティの記述方法
  - ソフトボックス(角の丸い四角形)で表現
  - 一意なエンティティ名を記述
  - 必要に応じてシノニム名を記述



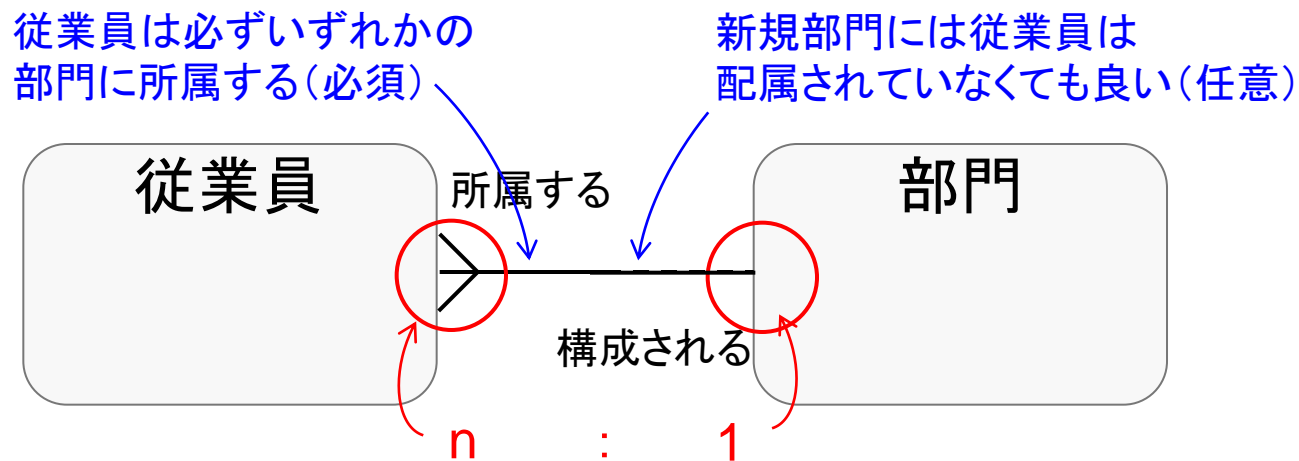
従業員

部門

# E-Rモデル

## リレーションシップ

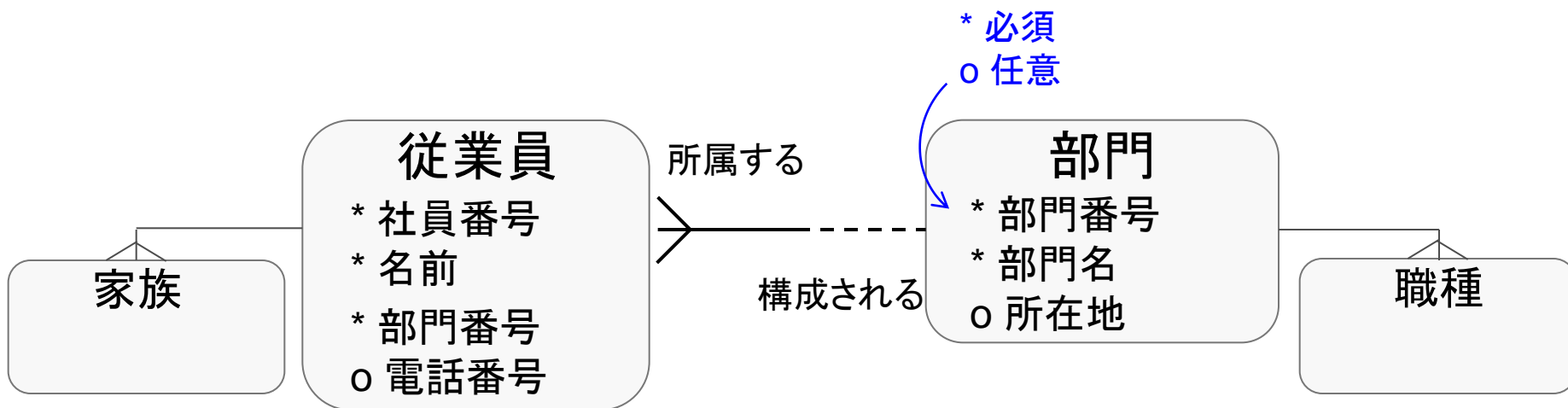
- エンティティ間の関連を表現するもの
- リレーションシップの記述方法
  - 関連があるエンティティを線で結ぶ
  - 相互関連が分かるように、関係性を記述
  - デグリー(多または1)、任意性(任意または必須)を記述



# E-Rモデル

## 属性

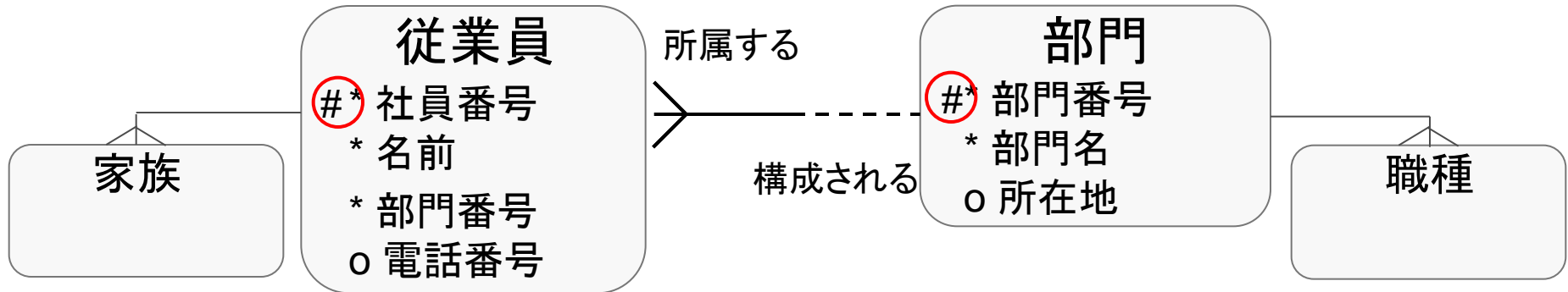
- 属性とは、エンティティを構成する情報
- 属性の記述方法
  - エンティティの中に記述
  - それぞれの属性の任意性を記述
- 属性に複数の値が入る場合は別のエンティティに分ける
  - 例) 社員の家族名、部門内の職種



# E-Rモデル

## 一意識別子

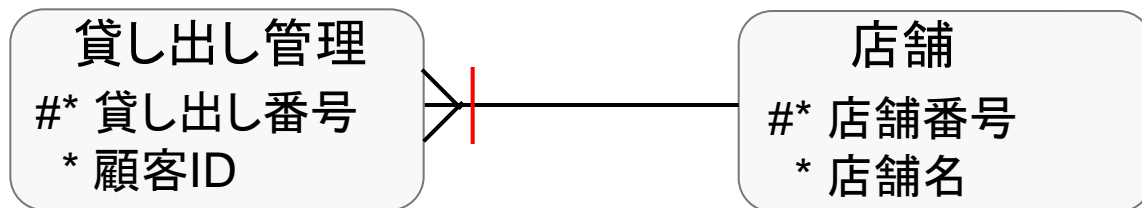
- エンティティを一意に識別するもの
- 属性の記述方法
  - 一意識別子となる属性には「#」印を記述



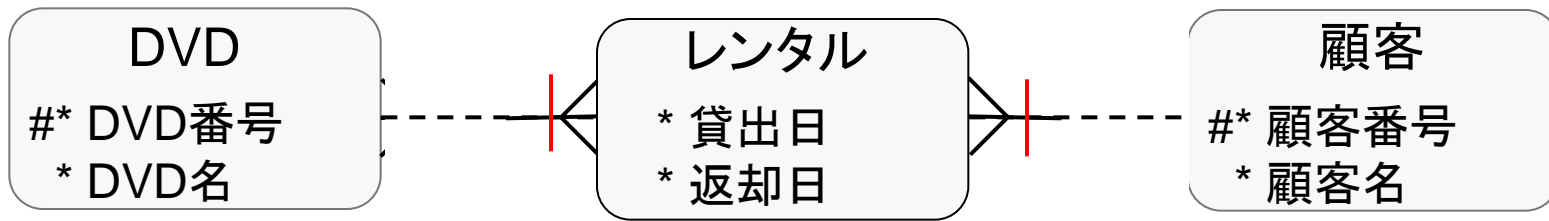
# [参考]E-Rモデル

## 応用例

- リレーションを含む一意識別子
  - 関連するリレーションシップに縦棒「|」を引く  
例) レンタルビデオ店で、店舗ごとに貸し出し番号を出す  
(店舗番号と貸し出し番号で一意)



