

【B3】DatabaseGearテクニカルセッション



## データモデリングで楽々！DB構築

日揮情報システム株式会社  
村上 浩貴

### アジェンダ



- はじめに
- データモデルを使ってDBを作る！
- ER/Studioをフル活用して効率化！
- DB構築はこれだけ簡単になる！
- おわりに

はじめに

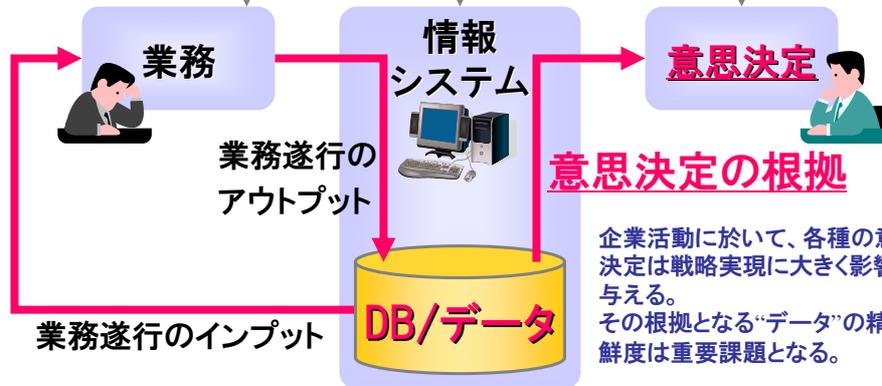
データは企業の重要資産

ビジネス戦略 / IT戦略

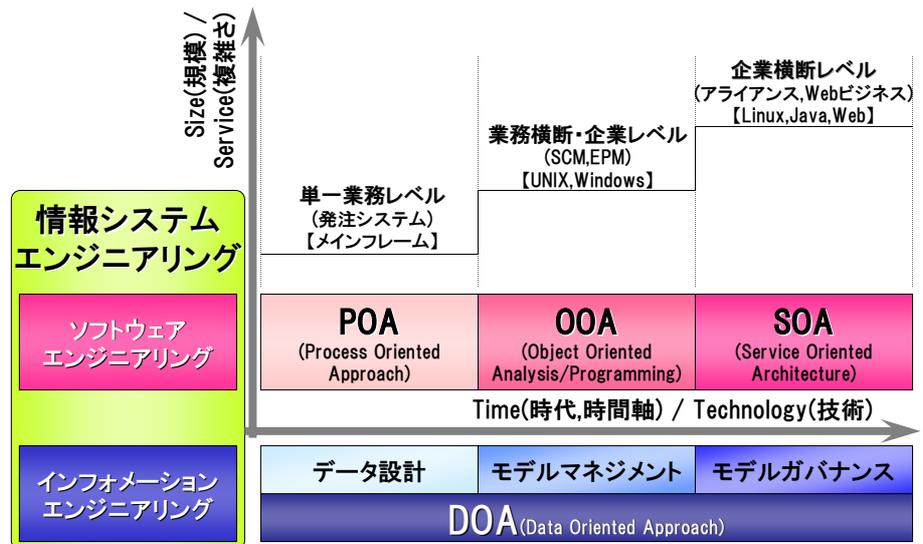
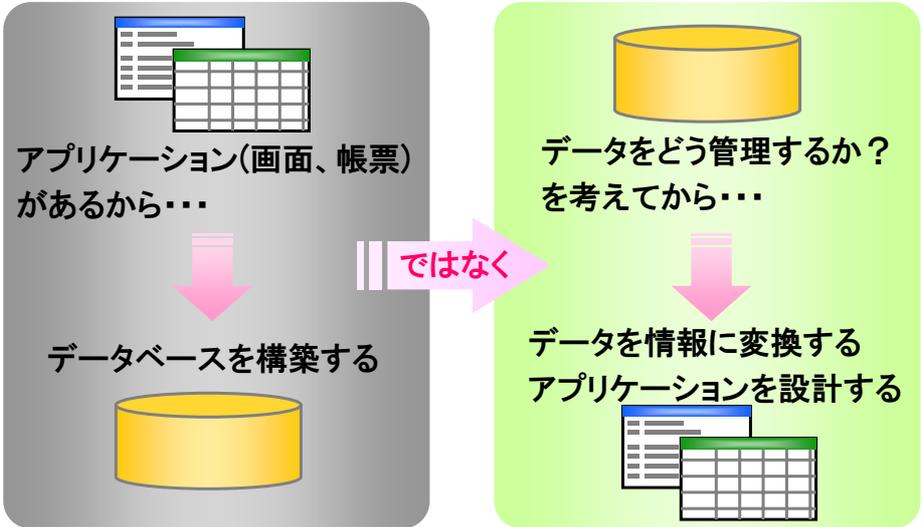
戦略的BPR

