



開発版

テクニカルリファレンス
マニュアル

Ver 5.1.0.0

株式会社 ニューコム

2010年11月

《法的な注意事項》

本書に記載した内容は、将来予告なしに変更することがあります。

本ドキュメントの作成にあたっては細心の注意を払っていますが、本書マニュアルの記載事項の誤り、または欠落により発生した損害については責任を負いかねますのでご了承ください。

本書は、株式会社ニューコムが事前に承諾している場合を除き、製品の使用許諾契約書に記載の条件に同意をされたライセンス所有者に対してのみ供給されるものです。

《著作権表示》

本書に関する著作権は、株式会社ニューコムへ独占的に帰属します。

株式会社ニューコムが事前に承諾している場合を除き、形態および手段を問わず、本書またはその一部を無断で複製・転載・改編することを禁止します。

《商標表示》

本書に記載されている「dbSheetClient実行版」、「dbSheetClient開発版」、「dbSheetClientサーバ版」は株式会社ニューコムの製品です。

本書に記載されているマイクロソフト製品（Windows、Office、Excel等）は、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。

その他記載の各社の社名、製品名およびサービス名は、各社の商標または登録商標です。

§ はじめに

本マニュアルは dbSheetClient の動作環境や内部処理、データベースアクセス制御方法などの技術情報について記載しています。

第1章 Web アプリケーション開発での Excel 関する注意事項

Excel を Web アプリケーションとして使用する場合に注意することや制限事項について説明します。

第2章 内部処理フローと処理の定義方法

dbSheetClient の動作原理や内部処理フローについて、dbSheetClient による業務処理の定義方法（プログラミング方法）について説明します。

第3章 照会系データベースの展開方式の定義について

dbSheetClient で採用する同時実行制御機能と、データ更新処理の定義方法について説明します。

第4章 更新系データベースの仕組みについて

dbSheetClient の起動時間や実行処理のレスポンス改善、データベース更新処理の方法などについて説明します。

第5章 同時実行制御機能（排他制御）について

dbSheetClient が提供する同時実行制御のしくみと定義の方法について説明します。

第6章 レスポンス改善のための対策

dbSheetClient の起動時間や実行処理のレスポンス改善、データベース更新処理の方法などについて説明します。

第7章 機能別動作原理と定義について

ハイパーテスク実行の動作原理と定義の方法について説明します。

第8章 IOTG(アイオーティージー)の仕組みと作成手順について

dbSheetClient で、データベースの読み込みや更新用のタスクを自動的に作成する機能の IOTG 機能について説明します。

第9章 制限事項とトラブルシューティング

dbSheetClient の制限事項やトラブルシューティングについて説明します。

付録

改版履歴を記載します。

§ 目 次

§ はじめに	2
第 1 章 Web アプリケーション開発での Excel 関する注意事項	6
1.1 セル書式と値の自動変換に関する注意点	6
1.1.1 データ入力時の注意事項	6
1.1.2 Excel の自動変換機能（代表的な現象）の注意事項	7
1.1.3 数式で他のセルを参照（例、数式が”=A1”）した時の注意事項	7
1.1.4 その他（他のソフトとの互換性）の注意事項	7
1.2 Excel の仕様および制限	8
1.3 Excel での高度な入力チェックの方法	9
1.3.1 入力時に操作ガイドメッセージを表示する	9
1.3.2 選択範囲外の入力を可能にする方法	10
1.3.3 複合的にエラーチェックをする方法	10
第 2 章 内部処理フローと処理の定義方法	12
2.1 dbSheetClient の内部処理フロー	12
2.1.1 タスク処理の高速化	12
2.1.2 プロジェクト選択後のブックオープン処理	13
2.1.3 メニューをクリックしたときの処理	14
2.1.4 ボタンをクリックしたときの処理	15
2.1.5 タスク内の処理	16
2.1.6 タスクタイプ別の処理	17
第 3 章 照会系データベースの定義について	22
3.1 一括展開方式	23
3.2 個別指定展開方式	24
3.3 付加展開方式	25
3.4 コピーによる展開方式	26
3.5 複数展開方式	27
3.5.1 「展開定義」の「セル展開範囲定義名」に名前を指定する	27
3.5.2 「展開定義」の「展開属性」を“全選択データ一括保持”にする	27
第 4 章 更新系データベースの仕組みについて	28

4.1 DBS 同時実行制御フィールド	28
4.2 更新の種類	29
4.3 「DBS 同時実行制御フィールド」を利用したデータ更新処理	30
4.4 リスト（表）形式の場合のデータ更新方法	31
4.5 カード（単票）形式の場合のデータ更新方法	31
第 5 章 同時実行制御機能（排他制御）について	32
5.1 楽観的同時実行制御（楽観的ロック方式）	33
5.1.1 楽観的同时実行制御のためのデータベースの定義	33
5.1.2 楽観的同时実行制御のための Excel シートの定義	34
5.1.3 楽観的同时実行制御のための定義シートの定義例	36
5.2 悲観的同時実行制御（悲観的ロック方式）	37
5.2.1 悲観的同时実行制御のしくみ	38
5.2.2 悲観的同时実行制御のためのデータベースの定義	39
5.2.3 悲観的同时実行制御のための Excel の定義	39
5.2.4 悲観的同时実行制御のための定義シートの定義例	39
5.2.5 悲観的同时実行制御の仕様上の注意	40
5.3 同時実行制御を行わない	41
5.3.1 同時実行制御を行わない場合のデータベースの定義	41
5.3.2 同時実行制御を行わない場合の Excel の定義	42
5.3.3 同時実行制御を行わない場合の定義シートの定義例	43
5.4 データ更新のまとめ	44
5.4.1 更新処理の種類と設定方法	44
5.4.2 タスクタイプ属性が「更新」のとき	45
5.4.3 タスクタイプ属性が「指定更新」のとき	46
5.5 更新処理の注意事項	47
5.5.1 Null 値を取り扱うときの注意事項	47
5.5.2 更新処理時のデータ型とセル書式設定の注意事項	48
5.5.3 更新処理時の処理スピードについて	51
第 6 章 レスポンス改善のための対策	52
6.1 起動時間や実行時のレスポンス改善のための対策	52
6.1.1 プロジェクトの起動までの処理と、起動時間短縮のための対策	52
6.1.2 実行時のレスポンス改善の各種対策方法について	54
6.1.3 Excel の使い方として	55

第 7 章 機能別動作原理と定義について	56
7.1 ハイパーテスク実行の原理と定義方法	56
7.1.1 ハイパーテスク実行の動作原理	56
7.1.2 ハイパーテスク実行の定義方法	58
7.1.3 タスクタイプ：426. ハイパーテスク実行設定 の使用方法	59
7.1.4 ハイパーテスク実行を利用するときの注意点	60
第 8 章 IOTG(アイオーティージー)の仕組みと作成手順について	61
8.1 IOTG(アイオーティージー)の仕組み	61
8.1.1 IOTGの目的	61
8.1.2 IOTGの機能	61
8.1.3 IOTGの使用制限	61
8.2 基本的な定義シートの作成手順	62
8.2.1 作成されるプロジェクト定義情報	68
8.2.2 読込用タスク TG_T1_Read	70
8.2.3 更新用タスク TG_T2_Write	75
8.2.4 作成されるシート情報	80
8.2.5 作成されるプロジェクト定義情報	84
8.2.6 読込用タスク TG_T1_Read	86
8.2.7 更新用タスク TG_T2_Write	87
8.2.8 作成されるシート情報	88
8.3 読込みのみの場合の定義シートの作成手順	90
8.3.1 更新機能の削除	90
8.3.2 読込機能の変更	96
第 9 章 制限事項とトラブルシューティング	102
9.1 制限事項	102
9.2 トラブルシューティング	106
§ 付 錄	110
A. 改版履歴	111

第1章 Web アプリケーション開発での Excel 関する注意事項

dbSheetClient では、Excel シートを UI として使用します。そのため、Excel を使用で注意点を下記にまとめました。

1.1 セル書式と値の自動変換に関する注意点

Excel のセル書式により、自動変換される内容が異なります。

1.1.1 データ入力時の注意事項

No	セル書式	入力文字	変換後の値	変換内容	変換後の書式	対策
1	標準	1/10 1-10	1月10日	日付に変換	m"月"d"日"	セル書式を文字列にする
2		(1)	-1	負数の -1 になる	標準(数値)	
3		0488158460	488158460	数値に変換され、先頭の0がなくなる	標準(数値)	
4		1000,2000	10,002,000	3桁区切りの数字に変換	標準(3桁区切り数値)	
5		12:05	12:05	時刻書式に変換される	時刻(y/mm)	
6		H19.10.01	H19.10.1	日付の書式に変更され、日の01が1に変わる	日付(和暦 [\$-411]ge.m.d)	
7		2007/10/01	2007/10/1	日付の書式に変更され、日の01が1に変わる	日付(yyyy/m/d)	
8	数値	1/10	0.1	除算の結果		セル書式を文字列にする
9		1-10	39092	日付シリアル値に変換		
10		0488158460	488158460	先頭の0がなくなる		
11		1000,2000	10002000	3桁区切りのない数字に変換		
12		12:00	0.5	時刻書式に変換される		
13		H19.10.01	39356	日付シリアル値に変換		
14		2007/10/01	39356	日付シリアル値に変換		

※ セル書式をクリアすると、書式は「標準」になり、データ入力時に自動変換が機能するため注意してください。

1.1.2 Excel の自動変換機能(代表的な現象)の注意事項

No	セルの書式	入力文字	変換後の値	変換内容	対策
1	標準	(C)	⌚	⌚ が表示される	オートコレクションのオプションを変更する
2	標準	the	The	先頭文字のみ大文字に変換	ツール→オートコレクトのオプション →入力オートフォーマットタブの対応するチェックを外す (「入力中に自動修正する」など)
3	標準	URL	URL	ハイパーリンク付きに変わる	

1.1.3 数式で他のセルを参照(例、数式が”=A1”)した時の注意事項

No	数式のセル	入力文字	変換後の値	変換内容	対策
1	標準	未入力	0	0 が表示される	
2	数値	未入力	0	0 が表示される	数式を入力する =IF(A1="","",A1)
3	日付	未入力	1900/1/0	1900/1/0 が表示される	