- 多:多のリレーションシップの分解
  - 交差エンティティを追加して分解  
例) レンタルビデオ店で、DVDを顧客に貸し出す



# 表の正規化

## 正規化とは

- たくさん存在するデータ項目を構造的にグループ化していくこと
- 正規化することによるメリット
  - データを追加・削除・変更をする時に、整合性を保つことが容易になる
  - データの重複をなくすことで、業務の変化やシステムの拡張に柔軟に対応することができるようになる
- 正規化の手順
  - 非正規化
  - 第1正規化
  - 第2正規化
  - 第3正規化
  - 正規化の見直し

# 正規化

## 非正規化

- 受注伝票のデータから表を設計する

### 受注伝票

-----  
受注番号 : 12345678  
受注日 : 2009年06月01日  
顧客番号 : 1230  
顧客氏名 : 田中 太郎  
住所 : 〒107-0001 東京都港区北青山X-X-X  
支払方法 : 代金引換  
担当者コード : 250  
担当者氏名 : 鈴木 花子

### お買い上げ明細

商品番号	商品名	単価	個数	価格
A001	Oracle Database入門	1200	1	1200
M021	Oracle Master Gold参考書	2800	2	5600
X005	データベース設計	1700	1	1700

-----  
計 8500

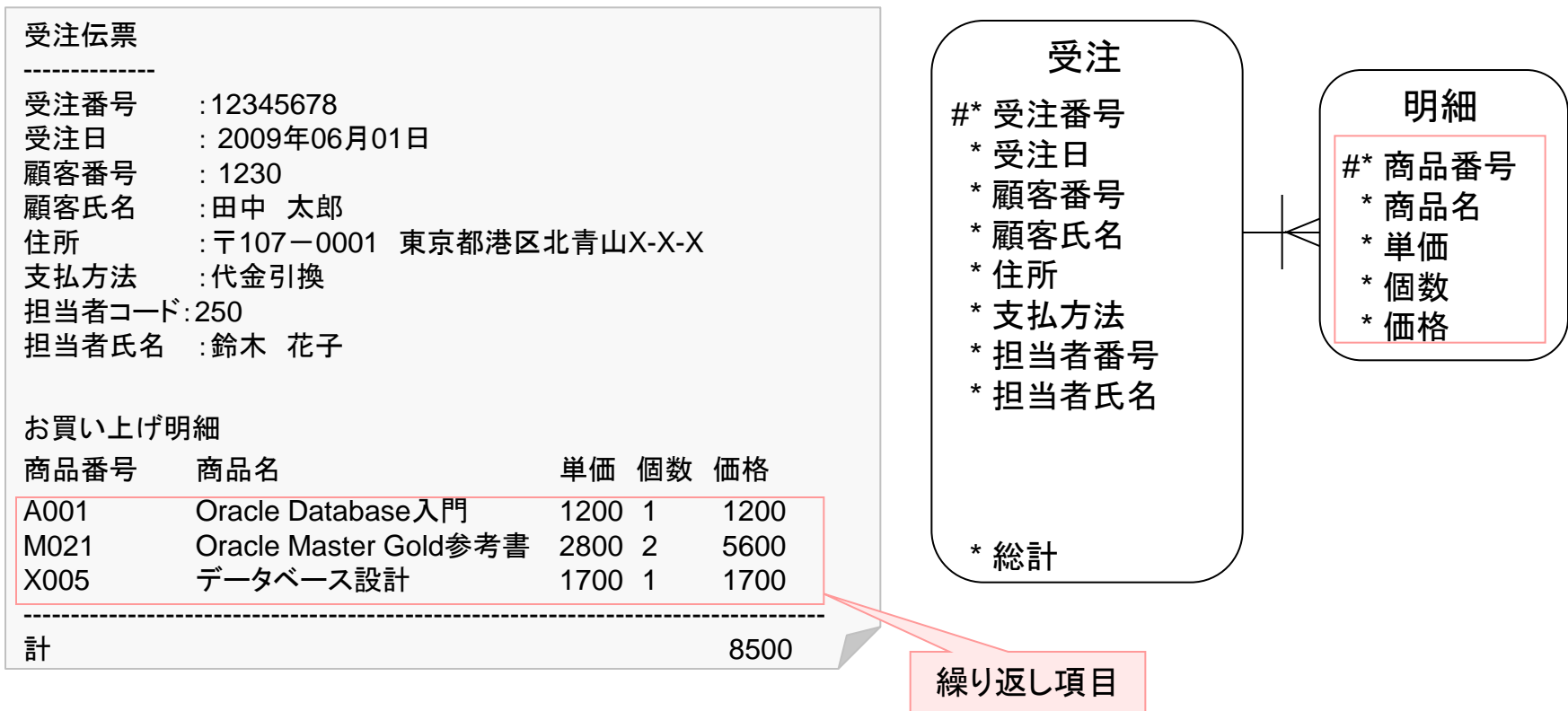
### 受注

#\* 受注番号  
\* 受注日  
\* 顧客番号  
\* 顧客氏名  
\* 住所  
\* 支払方法  
\* 担当者番号  
\* 担当者氏名  
\* お買い上げ明細  
\* お買い上げ明細  
\* お買い上げ明細  
\* 総計  
\* 総数