戦略的IT投資

戦略的  
意思決定



まず、「重要資産である“データ”をどう管理するか？」



† 如何に重要資産(データ)をマネジメントするか？

† ツールを使って、如何に楽にDBを構築するか？

† 楽をしながら、如何にDBの品質をキープするか？

データモデルを使ってDBを作る！

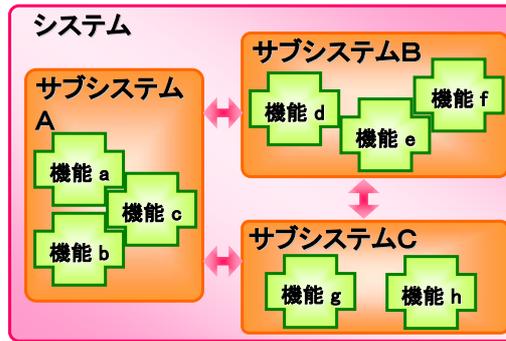
# システムとは？

“システム”とは、「個々要素が有機的に結合された全体」のことを指します。

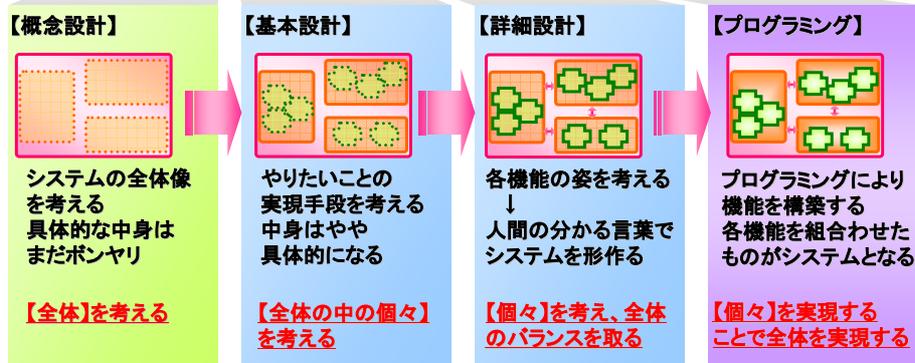
個々要素の結合とは、例えば・・・



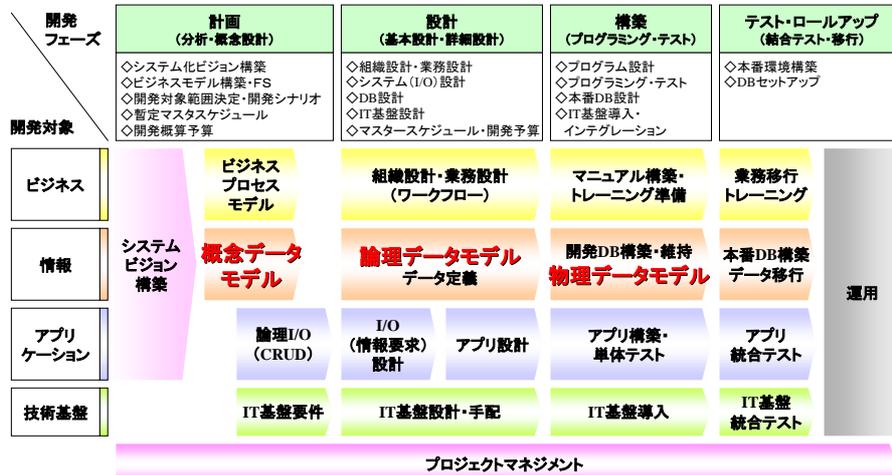
と言った具合です。



# システム開発のステップ

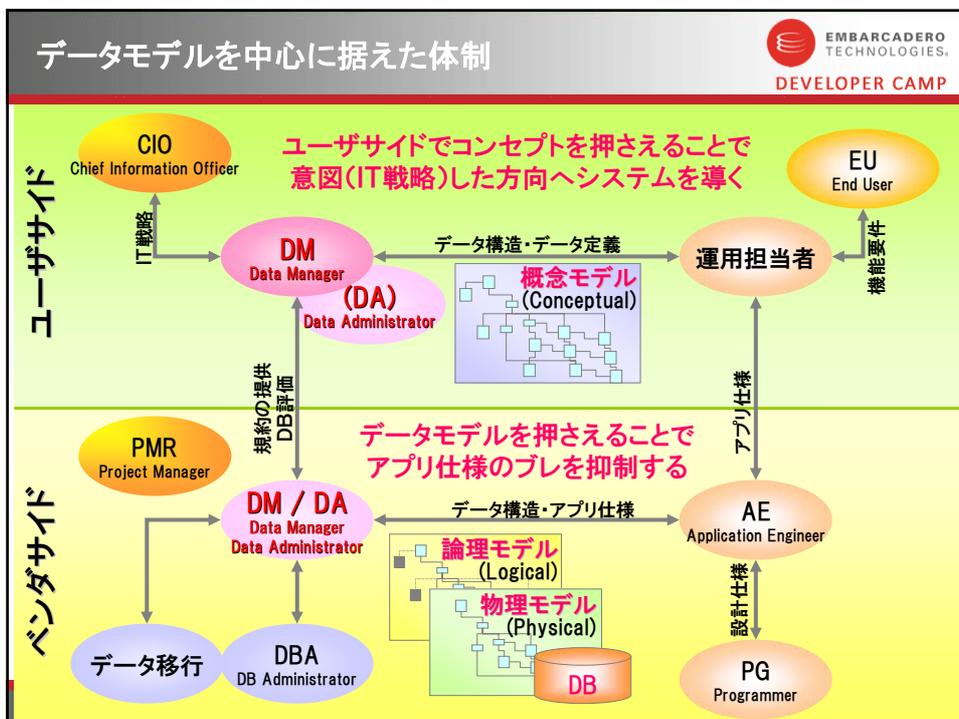
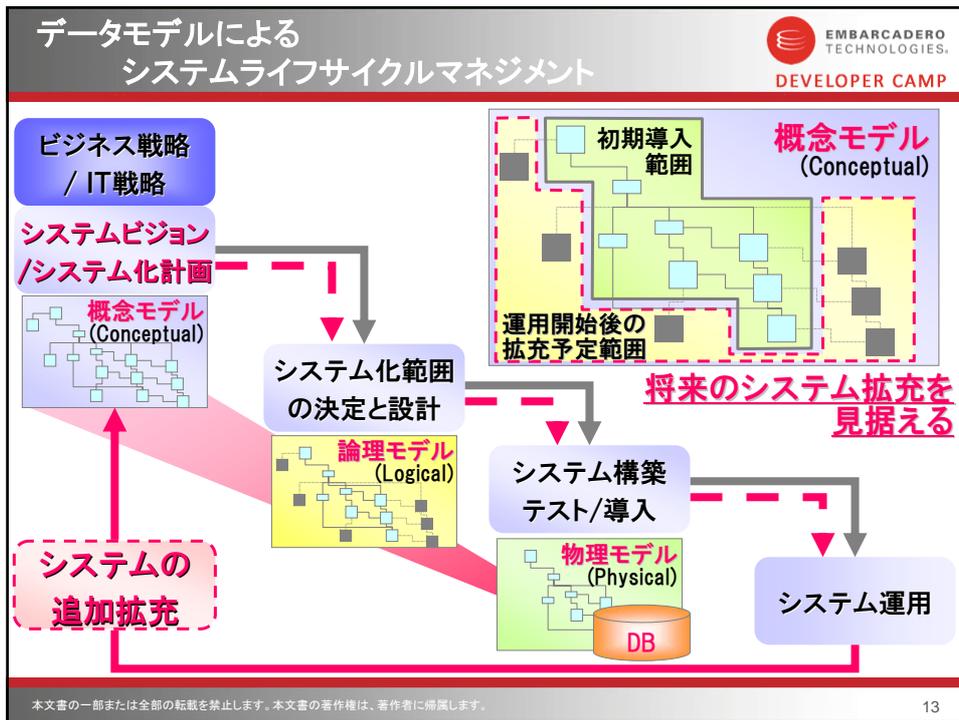