1.1.4 その他(他のソフトとの互換性)の注意事項

No	セルの書式	入力文字	原因	対策
1	標準	1900/2/29 1440時間	1900年はうるう年のない年であるけれど、他のソフトとの互換性を持たせたために入力可能です。 また、Excelの内部では、1900/1/1からの経過日数(シリアル値)で管理しています。よって1440時間=60日となりますので、”1900/2/29”と同じになります。	①DATEVALUE関数を用いて、シリアル値が”60”的場合と変換エラー(#VALUE!)は、エラーとする ②ツール→オプション→計算方法の「1904年から計算する」をチェックすることで回避可能です。ただし、基準年が変わりますので、内部で管理するシリアル値が変わります。これにより、データベースや他のブックとの互換性について注意が必要です。
2	標準	=1*(0.5-0.4-0.1)	Excelの機能制限で、小数点以下の計算に誤差が発生するため	数式を小数点第2位で丸める =round(1*(0.5-0.4-0.1),2)
3	ユーザ定義	2009/08/25 16:40:35.240	Excelでは、秒以下の時間は、100分の1秒単位までしか扱えません。また、更新をすると、秒単位までになります。 (セルの書式を”yyyy/mm/dd hh:mm:ss.000”にした場合)	案1.セル書式を文字列とテーブルの型を文字型にする。 案2.変換するSQL文を作成する。

1.2 Excel の仕様および制限

制限事項	Excel2003 以前	Excel2007 以降
最大行数	65,536 行	1,048,576 行
最大列数	256 列	16,384 列
行の高さ	0~409 ポイント (1 ポイント=0.35mm)	同左
列の幅	0~255 文字 (最大 1,726.5 ポイント)	同左
セル内の最大文字数	32,767 文字 (ただし、1,024 文字を超えた分は数式バーに表示されます)	同左
再計算の順序指定	順番を指定できないため、数式多い Excel ファイル程、処理に時間がかかります。	同左

そのほかの仕様及び制限

Excel2003 : <http://office.microsoft.com/ja-jp/excel/HPO51992911041.aspx>

Excel が上位になるほど利用できる機能が増えています。開発は Excel2003 ですが、運用は Excel2000 など異なるバージョンの場合は、必ず実行版で動作確認を行う必要があります。

1.3 Excel での高度な入力チェックの方法

Excel の入力チェックは、「入力規則」を使用する方法が一般的です。しかし、入力規則だけでは、すべてを入力チェックすることができません。

入力規則による入力チェックが動作しないとき	
セルに対し入力操作をしないとき	
コピー＆ペーストなどの操作でデータを貼り付けたとき	
関数（数式）により値を取得したとき	
フォームやコントロールツールボックスでデータを取得した場合	
手動計算になっている場合	

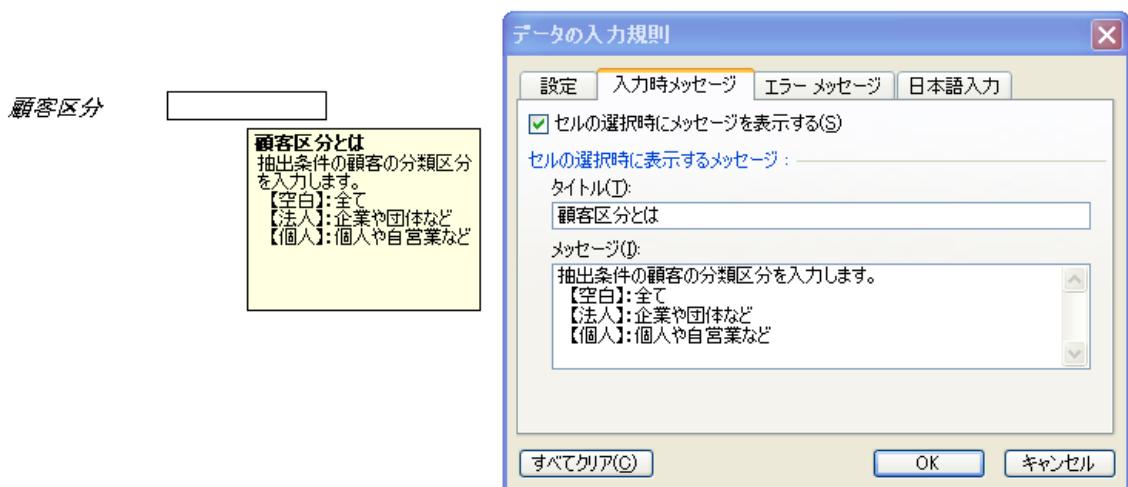
また、エラー時の動作（対応）を複数指定することができません。

データの入力規則		入力規則のスタイル	
		停止	入力されたデータが無効であることを通知し、セルに無効なデータが入力されないようにします。
		注意	入力されたデータが無効であることを通知しますが、データの入力は許可します。

1.3.1 入力時に操作ガイドメッセージを表示する

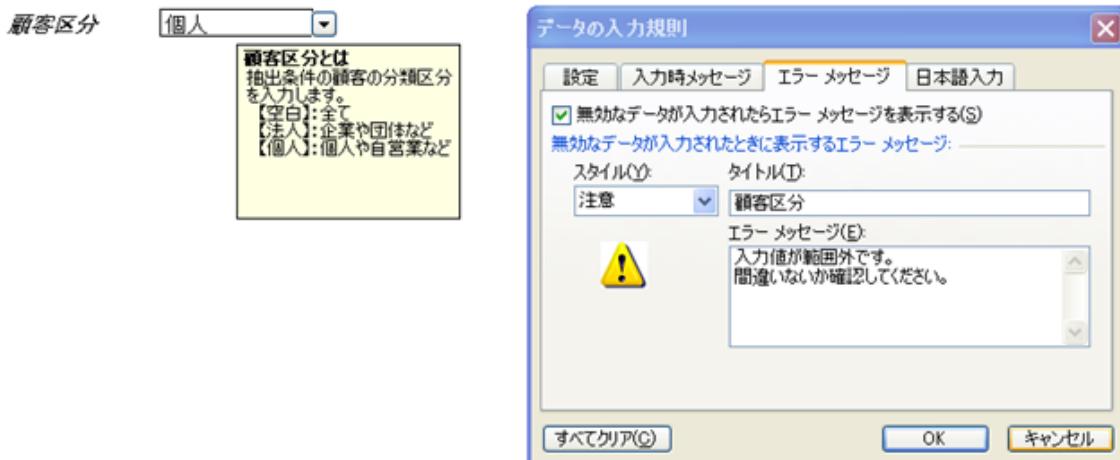
セルに入力するときに、利用者の行動を補佐するメッセージを表示します。これは誤操作防止や運用時の作業工数削減などに活用します。

設定は、「入力規則」の「入力時メッセージ」に行います。「セルの選択時にメッセージを表示する」にチェックをし、「タイトル」と「メッセージ」を入力します。



1.3.2 選択範囲外の入力を可能にする方法

「エラーメッセージ」の「スタイル」を“注意”にします。



1.3.3 複合的にエラーチェックをする方法

下記のチェックの中で A)～B)は入力規則でチェック可能ですが、C)～D)は入力規則でチェックする事が不可能です。そこで A)～C)は、入力時の「入力規則」によるチェックと、さらに更新や印刷などを行うときにも同じチェックをするようにします。そして D)は、更新前などにデータベースに問い合わせ、チェック結果を取得するようにします。このような方法でデータのチェックを強化します。

チェックの種類

- A) 入力範囲チェック
- B) 文字の種類チェック
- C) 必須入力のチェック
- D) データベースへの登録済のチェック（テーブルレコード総数：1万件）

① 入力規則と共有するチェックするための Excel シート定義

入力規則の設定で、チェック機能を共有させるには、「ユーザー設定」を使用することになります。参照先セルの結果が、“FALSE” の場合はエラーとなります。また、更新などのタスク定義においても、同じ参照先を利用して、実行制御をする仕組みを定義します。

日付 エラー=FALSE

チェック結果 (1)～(3)	FALSE	=AND(必須入力,文字種類,入力範囲)
(1)必須入力	FALSE	=OR(NOT(ISBLANK(入力日付)),入力日付<>"")
(2)文字の種類	TRUE	=OR(ISERROR(DATEVALUE(TEXT(入力日付,"YYYY/M/D"))),CELL("format",入力日付)="D1")
(3)入力範囲	FALSE	=AND(開始日<=入力日付,終了日>=入力日付)
From	2009/8/1	
To	2009/8/24	

項目=「日付」の入力チェックをするための Excel シート内の定義例です。

入力規則にて、「チェック結果」を指定するようにします。

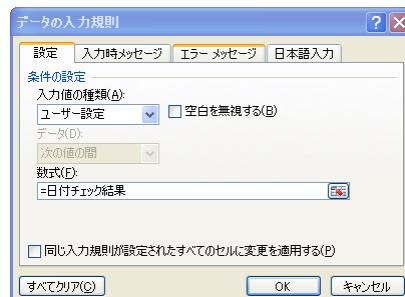
② 「ユーザー設定」の内容（“エラー=FALSE”になる数式にします）

「(1)必須入力」のチェックは、値が入れば“TRUE”になる数式を設定します。

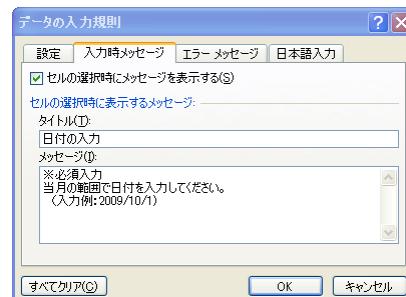
「(2)文字の種類」のチェックは、数値ならば、“ISNUMBER”関数でチェックできますが、日付の場合は、セルの書式をチェックするようにします。（日時は内部的に数値で管理しているため）