# 正規化

## 第1正規化

- 「繰り返し項目」を排除し、エンティティとして独立させる



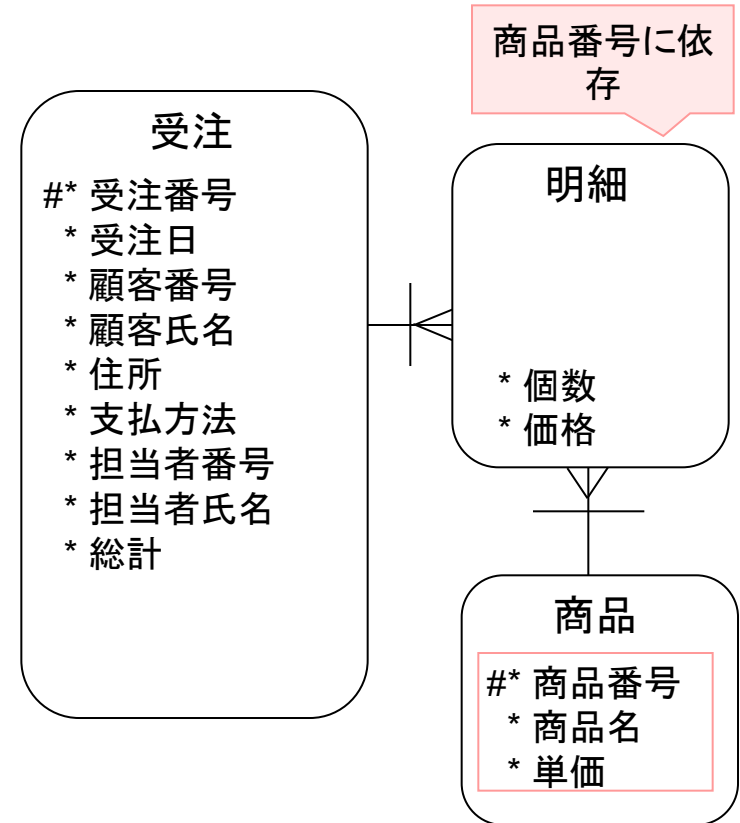


# 正規化

## 第2正規化

- 主キー項目に依存している項目を別のエンティティとして独立させる

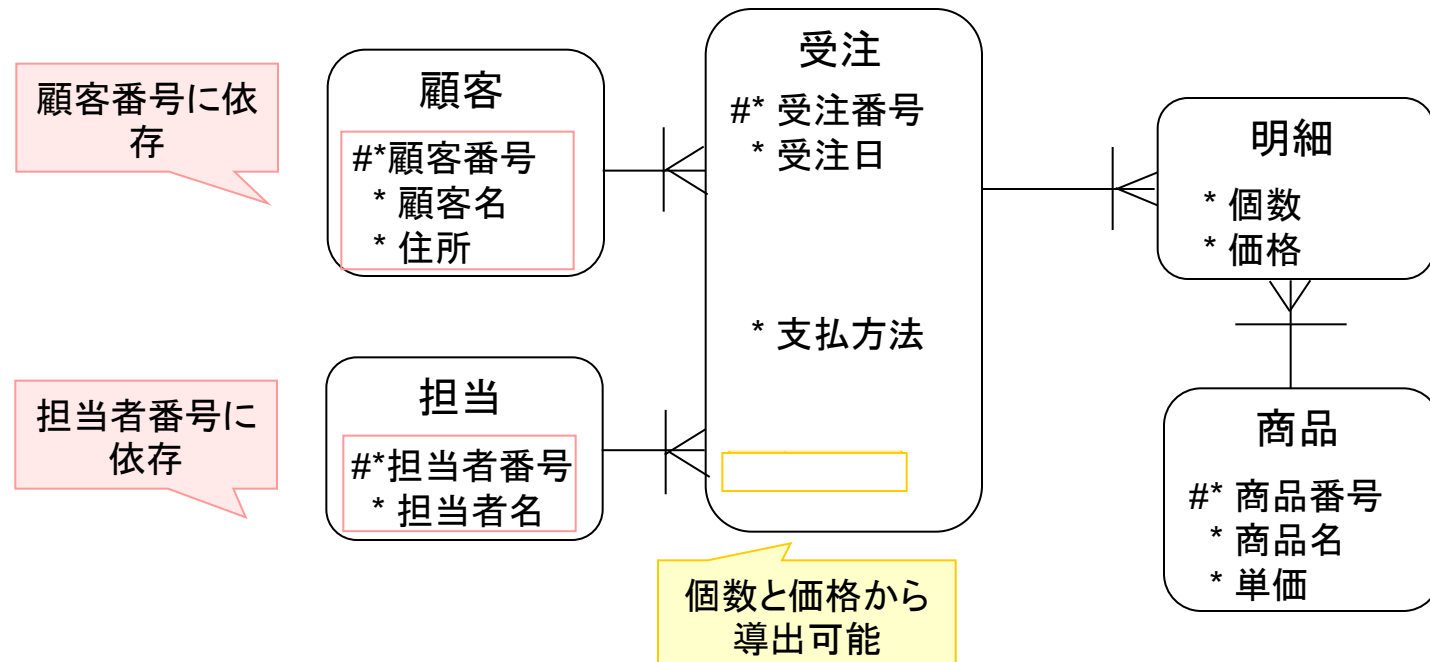
受注伝票				
-----				
受注番号	:	12345678		
受注日	:	2009年06月01日		
顧客番号	:	1230		
顧客氏名	:	田中 太郎		
住所	:	〒107-0001 東京都港区北青山X-X-X		
支払方法	:	代金引換		
担当者コード	:	250		
担当者氏名	:	鈴木 花子		
お買い上げ明細				
商品番号	商品名	単価	個数	価格
A001	Oracle Database入門	1200	1	1200
M021	Oracle Master Gold参考書	2800	2	5600
X005	データベース設計	1700	1	1700
-----				
計				8500



# 正規化

## 第3正規化

- 主キー以外の依存関係のある項目を、新しいエンティティとして独立させる
- 別の項目から導き出せる項目(計算値など)を属性から排除する



# 正規化の見直し

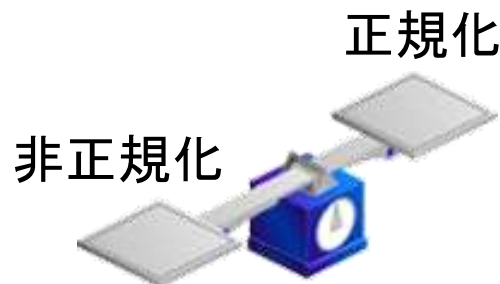
## あえて正規化をしない選択

- 業務処理(アプリケーション)を考慮して、正規化を見直しを検討
  - 検索処理への影響
    - 結合して使うことの多い表を結合した状態で格納(製品コードと製品名)
    - 決まった計算をするデータの計算結果を格納(例:社員の年収)
  - 更新処理への悪影響
    - データの重複によるパフォーマンス、メンテナンス性の低下

一度正規化をしたうえで、処理傾向やパフォーマンスを考慮して検討

検索処理の効率化

- 結合処理
- 計算処理



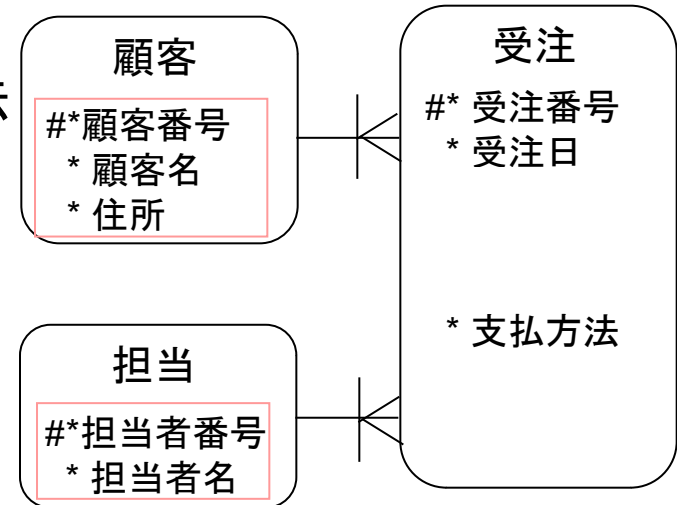
更新処理の効率化

- 検索パフォーマンスの維持
- 保守作業の効率化

# テーブル・インスタンス・チャート

## 表定義への変換

- E-Rモデルから、リレーショナル表への変換手法
  - エンティティを表に変換
  - 属性を列に変換
  - 一意識別子を主キーに変換
  - リレーションシップを外部キーに変換



表名: ORDERS

列名	ORDER_ID	ORDER_DATE	CUST_ID	EMP_ID	PAYMENT
キータイプ	PK				
Not Null/ Unique	NN,U	NN	NN	NN	NN
外部キー 参照表			CUSTOMERS	EMPLOYEES	
データ型	NUMBER	DATE	NUMBER	NUMBER	VARCHAR2
最大長	10		5	5	20
サンプルデータ	12345678	09-06-01	1230	250	代金引換

# CRUD図

## データ処理の流れの整理

- 業務処理と表の関係を整理し、データのライフサイクルを明確化する手法
- 表の整理、問題の洗い出しが可能
  - 読み取り(R)されないのに存在する表はないか
  - 更新(U)、削除(D)されない表のメンテナンスは誰がどのように行うのか
  - どの表にどのような権限や監査が必要か

業務

	顧客	担当者	受注	明細	商品
顧客管理	CRUD	R	R		
商品管理		R			CRUD
受注	R	R	CRUD	CRUD	R

表

C: Create  
R: Read  
U: Update  
D: Delete

# アジェンダ



- データベース設計の流れ
- 論理設計
  - データ・モデリング
    - E-R図
    - 正規化
- 物理設計
  - オブジェクトの設計
  - 領域の設計
  - アクセス・コントロールの設計

# データベースの物理設計

## 物理設計とは

- 物理設計とは
  - 論理設計を実際の環境にあてはめて計画を立てること
  - システムの性能について考慮し、使用するデータベースに依存する機能の使用を検討する
- 物理設計で検討するポイント
  - オブジェクトの設計
  - データ格納領域の設計
  - アクセス・コントロールの設計

# データベースの物理設計

## オブジェクトの設計

- 物理設計のポイント
  - オブジェクトの設計
    - 表(テーブル)の設計
    - その他のオブジェクト
      - ビュー
      - 索引
  - データ格納領域の設計
  - アクセス・コントロールの設計