- DOAをベースにしたシステム開発方法論
- モデルを中核としたMDA (Model Driven Approach)



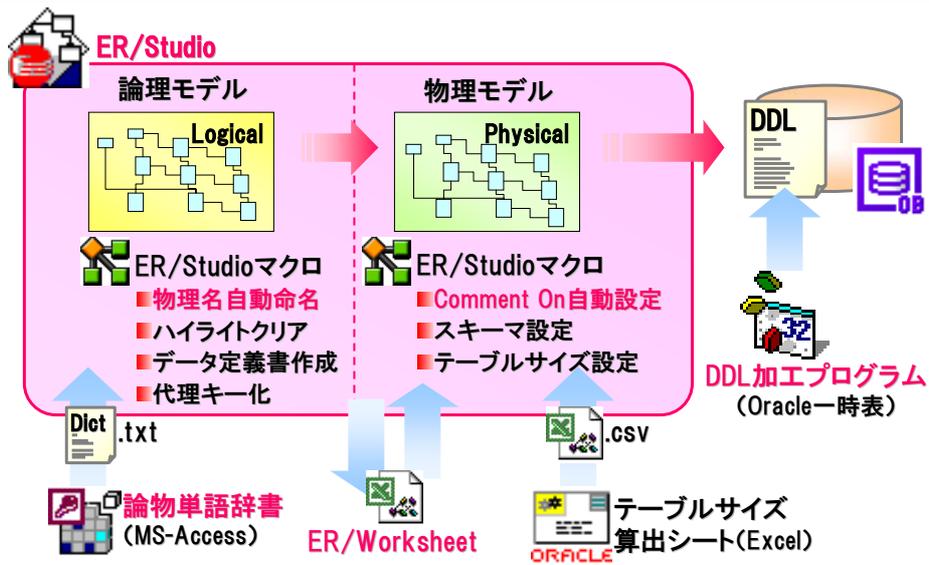
### 3種類のデータモデル





ER/Studioをフル活用して効率化！

データモデリング・オートメーション



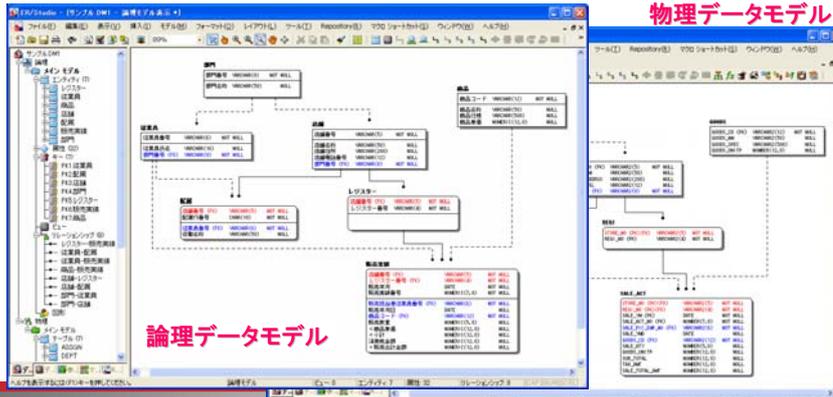


<<主要機能>>

- 論理/物理データモデル管理
- データ型、データ定義、ドメイン
- DDL生成
- モデル間比較マージ など

<<補足>>

- 現在の最新バージョンは7.1 Ver. 7.1ではUndo/Redoが可能に！！
- Sax Basicによるマクロ実行が可能
- ER/Worksheetとの連携が可能



本文書の一部または全部の転載を禁止します。本文書の著作権は、© 2014 Embarcadero Technologies, Inc. によって保護されています。 ※画面スナップはER/Studio 6.5