「(3)入力範囲」のチェックは、規定範囲内ならば“TRUE”になる数式を設定します。

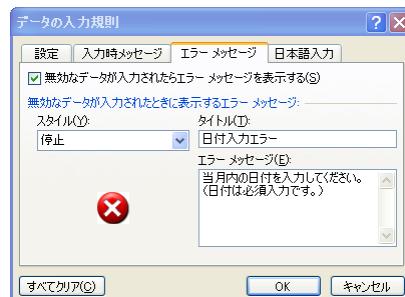
入力規則の設定例



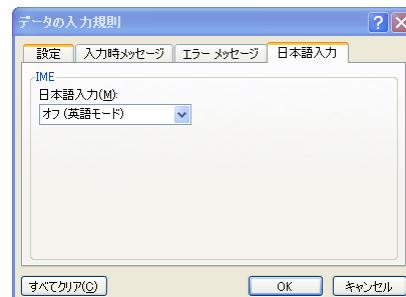
入力時、空文字をエラーにするには、「空白を無視する」のチェックを外します。



セルがアクティブになった時、ガイダンスマッセージを表示したい場合に指定します。

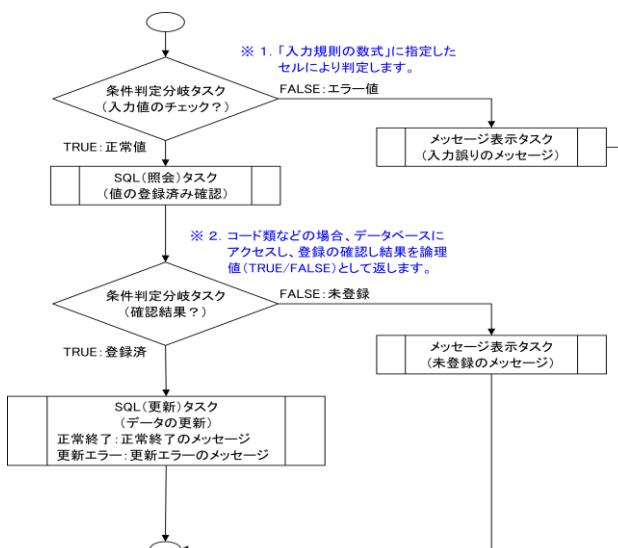


エラー時の動作やメッセージを指定します。



日本語入力のモードを指定します。

③ 更新などのボタンを押したときのタスク定義のフロー



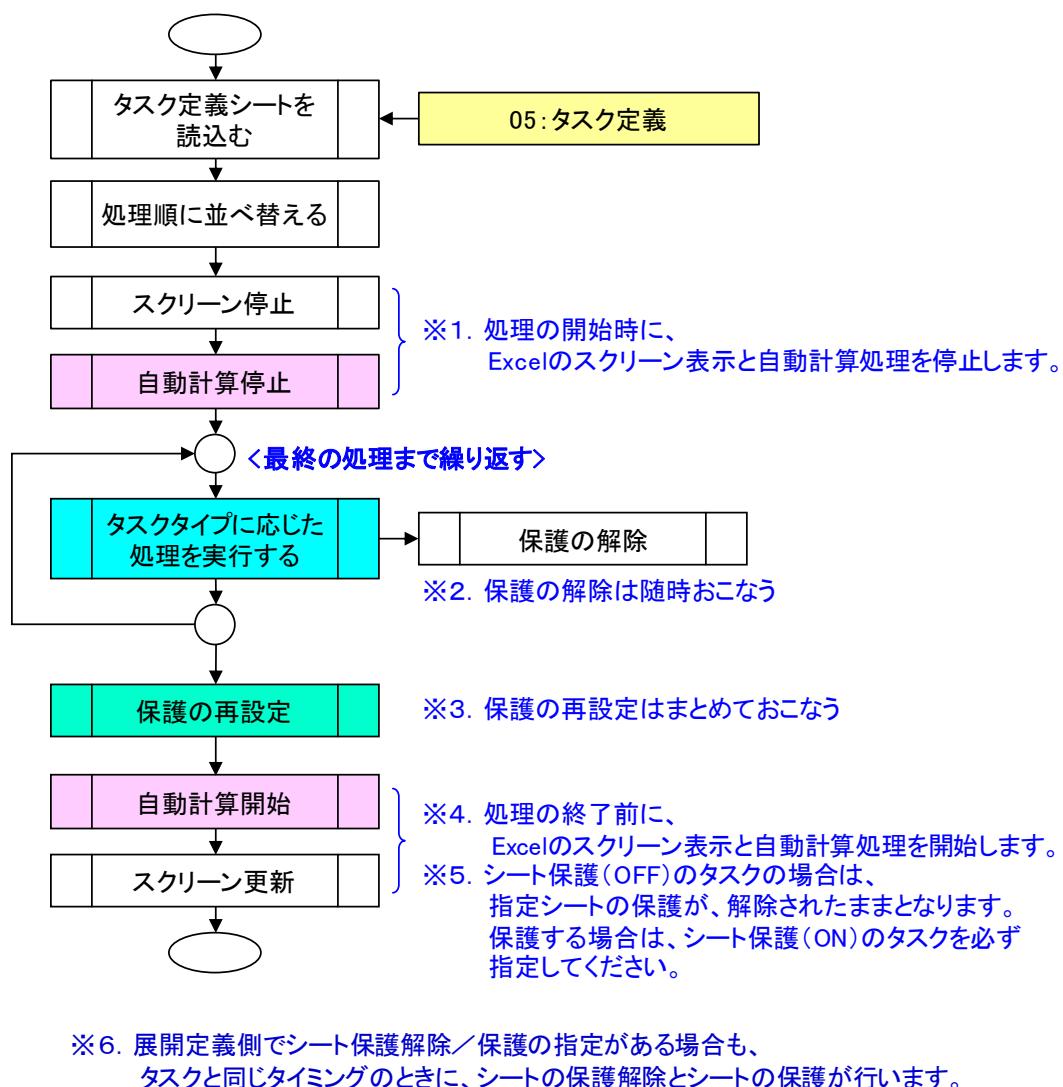
最初の「条件分岐タスク」は、入力規則にて使用したセルを使用します。

第2章 内部処理フローと処理の定義方法

2.1 dbSheetClient の内部処理フロー

2.1.1 タスク処理の高速化

- ① タスク処理を高速に行うために、タスク処理中は、Excel のスクリーン表示と自動計算を停止します。
全タスクが終了するときに、Excel のスクリーン表示と自動計算を再開します。
- ② シートの保護解除/保護の回数が多いと処理スピードに影響します。そのため、dbSheetClient では保護解除が必要なシートの処理開始時に保護解除をおこないますが、保護の再設定は、一連のタスクが終了した段階でまとめておこないます。

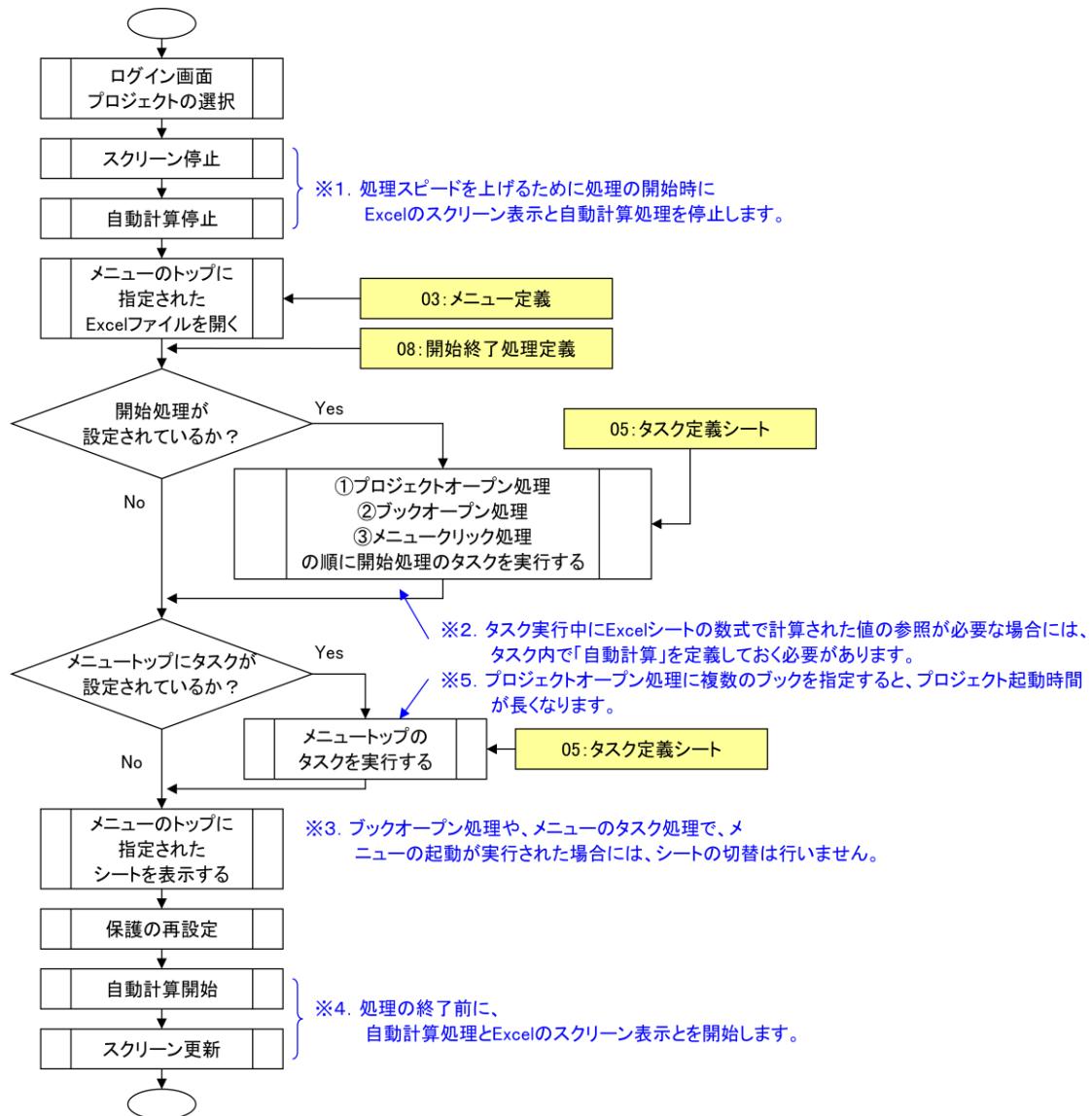


自動計算をタスク内での制御する方法については、『タスクタイプリファレンス「1.9 自動計算について」』および「2.83 自動計算（再計算実行）」、「2.84 自動計算（ON、OFF）」を参照してください。
スクリーンの表示をタスク内で制御する方法については『タスクタイプリファレンス「2.48 スクリーン制御」』を参照してください。