表作成手順の詳細はこのDirect Seminarで！

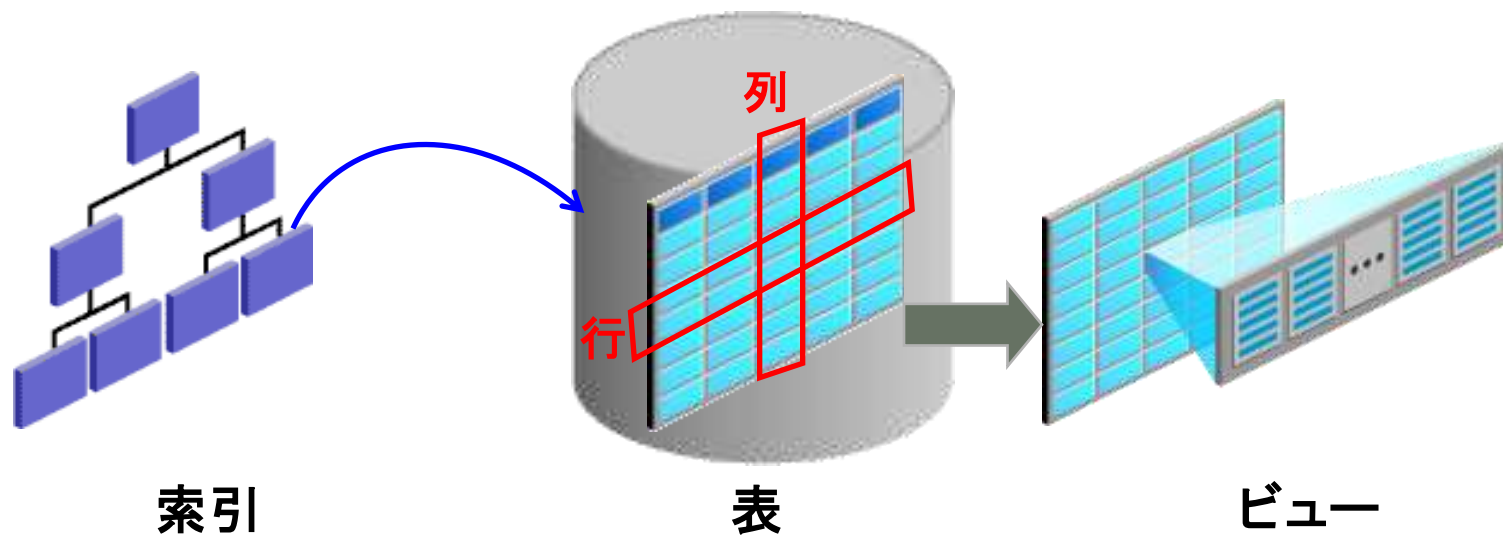
意外と簡単！？ Oracle Database 11g データベース設定編



# オブジェクトの計画

## 必要なオブジェクトの計画

- 論理設計したモデルを元に、表を作成
- 表以外にも、データの管理をするためのオブジェクトを計画
  - 索引(インデックス): 早く検索をするために、表につける「索引」
  - ビュー: 表データの一部を加工して見せる「仮想表」



# 表の設計

## 表定義の決定

- 表設計の際に決定すること
  - 表名
  - 列名(格納項目)
  - データ型、サイズ(どんな種類のデータが、どれくらい入るのか)
  - 制約(格納するデータに対するビジネス・ルール)

### 表名、列名の命名規則

- 30バイト以内
- 英数字、日本語、記号( \_, \$, # )  
(シングルバイト推奨)
- 先頭は数字以外の文字
- 予約語使用不可

### ORDERS

ORDER_ID	ORDER_DATE	CUST_NAME	AMOUNT
1	2008-12-05	たなか	30000
2	2008-12-05	すずき	25000
3	2008-12-09	さとう	20000
4	2008-12-09	よしだ	20000
5	2008-12-10	こばやし	15000

# 表の設計

## データ型の決定

- 列ごとに格納するデータのタイプを決定
- Oracleで使用可能なデータ型で実装

社員表

社員番号	社員名	給与	入社日	部門番号
1	田中	30000	2000/04/01	10
2	鈴木	25000	2002/09/01	20
3	吉田	20000	2001/10/01	20
4	佐藤	20000	2004/04/01	30
5	小林	15000	2006/11/01	30

↑  
数字  
(NUMBER型)

↑  
文字  
(VARCHAR2型)

↑  
日付  
(DATE型)

### Oracleで使用可能なデータ型

- 数値型 NUMBER
  - 文字型 VARCHAR2(可変長)  
CHAR(固定長)
  - 日付型 DATE  
TIMESTAMP(時差含む)
  - LOB型 CLOB  
BLOB など
- 詳細はマニュアル「データベース概要」

# 表の設計

## 制約の設計

- 表の列に対して定義するビジネス・ルール
- 表に対して定義することで、アプリケーション依存せずルールを統一することが可能

### Oracleで使用可能な制約

PRIMARY KEY	行を一意に特定する主キー(一意+必須)
NOT NULL	その列には必ずデータが入ることを保証
FOREIGN KEY	他の表との参照関係を定義する外部キー 関連表の間でのデータの整合性を保証
UNIQUE	その列のデータが重複しないことを保証
CHECK	その他条件(給与が0以上である など)

社員表

社員番号	社員名	給与	入社日	部門番号
1	田中	30000	2000/04/01	10
2	鈴木	25000	2002/09/01	20
3	吉田	20000	2001/10/01	20
4	佐藤	20000	2004/04/01	30
5	小林	15000	2006/11/01	30

↑ PK    ↑ NN    ↑ CK

部門表

部門番号	部門名
10	営業
20	技術
30	サポート

↑ FK    reference

### ビジネス・ルール例

- ✓ 社員番号で社員を見分ける (Primary Key)
- ✓ 社員名は必須 (Not Null)
- ✓ 部門番号は部門表を参照 (Foreign Key)
- ✓ 給与は0より大きい (Check)

ORACLE

# 【参考】その他のオブジェクトの計画 ビューの設計

- 表のデータを加工して提供するオブジェクト
- ビューの目的
  - 実表への直接アクセスを防ぐ(セキュリティ)
  - 実表のデータを様々に加工して提供できる(利便性)

部門表

部門番号	部門名
10	営業
20	技術
30	サポート

社員表

社員番号	社員名	給与	部門番号
1	田中	400000	10
2	鈴木	320000	20
3	吉田	240000	20
4	佐藤	300000	30
5	小林	250000	30

社員名と部門名を  
結合した結果  
配属ビュー

部門名	社員名

部門ごとに平均給与を  
算出した結果  
平均給与ビュー

部門	平均給与

# 【参考】その他のオブジェクトの計画

## 索引の設計

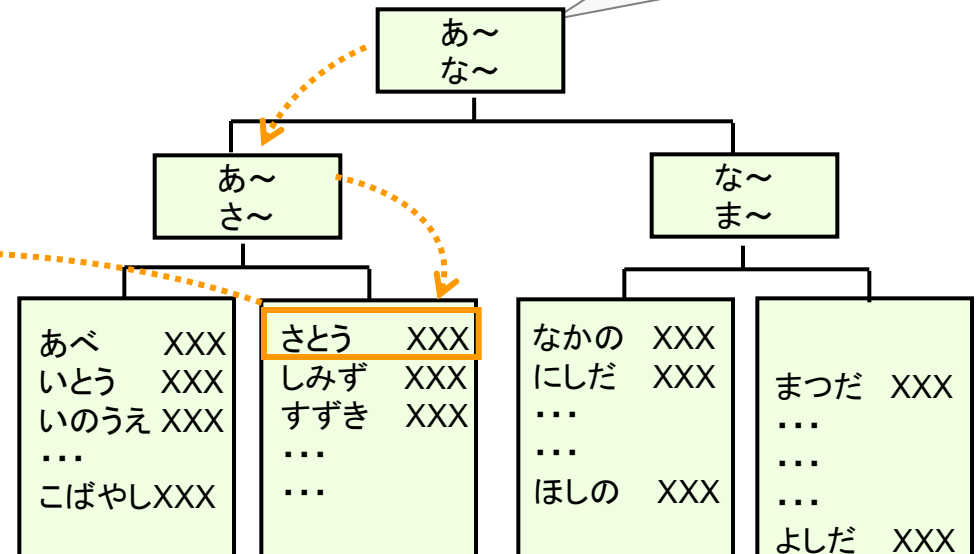
索引の詳細はこのDirect Seminarで！  
実践！パフォーマンスチューニング 索引編