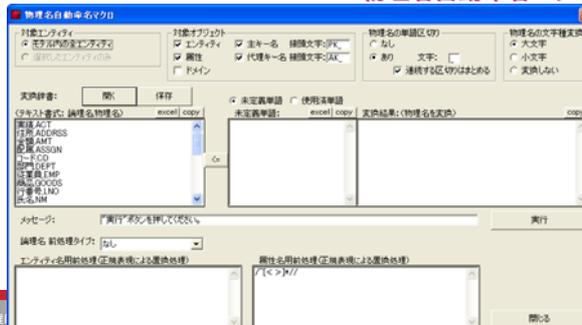


<<主要マクロ>>

- 物理名自動命名
  - データ定義書作成
  - ハイライトクリア
  - Comment On自動設定
  - 代理キー化
- ... 論物変換辞書をベースに物理名を設定 冗長記号(<>)、エンティティIDの自動排除 など
  - ... 納品用データ定義書生成(※固定フォーマット)
  - ... 属性色設定のクリア(e.g. 赤文字⇒黒文字)
  - ... 定義欄(COMMENT ON...)にオブジェクト名をコピー
  - ... 全てのテーブルを単一PK(代理キー)に変換

マクロの活用はモデル管理の効率化に必要不可欠！

物理名自動命名マクロ



本文書の一部または全部の転載を禁止します。本文書の著作権は、© 2014 Embarcadero Technologies, Inc. によって保護されています。

# ER/Studioマクロ 物理名自動命名マクロ



面倒な物理名設定は、ツールによる一括変換で効率化します。

正常置換された変換結果と、未定義単語の確認/出力

正規表現を使った柔軟なテキスト処理

本文書の一部または全部の転載を禁止します。本文書の著作権は、著作者に帰属します。

19

# 論物変換辞書(独自構築)



## <<主要機能>>

- 論理名单語に対する英単語と略語の管理
- 略語辞書出力/正式辞書出力
- 豊富な検索機能

## <<補足>>

- 論理データモデル構築の時点で設定
- フルスペルの英単語管理で理解度向上
- 正式辞書はJavaオブジェクト名として利用可
- ただし、メンテナンスには明確な方針が必要

## 略語辞書

正式辞書

## 論物辞書管理ツール

Ver 1.0.3

本文書の一部または全部の転載を禁止します。本文書の著作権は、著

# DBオブジェクト命名に於ける工夫 論物変換辞書の活用

『ある論理名は必ず決まった物理名になる』ことで、DB可読性を向上します。

## 略語辞書

**【参考】正式辞書**

1	実務, ACT↓	1	実務, Actual↓
2	住所, ADDRESS↓	2	住所, Address↓
3	金額, AMT↓	3	金額, Amount↓
4	配属, ASSGN↓	4	配属, Assien↓
5	コード, CD↓	5	コード, Code↓
6	部門, DEPT↓	6	部門, Department↓
7	従業員, EMP↓	7	従業員, Employee↓
8	商品, GOODS↓	8	商品, Goods↓
9	行番号, LNO↓	9	行番号, Line Number↓
10	氏名, NM↓	10	氏名, Name↓
11	番号, NO↓	11	番号, Number↓
12	担当, RAC↓	12	担当, Reason In Charge↓
13	販売, SALE↓	13	販売, Sale↓
14	役職, POST↓	14	役職, Post↓
15	数量, QTY↓	15	数量, Quantity↓
16	レジスター, REGI↓	16	レジスター, Register↓
17	販売, SPEC↓	17	販売, Spec↓
18	仕様, SPEC↓	18	仕様, Spec↓
19	店舗, STORE↓	19	店舗, Store↓
20	小計, SUB_TOTAL↓	20	小計, Sub Total↓
21	消費税, TAX↓	21	消費税, Tax↓
22	電話番号, TEL↓	22	電話番号, Telephone Number↓
23	合計, TOTAL↓	23	合計, Total↓
24	単価, UNITP↓	24	単価, Unit Price↓
25	年月, YM↓	25	年月, Year Month↓
26	年月日, YMD↓	26	年月日, Year Month Day↓
27		27	

### 論理名から見る

↓

“部門番号”  
(論理名)

↓

“DEPT\_NO”  
(物理名)

### 物理名から見る

↓

“DEPT\_NO”  
(物理名)

↓

“部門番号”  
(論理名)

本文書の一部または全部の転載を禁止します。本文書の著作権は、著作者に帰属します。

# 論理設計時点でのDBオブジェクト名の意識

論理設計時点で、論物辞書を意識し、命名することが肝要です。

**論理設計** → **物理設計** → **実装**

論物辞書

Too Late !!