2.1.2 プロジェクト選択後のブックオープン処理

【処理の概要】

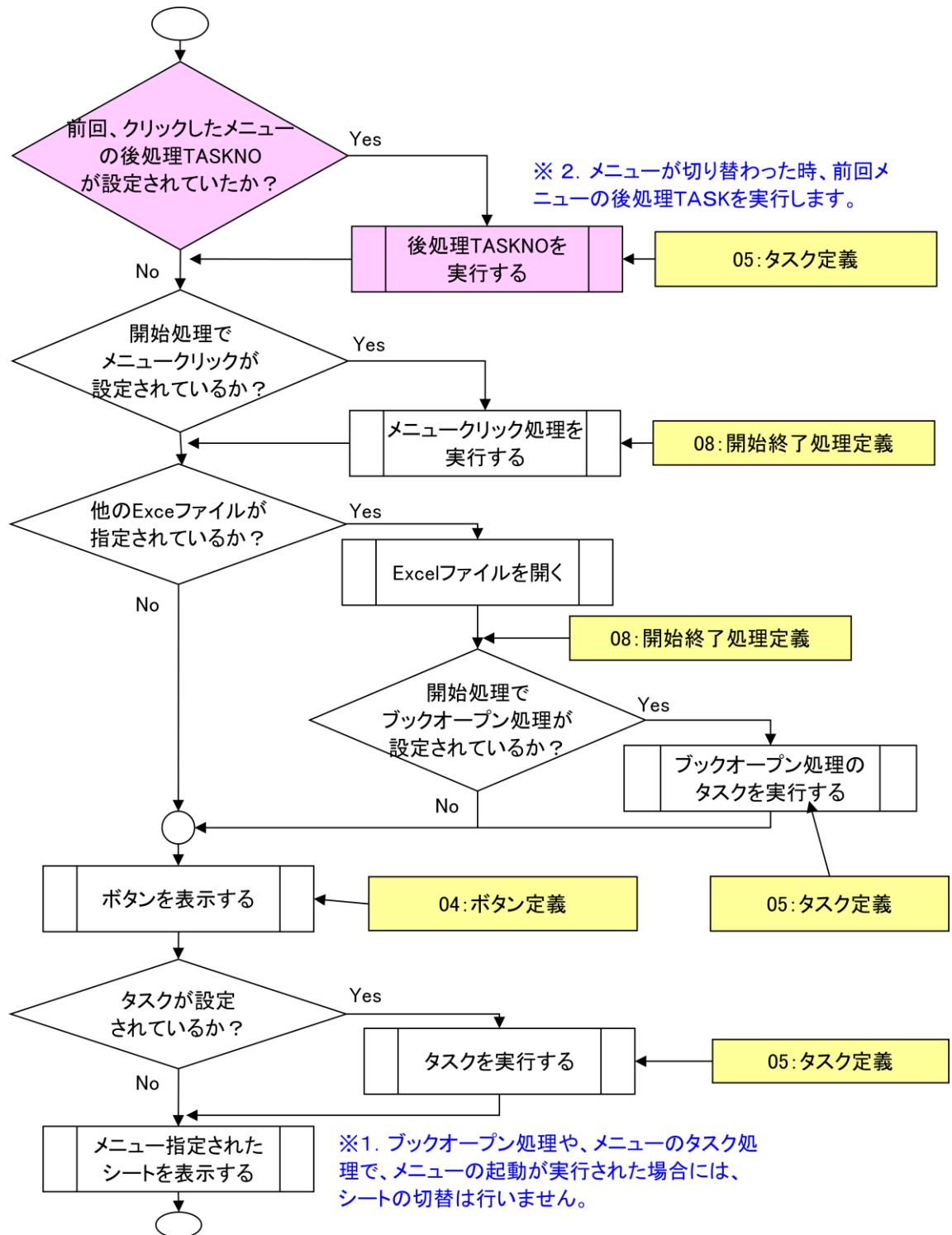
ログイン後プロジェクトを選択したのち、実行画面が表示される前に、データベースからマスターデータなどを Excel ファイルに取り込む必要があるときには、開始処理を実行します。



2.1.3 メニューをクリックしたときの処理

【処理の概要】

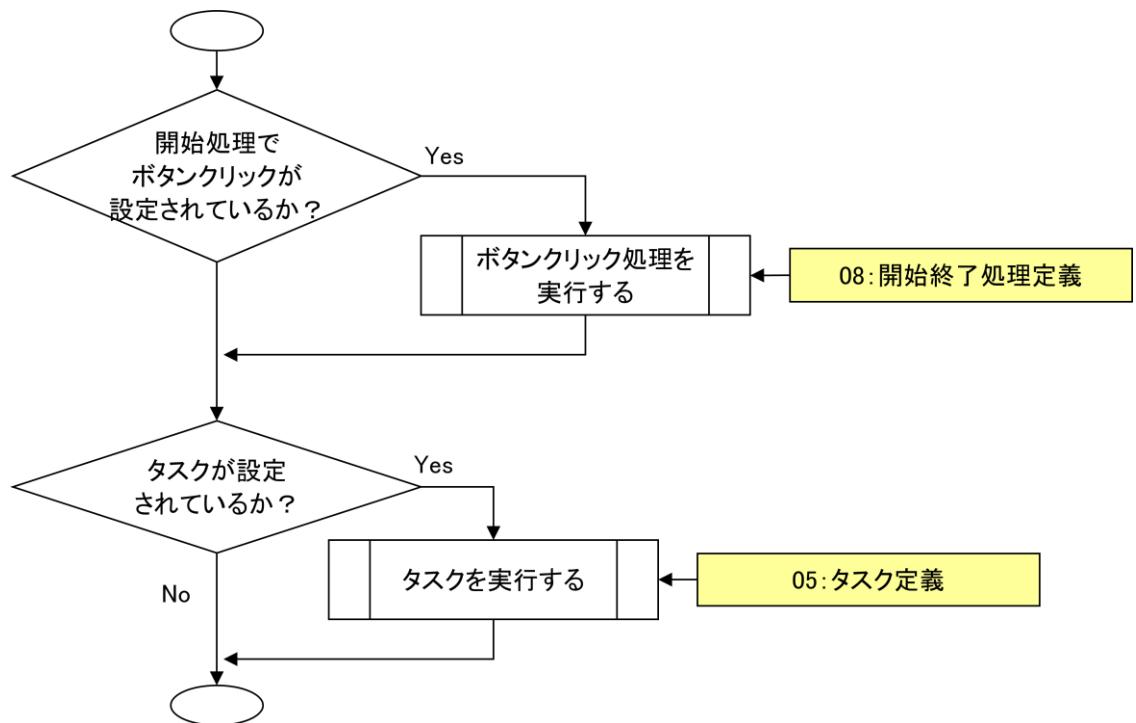
- メニューをクリックすると、別の Excel ファイルを開く必要があるときには、先に Excel ファイルを開きます。
- ボタンを表示した後、メニュー定義にタスクが設定されているときにはタスクを実行します。



2.1.4 ボタンをクリックしたときの処理

【処理の概要】

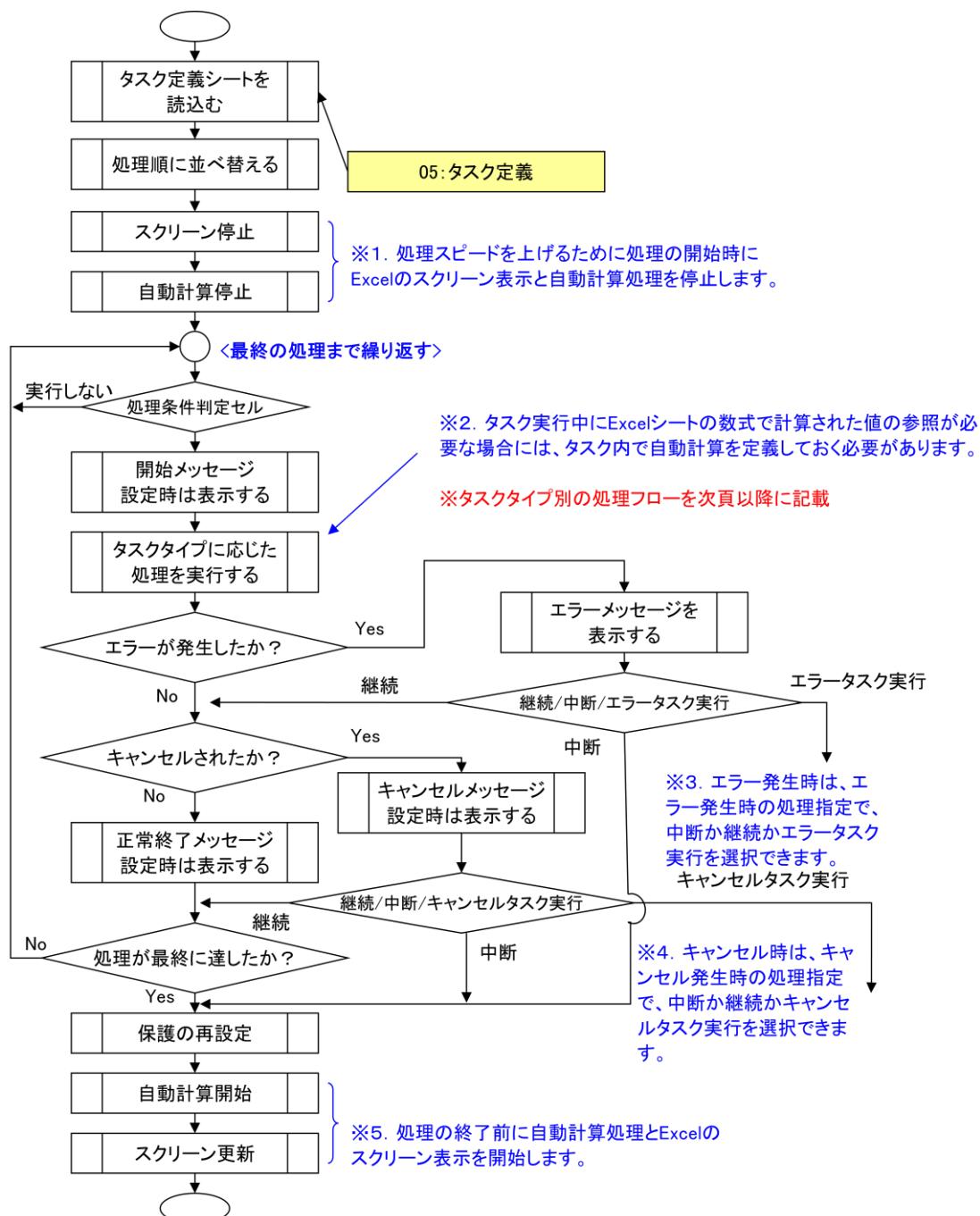
- ボタンをクリックすると、ボタン定義に設定されているタスクを実行します。
- タスクの処理については、『2.1.5 タスク内の処理』のフローに基づいて実行されます。