- 特定行へのアクセスを高速にするオブジェクト
- 索引の作成指針
  - 検索列に対して作成
  - 更新のパフォーマンスは低下する傾向にあるため、検索処理と更新処理のバランスによって設計

```
SELECT *  
FROM 社員表  
WHERE 社員名='さとう'
```

社員表

社員番号	社員名	給与	性別
1	たなか	400000	男
2	すずき	320000	男
3	よしだ	240000	女
4	さとう	300000	男
5	こばやし	250000	女



# データベースの物理設計

## データ格納領域の設計

- 物理設計のポイント
  - オブジェクトの設計
  - データ格納領域の設計
    - 表領域の設計
    - 領域割り当ての設計
    - ブロックの設計
  - アクセス・コントロールの設計

領域作成手順の詳細はこのDirect Seminarで！  
意外と簡単！？ Oracle Database 11g データベース設定編

# Oracleの領域管理の概念

## 表領域とは

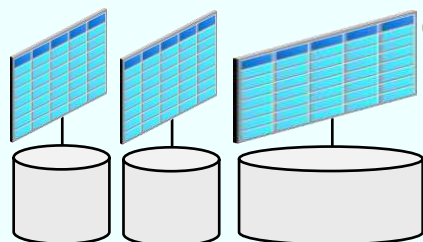
- データ格納先のファイルを「表領域」という単位で管理
  - 表領域は表の論理的な格納先
  - データは物理的にはデータ・ファイルに格納される
  - 表領域は1つ以上のデータ・ファイルから構成される

### 補足: 表領域の特徴

#### <ファイル単位での管理>

**ファイルの物理的な場所やサイズを意識する必要**

- 表作成時、ファイルの物理的な場所を指定
- データが単一ファイルに収まりきらない可能性
- 表ごとにファイルができるとメンテナンス性低下



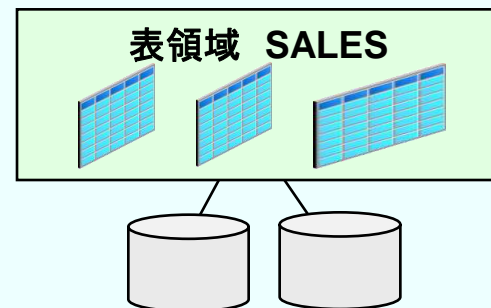
D:\oracle¥...??  
どのデータからどのデータまでが  
どのファイル??



#### <表領域の管理>

**ファイルの場所やサイズを意識せず、柔軟に管理**

- 表作成時、表領域名を指定
- 表領域に複数ファイルに対応付けて拡張可能
- 関連する表をひとつの表領域で管理可能

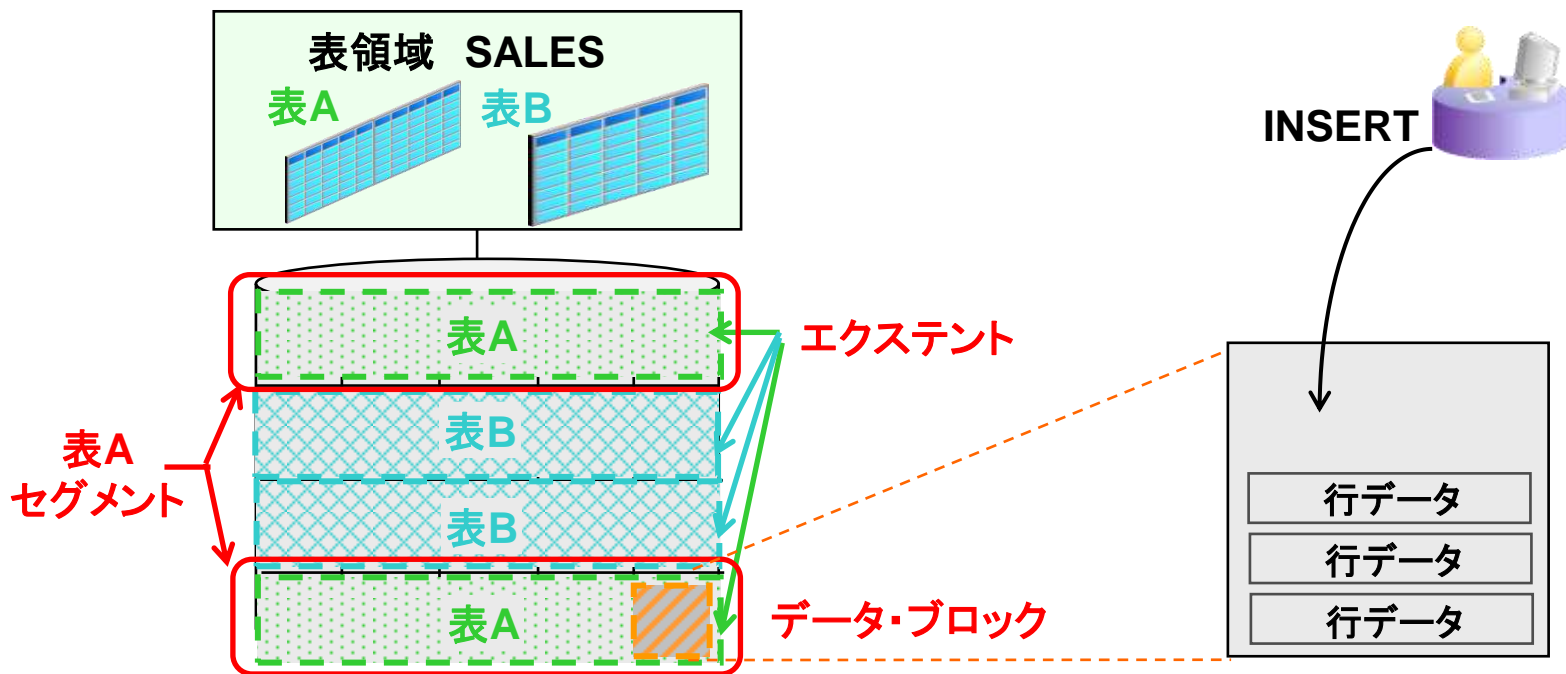




# 【参考】Oracleの領域管理の概念

## 領域の管理単位

- 表領域の論理単位
  - セグメント: ひとつのオブジェクトに割り当てられた領域
  - エクステント: 領域を追加する単位
  - データ・ブロック: Oracleが扱う最小単位(データ格納、I/Oの単位)



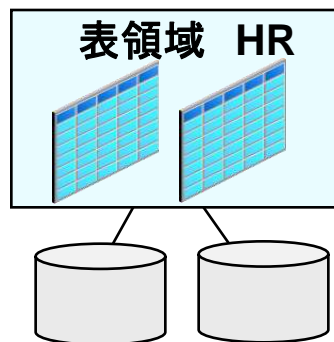
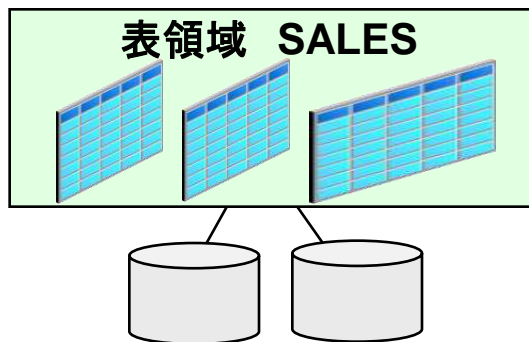
ORACLE

# 表領域の設計

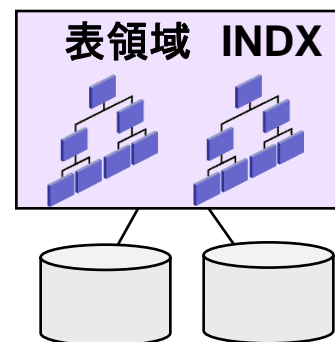
## 表領域の設計方針

- 表領域の設計方針の例
  - アプリケーション単位
  - 表と索引は別の領域に配置
  - メンテナンスの単位
    - バックアップの取得単位
    - オンライン・オフラインの切り替え単位