DB

論物辞書を意識した  
論理名称設定が、  
スタートポイント！  
(意味が分かる  
簡潔な名前)

シームレスに物理  
設計に繋げることで、  
DBA作業への専任化  
/効率化を支援

DBオブジェクト名 **バイト  
数制限を事前に意識**  
することで、手戻り検討  
を抑止

本文書の一部または全部の転載を禁止します。本文書の著作権は、著作者に帰属します。

# DBオブジェクト命名に於ける工夫



## COMMENT ON ... の活用

DEVELOPER CAMP



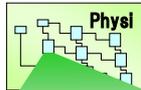
“Comment On...”を設定することで、DB可読性を向上します。



### 論理モデル

EC105:WP

- PJ番号(FK)
- WPコード
- FWBSコード(FK)
- PCWBSコード(FK)
- 原価区分
- コスト集計コード
- 機能部門コード(FK)
- 原価部門コード(FK)
- WP名称
- ...



### 物理モデル

WP

- PJ\_NO(FK)
- WP\_CD
- FWBS\_CD(FK)
- COST\_TYPE
- COST\_SUM\_CD
- FUNC\_DEPT\_CD
- COST\_DEPT\_CD
- WP\_NM
- ...

コメント情報の付与  
(ツールマクロでの自動実行)

オブジェクト名	所有者	タイプ	コメント
PJ_PIC	PMS_ADMIN	TABLE	EC103: PJ担当者
PCWBS	PMS_ADMIN	TABLE	EC104: PCWBS
WP	PMS_ADMIN	TABLE	EC105: WP

項目名	型	長さ	必須	省略時値	コメント
PJ_NO	VARCHAR2	8	<input checked="" type="checkbox"/>		PJ番号
WP_CD	VARCHAR2	8	<input checked="" type="checkbox"/>		WPコード
FWBS_CD	VARCHAR2	5	<input checked="" type="checkbox"/>		FWBSコード
PCWBS_CD	VARCHAR2	2	<input type="checkbox"/>		PCWBSコード
COST_TYPE	VARCHAR2	1	<input checked="" type="checkbox"/>		原価区分
COST_SUM_CD	VARCHAR2	30	<input type="checkbox"/>		コスト集計コード
FUNC_DEPT_CD	VARCHAR2	6	<input type="checkbox"/>		機能部門コード
COST_DEPT_CD	VARCHAR2	6	<input type="checkbox"/>		原価部門コード
WP_NM	VARCHAR2	60	<input checked="" type="checkbox"/>		WP名称
WP_REM	VARCHAR2	100	<input type="checkbox"/>		WP備考
WP_DECTS_FLG	VARCHAR2	1	<input checked="" type="checkbox"/>	'n'	WP確認フラグ

本文書の一部または全部の転載を禁止します。本文書の著作権は、著作者

# ER/Worksheet



DEVELOPER CAMP



<<レポート種類>>

- エンティティ/属性(論理/物理)
- リレーションシップ(論理/物理)
- インデックス(論理/物理)
- テーブルスペース(物理) など

<<補足>>

- ラウンドトリップを使い、大量処理に活用
- データ型、データ定義の一括変更など
- レポート機能以外にも多機能保有

EntID	エンティティ名	テーブル名	エンティティ定義	AttrID	SEQ	属性名	カラム名	PK	FK	必須	属性定義	論理データ型	
5	レジスター	REGI		22	1	店舗番号	STORE_NO	Yes	Yes	Yes		VARCHAR	
				10	2	レジスター番号	REGI_NO	Yes	No	Yes		VARCHAR	
				12	1	商品コード	GOODS_CD	Yes	No	Yes		VARCHAR	
7	商品	GOODS		13	2	商品名称							
				14	3	商品仕様							
				15	4	商品単価							
				1	1	店舗番号							
				3	2	店舗名称							
3	店舗	STORE		4	3	店舗住所							
				5	4	店舗電話番号							
				17	5	部門番号							
				6	1	従業員番号							
1	従業員	EMP		7	2	従業員氏名							
				16	3	部門番号							
				24	1	店舗番号							
6	販売実績	SALE.ACT		23	2	レジスター番号							
				35	3	販売年月							
				25	4	販売実績番号							
				27	5	販売担当者							
				34	6	販売年月日							
				26	7	商品コード							
				29	8	販売数量							
				30	9	<商品単価							
				31	10	<小計							
				32	11	消費税金額							
				33	12	<販売合計							