2.1.5 タスク内の処理

【処理の概要】

タスクには複数の処理が定義されているため、最初に処理順に並べ替えた後、タスクタイプに応じた処理を順に実行します。途中でエラーが発生したときや、キャンセルが実行されたときにはエラー処理、キャンセル処理の設定に応じた処理を実行します。



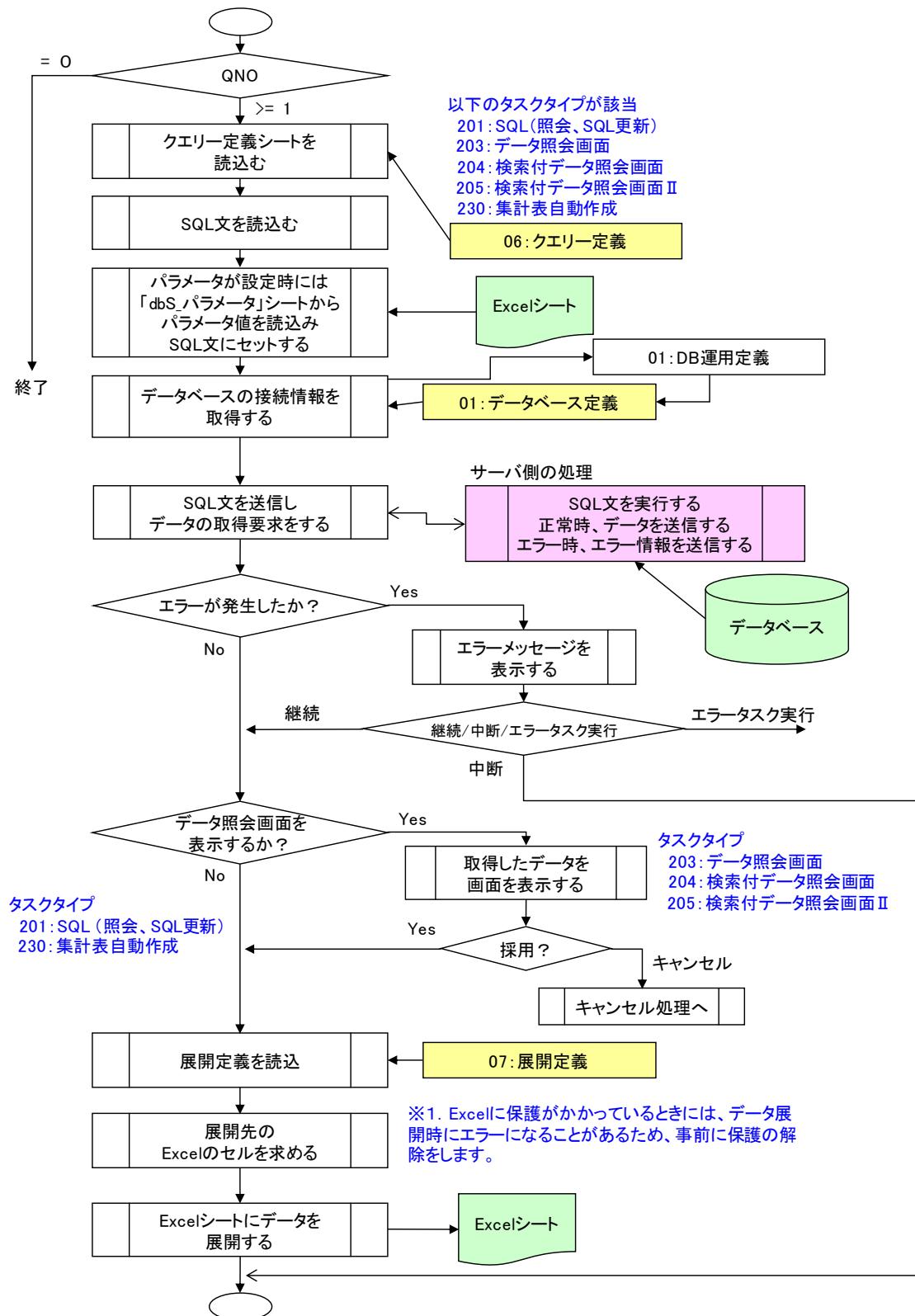
2.1.6 タスクタイプ別の処理

①データ照会系のクエリー定義を持つタスクタイプ

【処理の概要】

クエリー定義のSQL文を元に、データベースからデータを取得してExcelシートにデータを展開します。

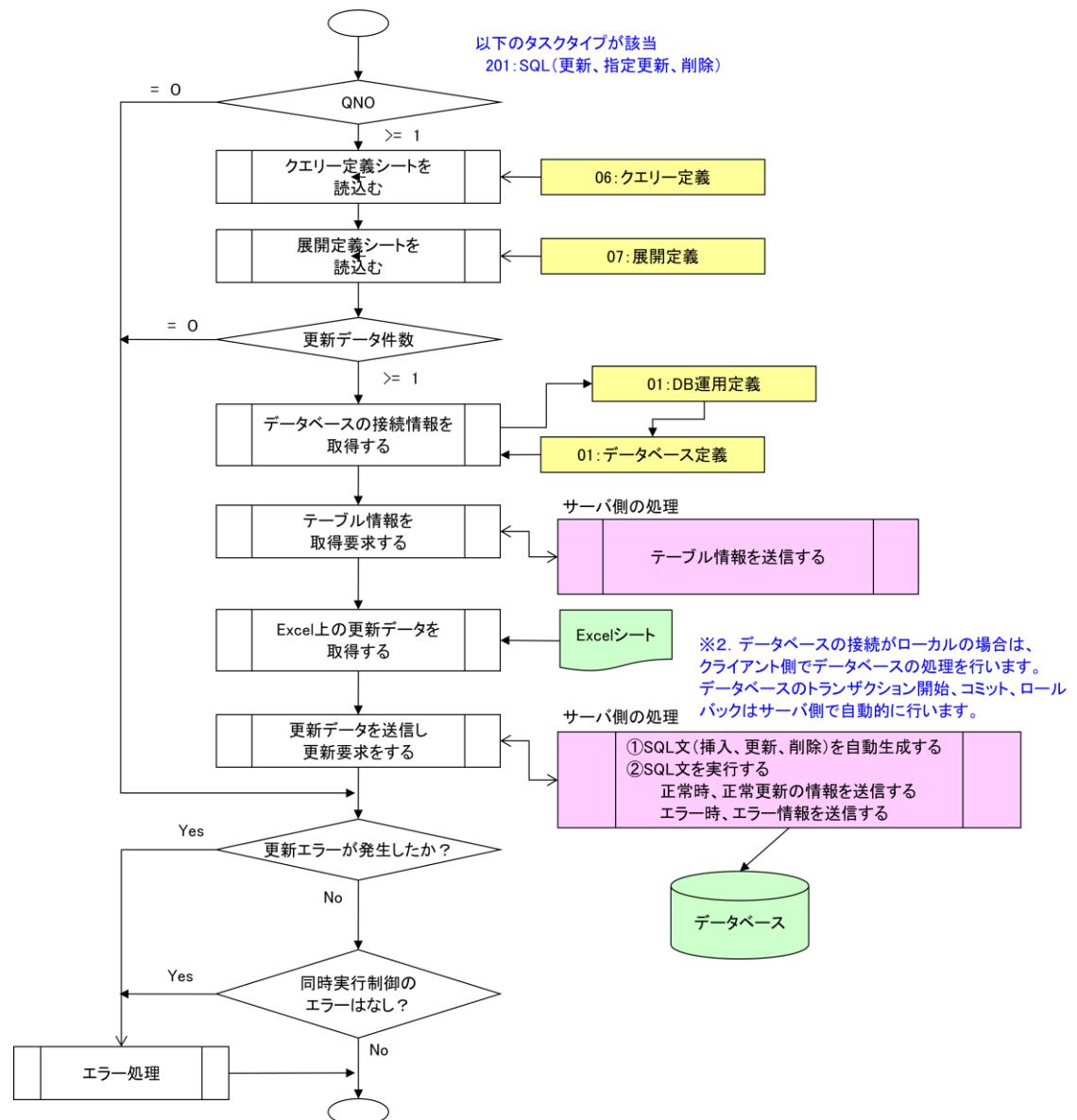
ただし、SQL 更新については、SQLを実行するだけでデータの取得は行いません。