例) アプリケーション毎に表領域を分けて、領域管理やバックアップを行う



例) 索引は再作成可能なため、定期バックアップから外す

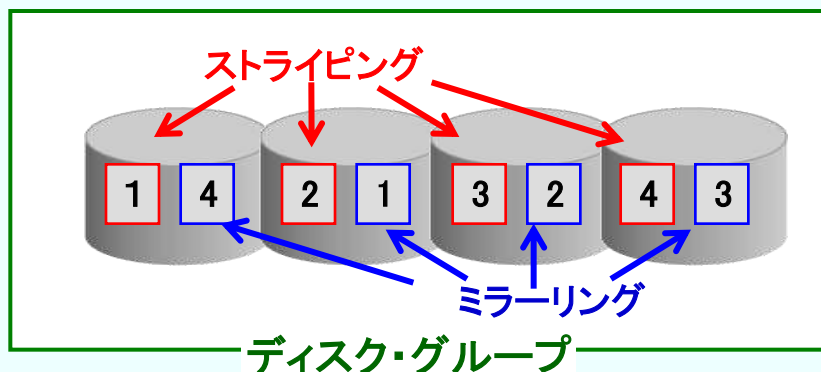
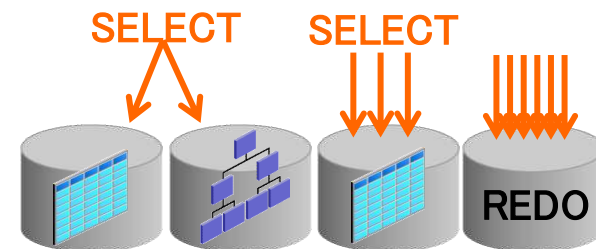


# 表領域の設計

## データ・ファイルの設計方針

- データ・ファイルの設計方針の例
  - データの分散
    - I/Oの分散を意識してファイルを配置
      - 大規模テーブルを複数ファイルに配置
      - REDOログ・ファイルは別ディスクに配置
  - データのストライピング
    - RAID構成のディスク装置に表領域を配置
    - ASMの利用

アクセスタイプによってデータ配置を実施することは実質困難



### 補足: ASM

- ディスクの構成を仮想化
  - ディスクの物理的な配置やサイズを意識しない
  - オンラインで、ディスクの追加や削除が可能
- ストライピング/ミラーリングが可能
  - ストライピング→I/Oが分散(パフォーマンス)
  - ミラーリング→1つのディスクの障害時にも処理の継続が可能(障害対策)

# 領域の設計

## データ・ブロックのサイズ設計

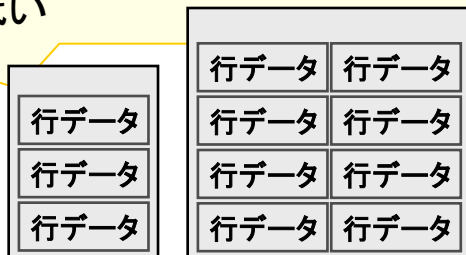
- データ・ブロックのサイズ
  - 標準サイズ: データベース作成時に指定（デフォルト8K）
    - 表領域作成時に指定しなかった場合のデフォルト・サイズ
  - 非標準サイズ: 表領域ごとに異なるブロック・サイズを指定可能
    - 表領域ごとにサイズを指定（2K, 4K, 8K, 16K, 32Kなど）

### 小さいブロックが向いている場合

- 同一ブロックに対する競合を避けたい
- 選択性が高い

### 大きいブロックが向いている場合

- 1行のデータが長い
- 選択性が低い

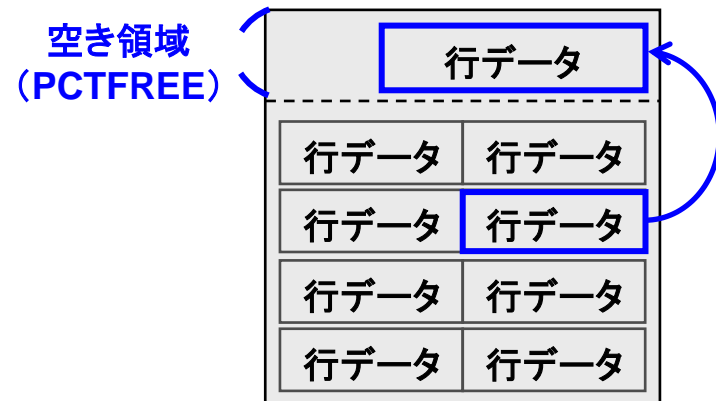


ブロックサイズ	小	大
1ブロックの格納行数	少ない	多い
ブロックI/Oのコスト	低い	高い
ブロックI/Oの回数 (選択性による)	多くなりやすい	少なくなりやすい
トランザクション競合	発生しにくい	発生しやすい
キャッシュ・ヒット	高くなりやすい	低くなりやすい

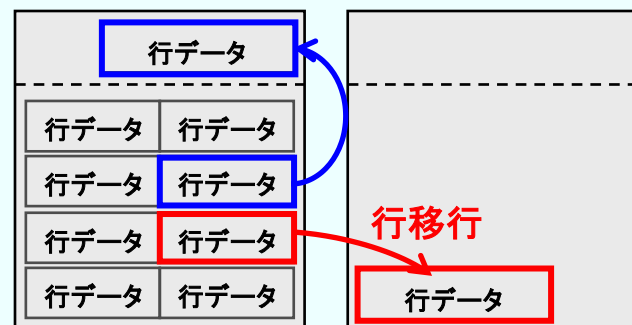
# 【参考】領域の設計

## データ・ブロック内の領域設計

- データ・ブロックの空き領域を「PCTFREE」で設定
  - 行データを格納後、更新処理でデータ量が増える場合の予備領域
  - デフォルトで10%がPCTFREEとして確保
  - 表ごとにPCTFREEのサイズを変更することも可能
- PCTFREEの設定指針
  - 更新の多い表ではPCTFREEを大きめに設定
  - 更新の少ない表ではPCTFREEを小さめに設定



**補足: 行移行**  
更新時のデータ量の増加により、一つのブロックに収まりきらず、別のブロックに行データが移動すること。  
該当行の検索時、2ブロック分読み込むことになるので効率が悪い。



# データベースの物理設計

## アクセス制御の設計

- 物理設計のポイント
  - オブジェクトの設計
  - データ格納領域の設計
  - アクセス制御の設計
    - スキーマの設計
    - 権限の設計

セキュリティの詳細はこのDirect Seminarで！

意外と簡単！？ Oracle Database 11g セキュリティ編  
きちんと知っておこう！基本機能で実現できる

Oracle Databaseのセキュリティ対策

# アクセス制御の設計

## アクセス制御とは

- 誰がどのデータにアクセスできるかを定義すること
- アクセス必要なユーザーが必要なデータにアクセスでき、それ以外のデータにはアクセスできないように制御すること
- アクセス・コントロールの設計
  - スキーマの設計
  - 権限の設計