本文書の一部または全部の転載を禁止します。本文書の著作権は、著作者に帰属します。

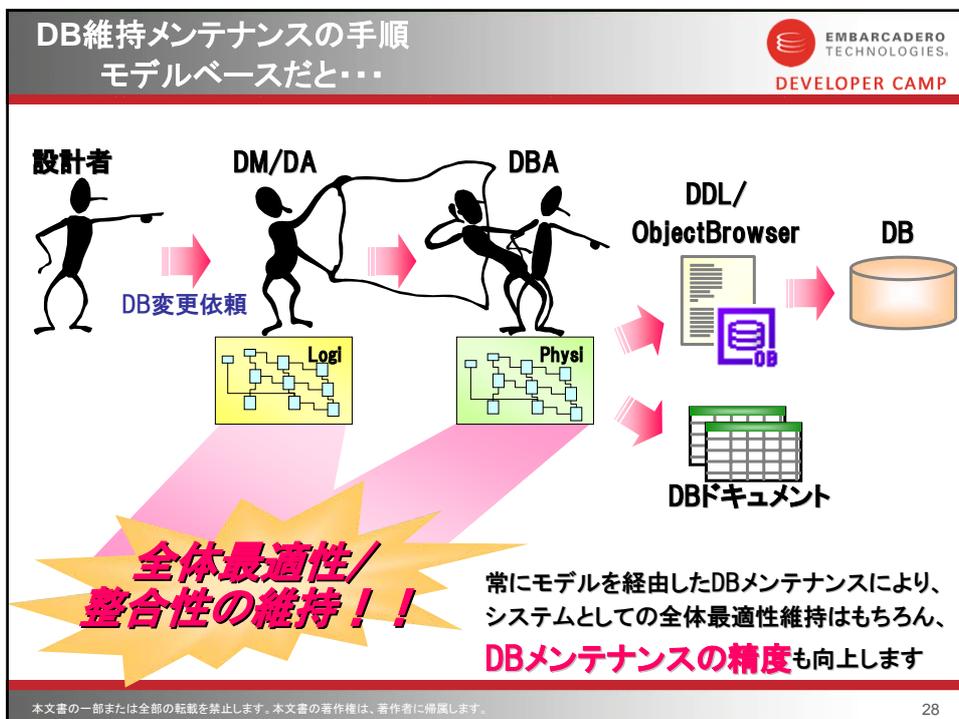
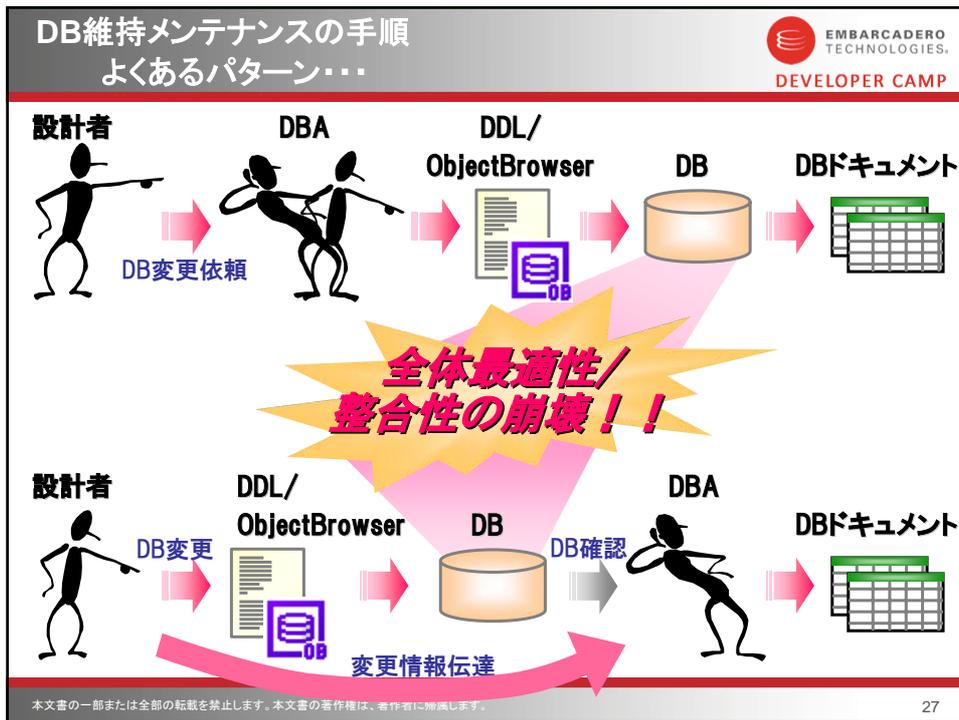
その他として、下記のツールを準備し、効率化/高精度化を図っています。