②更新、削除のタスクタイプ

【処理の概要】

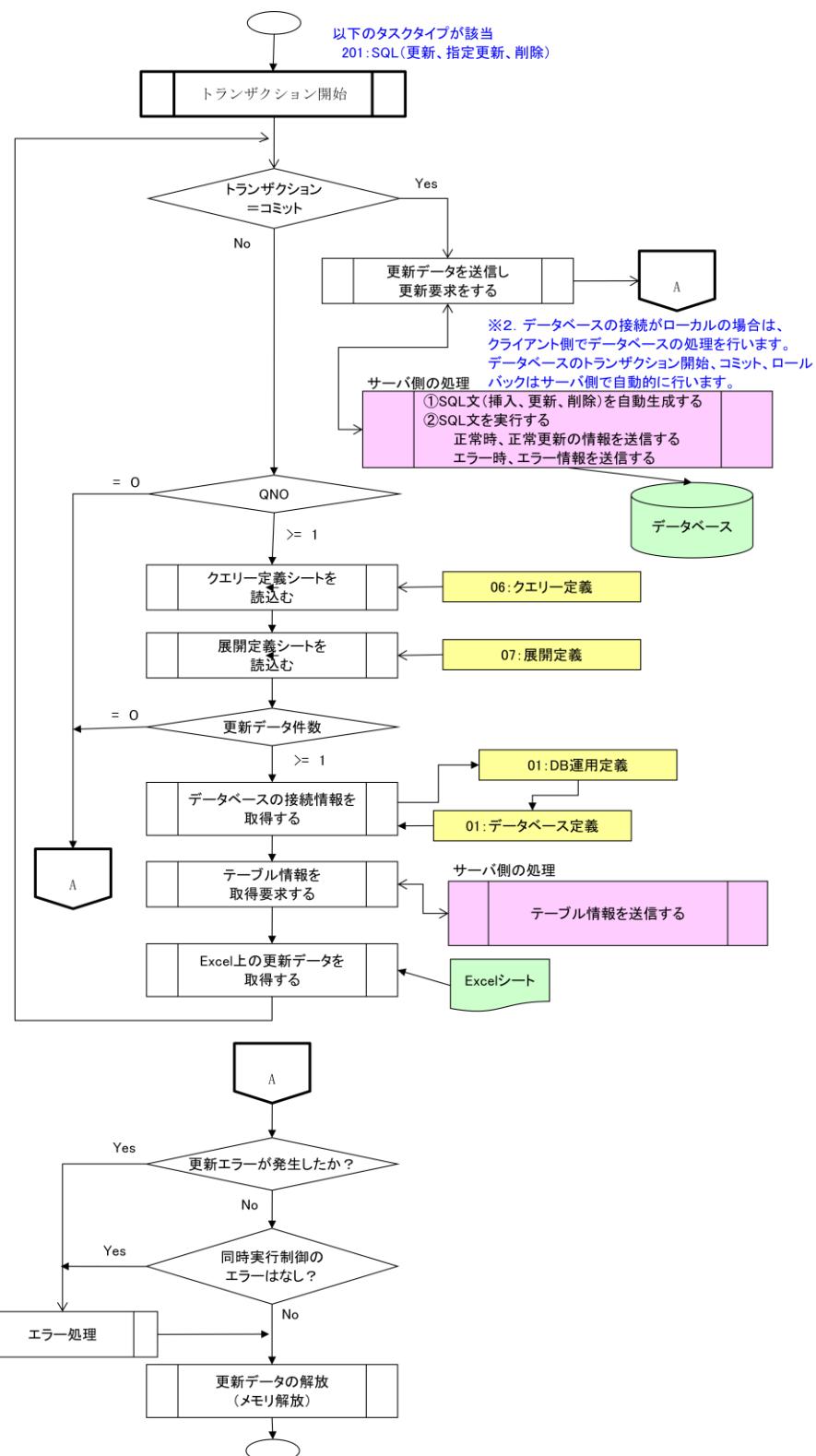
クエリーリストに定義されている更新専用シートのデータを取得して、データの更新、または削除のSQL文を自動生成して、dbSheetClient が提供する同時実行制御処理をおこないます。



③更新、削除のタスクタイプ（トランザクションによる複数テーブル更新）

【処理の概要】

複数のテーブルを更新する場合は、「トランザクション（開始）」～「トランザクション（コミット）」までにあるSQL（更新、指定更新、削除）のタスク定義を一括して実行します。基本的な処理は『②更新、削除のタスクタイプ』と同じです。

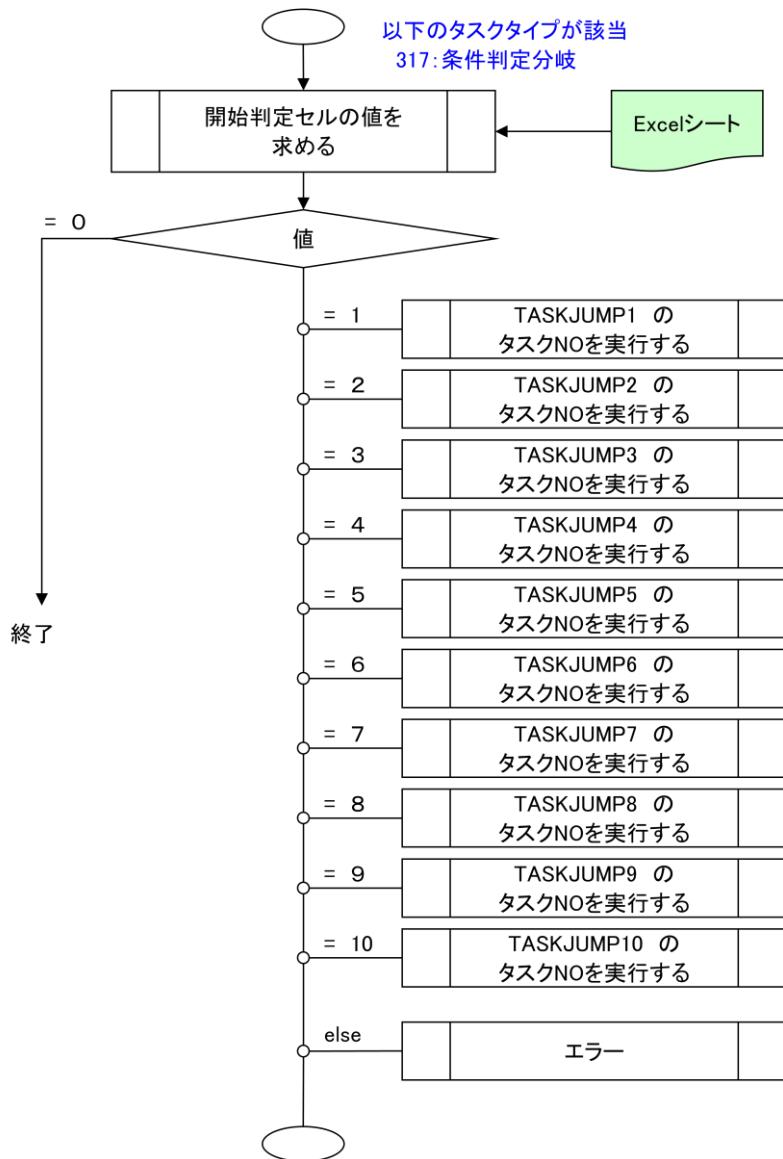


④条件判定分岐を持つタスクタイプ

【処理の概要】

開始判定セルの値（1～10）に応じて、TASKJUMP1～10に設定されているタスクを実行します。

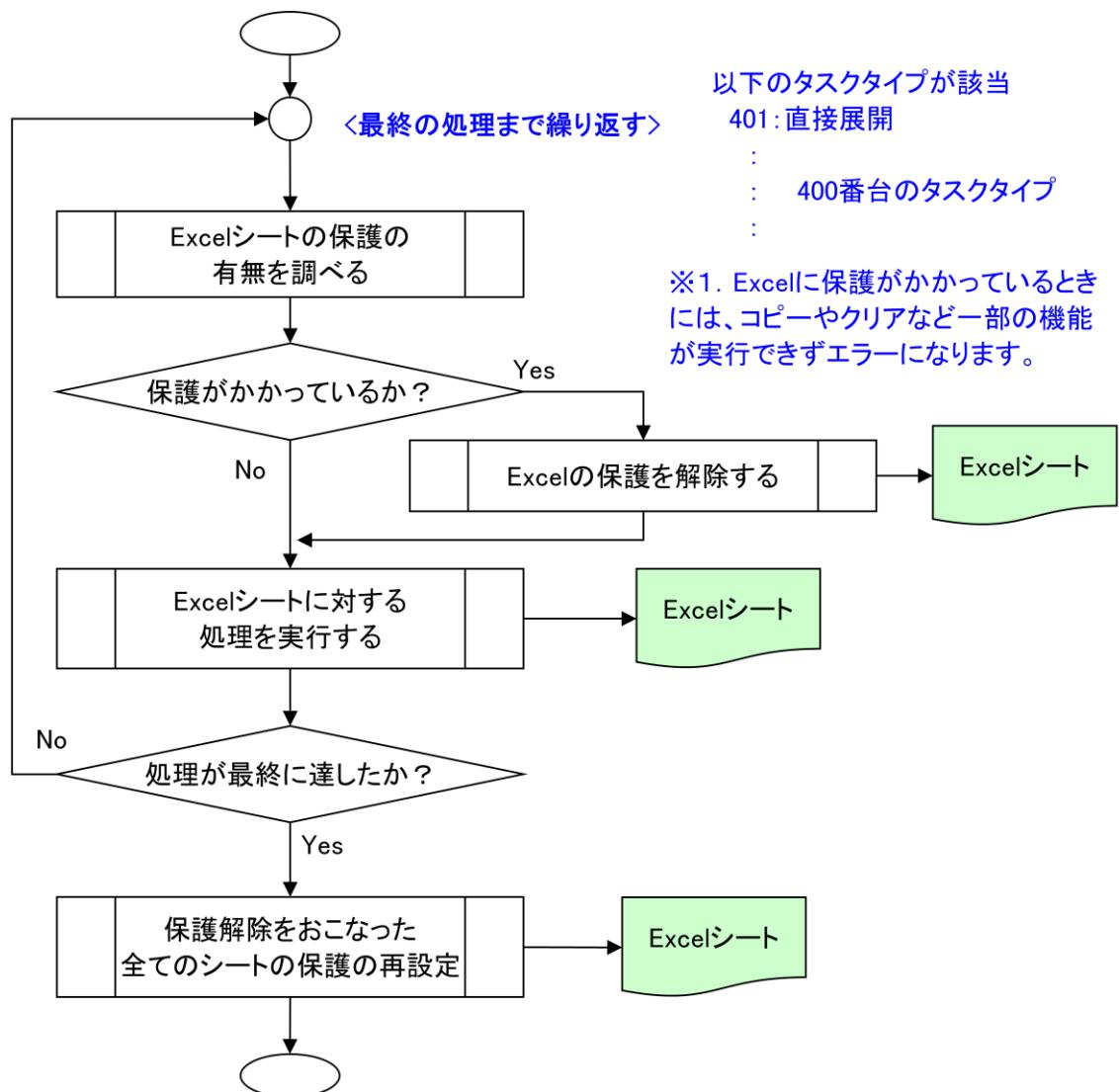
開始判定セルが0のときは何も処理をおこなわないで終了します。



⑤Excel シートの制御をおこなうタスクタイプ

【処理の概要】

- Excel やシートに対して制御をおこなう処理、たとえば、コピー、クリア、保護、ピボットテーブルなどの処理をおこないます。
- 処理の開始時に、シートに保護がかかっているかどうかを確認し、保護がかかっている場合には解除を行い、一連のタスク処理の終了時に元に戻します。



【参考】シートの保護について

保護の解除と保護の再設定は Excel 側の処理となるため、保護の解除と保護を繰返すなど量が多くなると時間を要します。

1つのタスクで複数のタスク処理を定義しているときには、タスク処理ごとに、保護の解除、再設定が行われるため、タスク処理数だけ時間が累積されることになり、処理スピードに影響を与えることがあります。