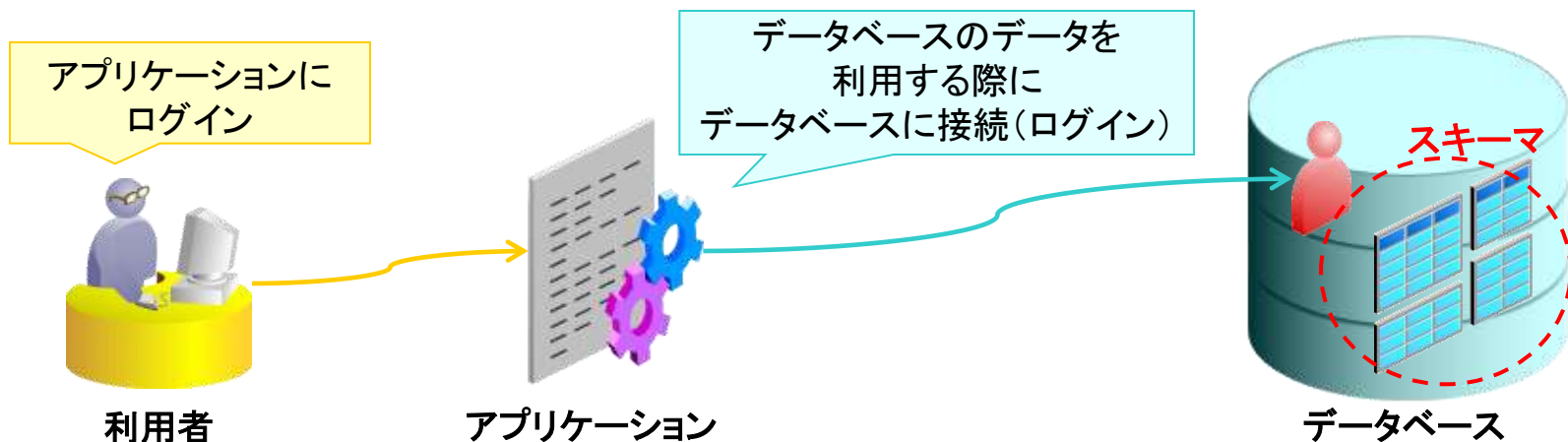


# Oracle Databaseのユーザー・アカウント

## データベース・ユーザーとは

認証やプロファイルの詳細はこのDirect Seminarで！  
意外と簡単！？ Oracle Database 11g セキュリティ編

- データベースに接続するユーザー・アカウント
  - いくつかの認証方式で認証
    - パスワード認証
    - OS認証
  - 「ユーザー・プロファイル」でパスワードやリソース使用の制限が可能
  - 表などのオブジェクトを作成、管理
  - 所有者が同じオブジェクトの集合を「スキーマ」という



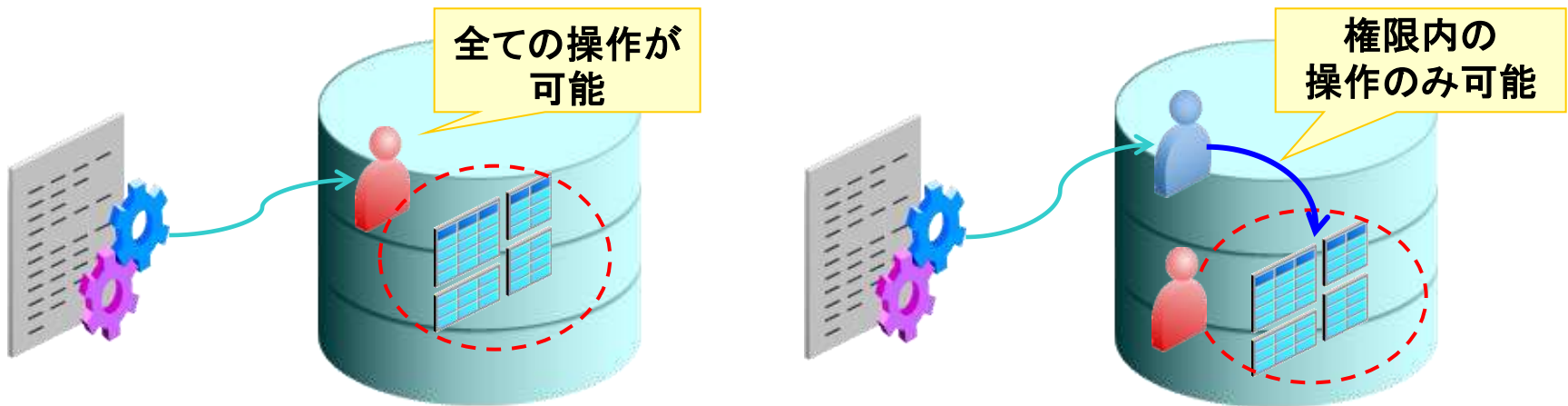
ORACLE



# スキーマの設計

## スキーマ設計の指針

- アプリケーションごとに、アクセスするためのユーザーを作成
  - 複数アプリケーションから同一アカウントでアクセスすると、問題発生時の切り分けが難しくなるため
- 表を所有するユーザーとアプリケーションからアクセスするユーザーを分離
  - アプリケーションで想定していない操作が行われた場合も、権限内の操作しかできないため



# オブジェクト・アクセスの設計

## 権限とは

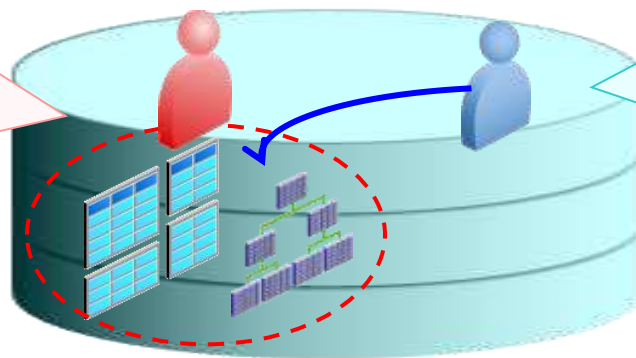
権限の詳細はこのDirect Seminarで！

意外と簡単！？ Oracle Database 11g セキュリティ編

- 実行可能なSQL文およびデータへのアクセスを制御するもの
- 権限の種類
  - データベースで特定の操作をさせるための許可：システム権限
    - 管理者が各データベース・ユーザーに対して付与
    - オブジェクトの作成、データベースの管理などに関する操作許可
  - 他のスキーマにアクセスするための許可：オブジェクト権限
    - 管理者、またはオブジェクトの所有者が付与
    - オブジェクト毎に許可する操作を指定

### システム権限

- 表の作成
- ビューの作成
- 管理権限
  - DBAロール
  - ユーザーの作成/削除
  - 全ての表の削除 (ANY権限)



### オブジェクト権限

- オブジェクト毎に指定
- SELECT
- INSERT
- UPDATE
- DELETE

ORACLE

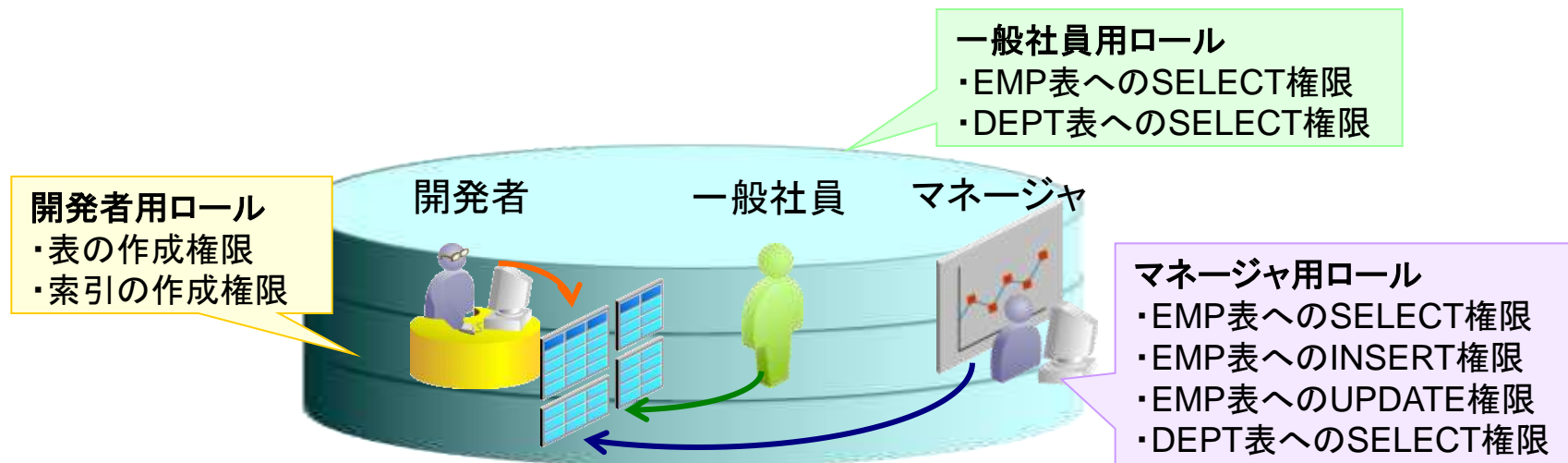
# 【参考】オブジェクト・アクセスの設計

## ロールとは

ロールの詳細はこのDirect Seminarで！

意外と簡単！？ Oracle Database 11g セキュリティ編

- 権限や他のロールの集合
- 代表的なシステム定義ロール
  - CONNECT: 接続権限
  - RESOURCE: 表やビューなどのオブジェクト作成権限セット
  - DBA: 管理者権限セット
- 管理者が必要な権限をまとめてロールを作ることも可能