一時表テーブルDDL整形ツール(VB)



テーブルサイズ設定(マクロ)





## DB構築“大変”バロメーター（一般的）

### 論理設計

- 構造設計
- データ定義
- データ型設定
- 想定件数算出

### 物理設計

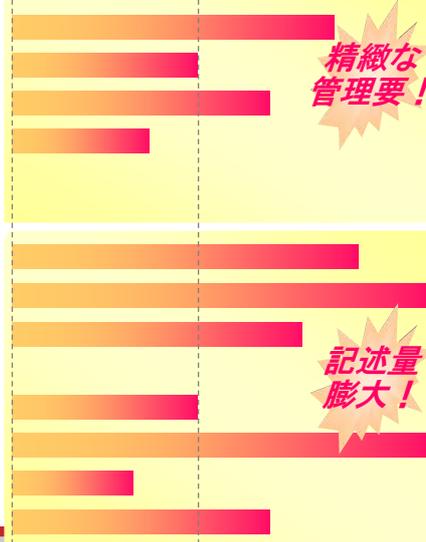
- オブジェクト命名
- スキーマ設定
- テーブルサイズ算出
- Index設計

### DDL記述



- Create Table
  - Comment On ...
  - PK/FK
- Create View
  - Comment On ...
- Create Synonym
- Create Index

### “大変”バロメーター



本文書の一部または全部の転載を禁止します。本文書の著作権は、著者に帰属します。

29

## DB構築“大変”バロメーター（ツール活用）

### 論理設計

- 構造設計
- データ定義
- データ型設定
- 想定件数算出
- 論物辞書

### 物理設計(モデル)

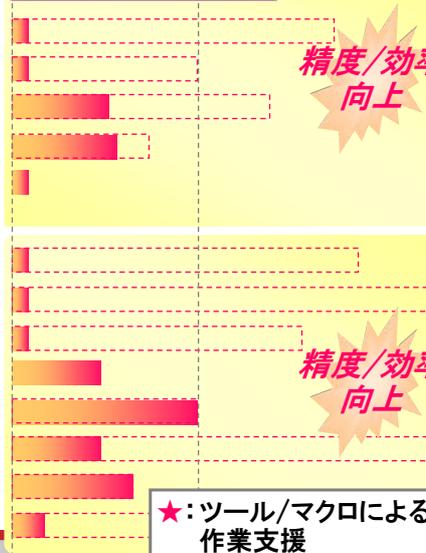
- オブジェクト命名(★)
- スキーマ設定(★)
- テーブルサイズ算出(★)
- Index設計(★)
- Comment情報付与(★)

### DDL記述



- Create Table(★)
  - Comment On ...(★)
  - PK/FK(★)
  - 一時表への変換(★)
- Create View
  - Comment On ...(★)
- Create Synonym
- Create Index(★)

### “大変”バロメーター



本文書の一部または全部の転載を禁止します。本文書の著作権は、著者に帰属します。

相変わらずDBAは、重要でありながら希少な存在です。  
DBAの負荷を軽減することで...

† マルチプロジェクトでの稼働

† DBリリーススピードの向上

† DB構築精度の向上

と言ったメリットを得ることができます。

モデルベースであること…

† ルール、思想をDBに埋め込む！

† システムライフサイクルを押さえる！

ツール活用で楽々！

† DBを開発のボトルネックにしない！

† DBを混乱のトリガーにしない！

## *Data Modeling on ER/Studio*