したがって開発時に、入力や表示するシート以外は保護を掛けないように工夫すると、処理スピードが改善します。

第3章 照会系データベースの定義について

dbSheetClient は、データベースから Excel シートにデータを展開するときに、クエリー定義と展開定義に定義情報を設定します。

データベースより取得したデータを Excel シートに展開方式として下記の5種類の方式があります。

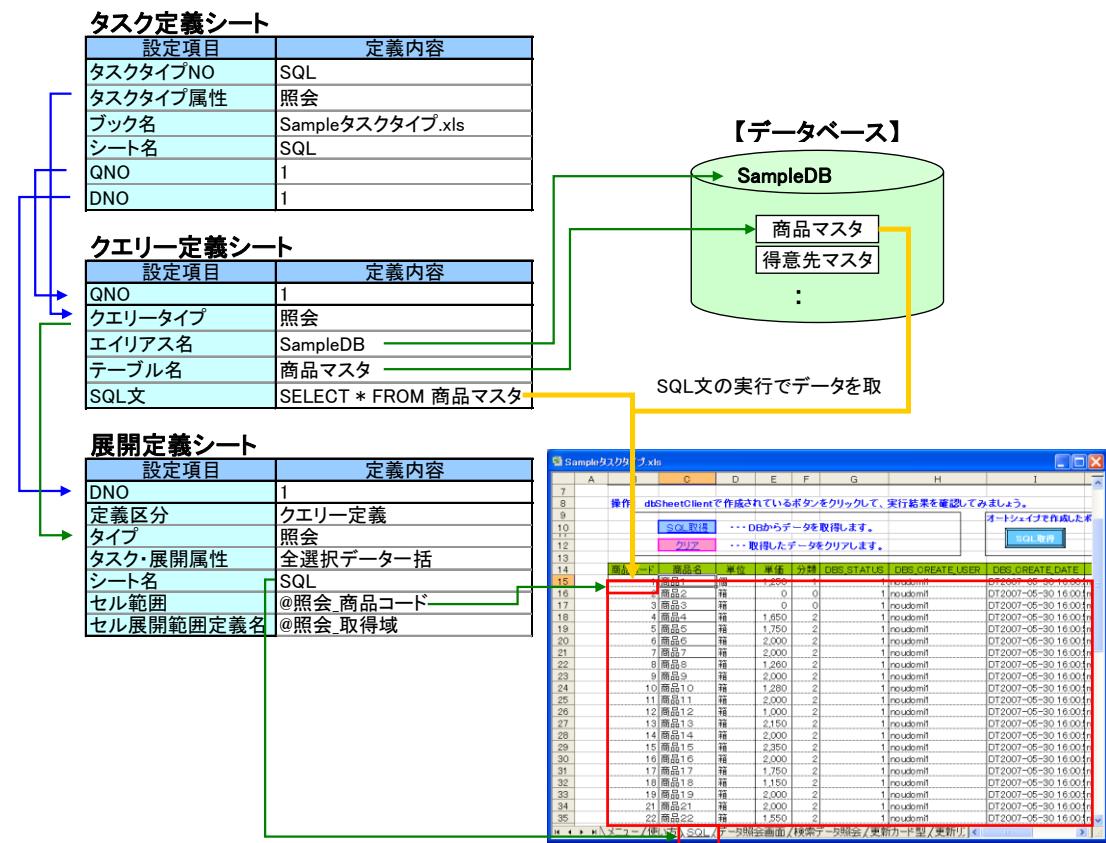
NO	展開方式	説明
1	一括展開方式	リスト(表)形式のフォームにデータを展開するための方式
2	個別指定展開方式	カード(単票)形式のフォームにデータを展開するための方式
3	付加展開方式	データ以外でフォームに“値”や“数式”を追加して展開するための方式
4	コピーによる展開方式	大量データのあるシートから別シートに展開するための方式
5	複数展開方式	1つのフォームに複数の展開場所がある場合の方式

3.1

一括展開方式

「一括展開方式」は展開定義シートの「タスク・展開属性」に「Query」、「全選択データ一括」、「全選択データ一括保持」を指定したときに実行されます。

下図は、「一括展開方式」で定義したときの、クエリー定義と展開定義、及びデータを展開する Excel シートの関係を示します。



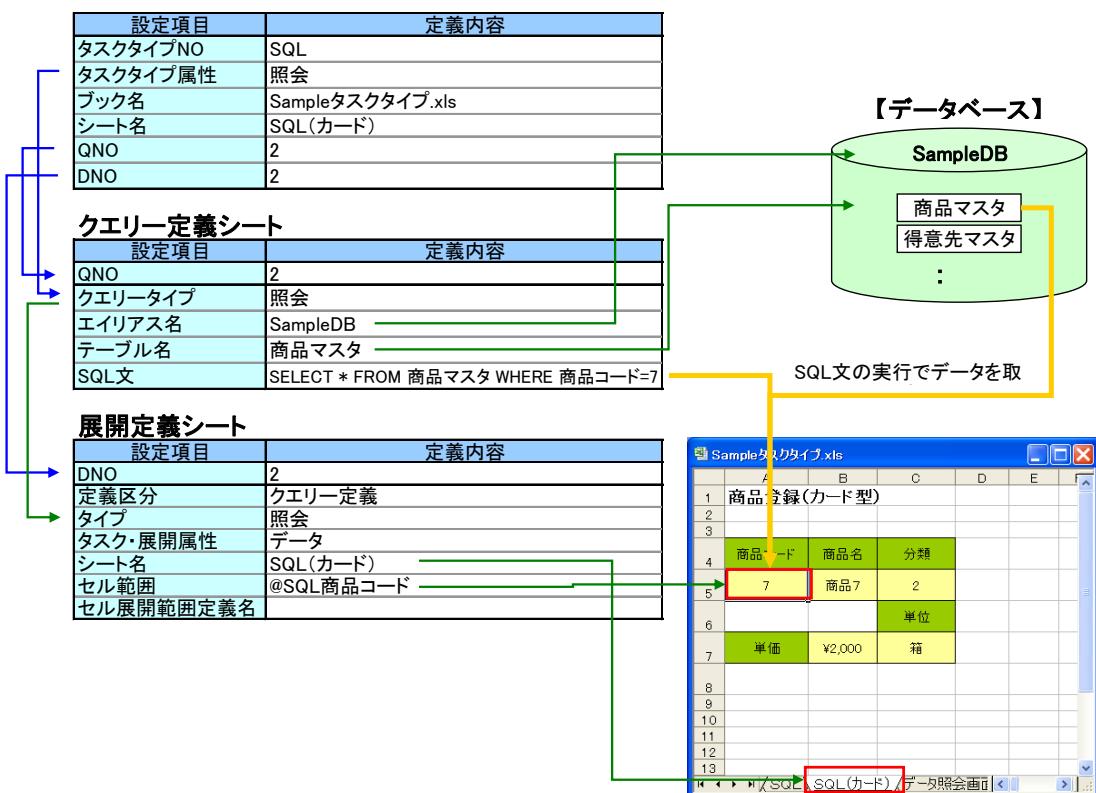
取得したデータは、展開定義シートの「セル範囲」に指定した Excel シートのセル位置を基点にして、SQL 文の SELECT 句に定義されているフィールドを列方向に展開し、取得したレコードを行方向に展開しますので、リスト型（表形式）になります。

3.2 個別指定展開方式

「個別指定展開方式」は、展開定義シートの「タスク・展開属性」に「データ」を指定したときに実行されます。また、取得するデータのレコード数が1レコードのときに定義します。この方式は、データをフィールド毎に Excel シートのセル位置を指定し、その位置に展開するための方式です。

複数のレコードが取得されたときには、1番目のレコードのみを展開します。

下図は、「個別指定展開方式」で定義したときの、クエリー定義と展開定義、及びデータを展開する Excel シートの関係を示します。



取得したデータは、展開定義シートの「セル範囲」に指定した Excel シートのセル位置毎に展開しますので、カード型（単票形式）になります。

3.3 付加展開方式

「付加展開方式」は、展開定義シートの「タスク・展開属性」に「数式」、「値」を指定したときに実行されます。『3.1 一括展開方式』または『3.2 個別指定展開方式』の展開定義と、一緒に指定することができます。

下図は、個別指定展開方式の後に「数式」を設定している例です。

【展開定義シートの定義内容の例】



社員コード	社員名	性別	生年月日	郵便番号	住所1	住所2	TEL	FAX	年齢
3	北海道 道夫	男	1972/1/23	120-0003	東京都足立区谷中	1-1-1			37

処理順NO	データ展開名	処理順NO	タスク	タイトル	ワブ	内容
1	DN0	1	データ	DN0		*
2	2110 展開 データ照会(203)個別	2	データ	データ展開名		展開 データ照会(203)個別
3	2110 展開 データ照会(203)個別	3	データ	処理順NO		11
4	2110 展開 データ照会(203)個別	4	データ	定義区分		クエリ一 定義
5	2110 展開 データ照会(203)個別	5	データ	タイプ		照会
6	2110 展開 データ照会(203)個別	6	データ	タスク 展開属性		数式
7	2110 展開 データ照会(203)個別	7	データ	データ項目名		=DATEDIF(データ照会画面,個別生年月,rd)
8	2110 展開 データ照会(203)個別	8	データ	数式OR値		年齢計算
9	2110 展開 データ照会(203)個別	9	データ	定義区分無効フラグ		データ照会画面
10	2110 展開 データ照会(203)個別	10	データ	シート名		@データ照会画面_個別年齢
11	2110 展開 データ照会(203)個別	11	無効	セル指定属性		
12				セル範囲		
13				セル展開範囲		
14				セル展開範囲定義名		
15				シート保護有無		
16				シート保護解除PW		
17				自動計算方法		
18				行別方向		
19				データ行列方向		
20				更新行OR列数指示セル 取得データ件数セル		

【参考】

Excel は、IF 関数を利用して、処理を分岐することができます。しかし、条件式が複雑になったりすると可読性が悪くなります。また、IF 関数を入れ子にして使用することができますが、入れ子の階層は 7 階層までという制限があります。

dbSheetClient は、「317:条件判定分岐」、「315:ループ」、「320:サブタスク実行」、「直接展開」などのタスクタイプを利用して、分岐やループなどの制御をしながら条件に適した“数式”や“値”を展開することができます。