# オブジェクト・アクセスの設計

## 権限設計のポイント

- 権限の設計
  - 権限最小化の原則
    - ユーザーには必要最低限の権限のみを付与する
      - 誤操作が行われた際の影響を小さくする
      - アカウント情報を悪用された場合の被害を少なくする
    - PUBLIC(全体)への権限付与を避ける
      - 必要な権限を個別に付与
      - ロールの活用
  - 管理権限を付与するユーザーには監査を設定する

# まとめ

- データベース設計の流れ
- 論理設計
  - E-R図によるデータのモデリングと正規化
  - テーブル・インスタンス・チャート、CRUD図による表への反映
- 物理設計
  - 表、ビュー、索引などのオブジェクト設計
  - 表領域、データブロックを考慮した領域設計
  - スキーマ、権限のによるアクセス制御
- データベースの構築、運用

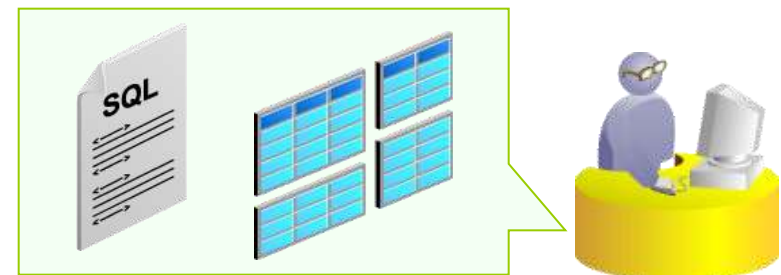
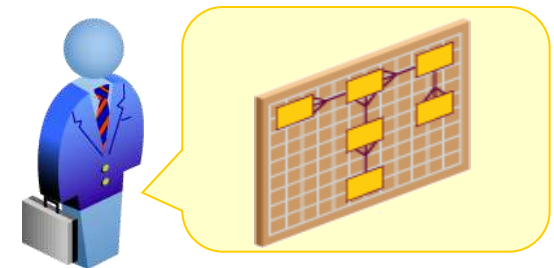
続きはこのDirect Seminarで！

意外と簡単！？ Oracle Database 11g データベース構築編

意外と簡単！？ Oracle Database 11g データベース設定編

今さら聞けない！？ バックアップ・リカバリ入門

今さら聞けない！？ パフォーマンス・チューニング入門



# OTN×ダイセミ でスキルアップ!!



- ・一般的な技術問題解決方法などを知りたい！
- ・ 세미나資料など技術コンテンツがほしい！

Oracle Technology Network(OTN)を御活用下さい。

<http://otn.oracle.co.jp/forum/index.jspa?categoryID=2>

一般的技術問題解決にはOTN揭示版の  
「データベース一般」をご活用ください

※OTN揭示版は、基本的にOracleユーザー有志からの回答となるため100%回答があるとは限りません。  
ただ、過去の履歴を見ると、質問の大多数に関してなんらかの回答が書き込まれております。

<http://www.oracle.com/technology/global/jp/ondemand/otn-seminar/index.html>

過去の 세미나資料、動画コンテンツはOTNの  
「OTNセミナー オンデマンドコンテンツ」へ

※ダイセミ事務局にダイセミ資料を請求頂いても、お受けできない可能性がございますので予めご了承ください。  
ダイセミ資料はOTNコンテンツ オン デマンドか、 세미나実施時間内にダウンロード頂くようお願い致します。

ORACLE

# OTNセミナー オンデマンド コンテンツ

ダイセミで実施された技術コンテンツを動画で配信中!!

ダイセミのライブ感はそのままに、好きな時間で受講頂けます。

最新のコンテンツ

 <p>エンジニアのための ITIL実践術 再生時間: 60分</p>	 <p>ここからはじめよう Oracle PL/SQL入門 再生時間: 60分</p>	 <p>実践!!高可用システ ム構築 -RAC基本 再生時間: 60分</p>	 <p>お悩み解決! Oracle のサイジング 再生時間: 60分</p>
--	--	---	---

Database

 <p>今さら聞けない!!バック アップ・リカバリ入 再生時間: 60分</p>	 <p>意外と簡単!? Oracle Database 11g -セ 再生時間: 60分</p>	 <p>実践!!バックアップ ・リカバリ 再生時間: 60分</p>	 <p>意外と簡単!? Oracle Database 11g -デ 再生時間: 60分</p>
---	---	--	---

>> もっと見る

OTN オンデマンド

検索

※掲載のコンテンツ内容は予告なく変更になる可能性があります。

期間限定での配信コンテンツも含まれております。お早めにダウンロード頂くことをお勧めいたします。

ORACLE



# Oracle エンジニアのための技術情報サイト オラクルエンジニア通信

<http://blogs.oracle.com/oracle4engineer/>

- 技術資料
  - ・ ダイセミの過去資料や製品ホワイトペーパー、スキルアップ資料などを多様な方法で検索できます
  - ・ キーワード検索、レベル別、カテゴリ別、製品・機能別
- コラム
  - ・ オラクル製品に関する技術コラムを毎週お届けします
  - ・ 決してニッチではなく、誰もが明日から使える技術の「あ、そうだったんだ！」をお届けします

オラクルエンジニア通信



先月はこんな資料が人気でした

- ✓ Oracle Database 11gR2 RAC インストレーション・ガイド ASM 版 Microsoft Windows x86-64
- ✓ Oracle Database 11gR2 旧バージョンからのアップグレード

ORACLE



# オラクル クルクルキャンペーン

あの**Oracle Database Enterprise Edition**が超おトク!!

おトクな買い方  
**オラクル5年分**

- ライセンス使用期間 を**5年**間に設定
- 初期のライセンスコストがなんと**67%OFF** !
- テクニカル・サポート価格も**53%OFF** !

Oracle Databaseの  
ライセンス価格を**大幅に抑えて**  
ご導入いただけます

多くのお客様でサーバー使用期間とされる  
5年間にライセンス期間を限定

- 期間途中で永久ライセンスへ差額移行
- 5年後に新規ライセンスを購入し継続利用
- 5年後に新システムへデータを移行



**Enterprise Edition**はここが違う!!

- 圧倒的な**パフォーマンス**!
- データベース**管理がカンタン**!
- データベースを**止めなくていい**!
- もちろん**障害対策**も万全!

この機能でこの価格  
**ライセンスパック**

- Oracle Databaseの機能を**存分に使える**!
- **2ノードRAC**構成も可能!
- サーバー構成によって計4種類のバックから**選べる**!

詳しくはコチラ

<http://www.oracle.co.jp/campaign/kurukuru/index.html>

Oracle Direct 0120-155-096

お問い合わせフォーム

[http://www.oracle.co.jp/inq\\_pl/INQUIRY/quest?rid=28](http://www.oracle.co.jp/inq_pl/INQUIRY/quest?rid=28)

ORACLE

あなたにいちばん近いオラクル



# Oracle Direct

まずはお問合せください

Oracle Direct

検索

システムの検討・構築から運用まで、ITプロジェクト全般の相談窓口としてご支援いたします。

システム構成やライセンス/購入方法などお気軽にお問い合わせ下さい。

## Web問い合わせフォーム

専用お問い合わせフォームにてご相談内容を承ります。

[http://www.oracle.co.jp/inq\\_pl/INQUIRY/quest?rid=28](http://www.oracle.co.jp/inq_pl/INQUIRY/quest?rid=28)

※フォームの入力には、Oracle Direct Seminar申込時と同じ  
ログインが必要となります。

※こちらから詳細確認のお電話を差し上げる場合がありますので、ご登録されている連絡先が最新のものになっているか、ご確認下さい。

## フリーダイヤル

**0120-155-096**

※月曜～金曜 9:00～12:00、13:00～18:00

(祝日および年末年始除く)

ORACLE



以上の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント(確約)するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

Oracle、PeopleSoft、JD Edwards、及びSiebellは、米国オラクル・コーポレーション及びその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標の可能性あります。