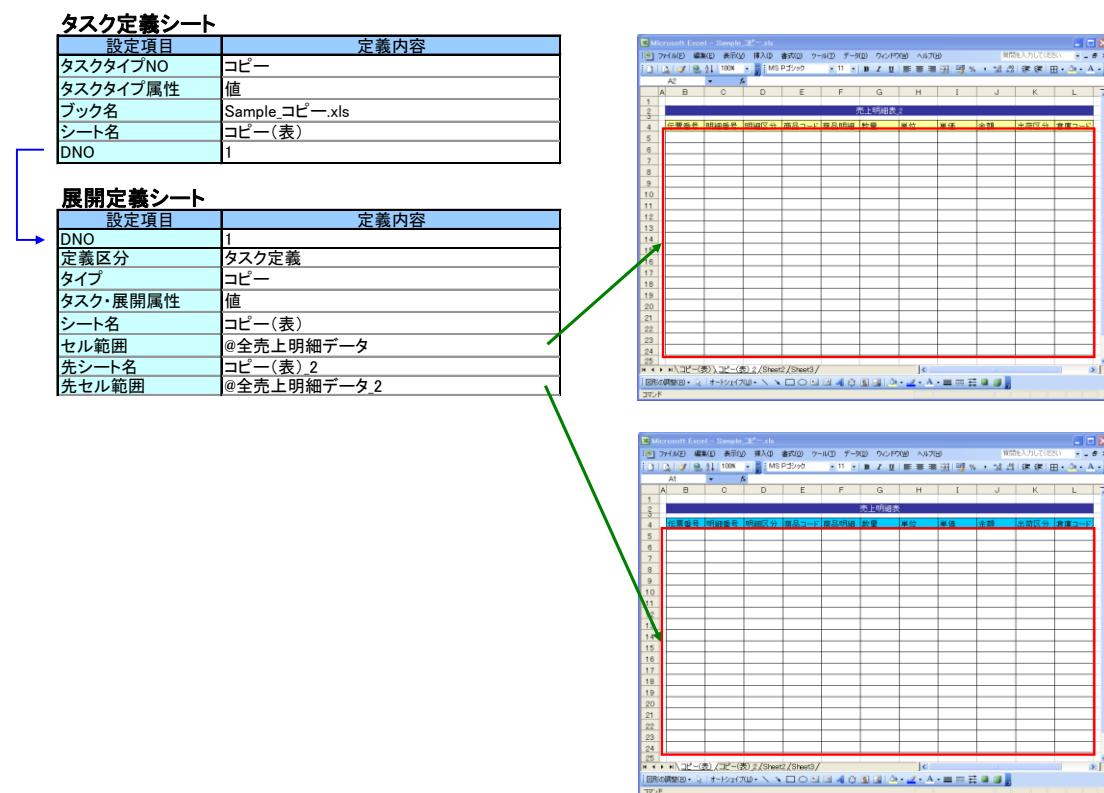
3.4 コピーによる展開方式

1,000 行を超える連続した大量データを別シートに展開する方法として、数式で参照する方法があります。しかし数式が大量にあり、Excel の自動計算が有効の場合、自動計算処理に要する時間が長くなり、パフォーマンスに影響が出ることがあります。

この場合、「410:コピー（値）」のタスクタイプを利用して、値のみを別シートにコピーする方法でパフォーマンスを改善することが可能です。

これは、Excel のコピー処理が高速であることと、数式の増加を抑えて自動計算処理の負荷を軽減することができるためです。

ただし、コピー処理ではクリップボードのメモリを使うため、一度に大量のデータをコピーするとメモリオーバーで失敗することがあるので、エラーが発生するときには、分割してコピーなどの調整が必要です。



3.5 複數展開方式

展開属性が“Query”、“全選択データー括”の場合で、「展開定義」の「セル展開範囲定義名」に指定するExcelシートの範囲を指定していない場合、展開処理において、展開後“dbs_シート名”（例、“dbs_照会”）という名前定義を実行中のExcelシートに登録します。

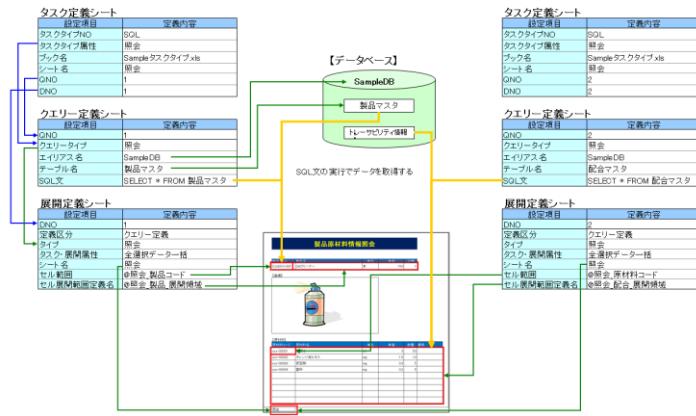
また、展開処理の順番として、①展開範囲のクリア、②データ展開、③名前定義を新しい展開範囲で更新という順序で行いますので、1つのシートに複数の展開範囲があるシートの場合は、先に処理した展開範囲がクリアされてしまう場合があります。

これを回避する方法が、2つあります。

- ①「展開定義」の「セル展開範囲定義名」に名前を指定する（推奨）
 - ②「展開定義」の「展開属性」を“全選択データー括保持”にする

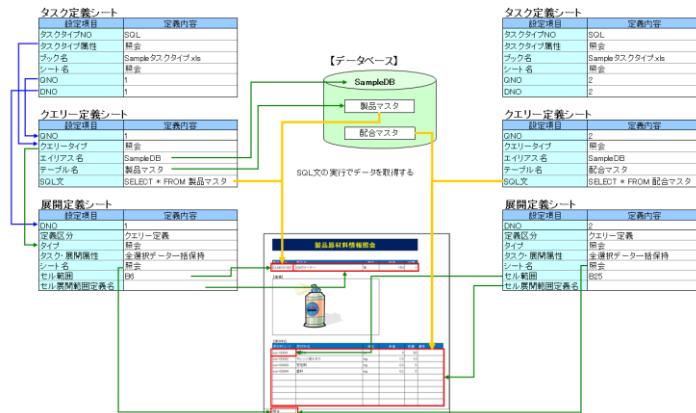
3.5.1 「展開定義」の「セル展開範囲定義名」に名前を指定する

「セル展開範囲定義名」に指定する名前は、Excel に名前定義してなくても可能です。ただし、シート内の数式として使用している場合は、事前に名前定義を登録しておく必要があります。



3.5.2 「展開定義」の「展開属性」を“全選択データー括保持”にする

展開範囲のクリアを行いませんので、展開データは保持されます。データのクリアが必要な場合は、クリアのタスク定義を追加する必要があります。



第4章 更新系データベースの仕組みについて

4.1 DBS 同時実行制御フィールド

データベースの更新は、整合性と内部統制のために、5つのフィールドをテーブルの最後に定義するようになります。この5つのフィールドのことを「DBS 同時実行制御フィールド」と言います。そして、その情報に基づいて「悲観的同時実行制御」と「楽観的同時実行制御」を実現しています。また、内部統制のための情報になります。

DBS_STATUS (可変文字型: 2桁)	<p>レコードの状態を表します。</p> <p>(1) データベースの場合</p> <p>1: 更新可能 3: 編集中(悲観ロックで使用: レコードロック中、読み取りは可能)</p> <p>(2) Excelシートの場合</p> <p>レコードの登録・更新・削除をする場合のレコード状態を表します</p> <p>1: 更新対象外のレコード(更新はおこなわない) 2: 更新対象レコード(更新をおこなう) 3: 悲観ロック中レコード(同一ユーザのみ更新対象レコード) 99: 削除対象レコード(レコードの削除をおこなう)</p> <p>※ クエリ一定義シートの「更新処理形態」が「修正データのみ」の場合です。「全データ更新」の場合は、「DBS_STATUS」の値に関係なくシート内の全データを処理します。</p>
DBS_CREATE_USER (可変文字型: 60桁)	レコードを新規に作成したときのユーザログインIDがセットされます。
DBS_CREATE_DATE (可変文字型: 30桁)	レコードを新規に作成したときの日時が 文字型で DTYYYY-MM-DD hh:mm:ss.nnn の書式でセットされます。 例) DT2007-10-01 10:20:30.397 (年月日 時分秒 1/1000秒までを表記)
DBS_UPDATE_USER (可変文字型: 60桁)	レコードを更新したときのユーザログインIDがセットされます。 レコードが誰に更新されたかを確認するためのフィールドです。
DBS_UPDATE_DATE (可変文字型: 30桁)	レコードを更新したときの日時が 文字型で DTYYYY-MM-DD hh:mm:ss.nnn の書式でセットされます。 更新された日時を1/1000秒単位で確認するためのフィールドです。

【 dbSheetClient で同時実行制御を利用するための注意事項 】

- 更新処理(タスクタイプ 201. SQL)がタスクタイプ属性「更新」、「指定更新」の場合は、「DBS 同時実行制御フィールド」が必ず必要となります。
- 既存のデータベースに対して更新処理を行うときには、「DBS 同時実行制御フィールド」の追加が必要になります。しかし、照会のみの場合は必要ありません。
- 各フィールドは可変文字型として定義します。なお、()内の数字は必要とする最小値の文字数を示しています。
- 「DBS 同時実行制御フィールド」はテーブルの最終フィールドとして定義する必要があります。

4.2 更新の種類

データベースの更新は、項目単位による分類とレコード単位による分類及び、排他制御による分類があり、その組み合わせによって更新方式が決定します。

→項目 による分類	更新			指定更新		
排他制御 レコードの分類	ロックなし	楽観ロック	悲観ロック	ロックなし	楽観ロック	悲観ロック
全データ更新	○	○	○	○	○	○
修正データのみ	○	○	○	—	○	○

項目による分類は、タスク定義（タスクタイプ「201:SQL」）の「タスクタイプ属性」にて指定します。

更新：全項目を更新する場合に指定します。

指定更新：指定した項目のみ（指定する項目にプライマリキーに該当する項目と、「DBS 同時実行制御フィールド」を含みます）を更新する場合に指定します。」

排他制御は、クエリー定義の「排他制御タイプ」にて指定します。

ロックなし：他者と競合することのないテーブルを更新する場合に指定します。

楽観ロック：（標準）データの参照時はデータのロックを行わないで、更新時にロックをかけます。ただし、他者が先に更新した場合は今回更新しません。その場合に指定します。

悲観ロック：データ参照時にロック（の状態に）します。更新時にロック中のデータを更新し、全ロックを解除します。その場合に指定します。

レコードの類は、クエリー定義の「更新処理形態」にて指定します。

全データ更新：シート内のデータを一括して、**追加・更新、または削除**を行う場合に指定します。

修正データのみ：新規のレコード、変更したレコード、削除指定したレコードのみを更新処理する場合に指定します。

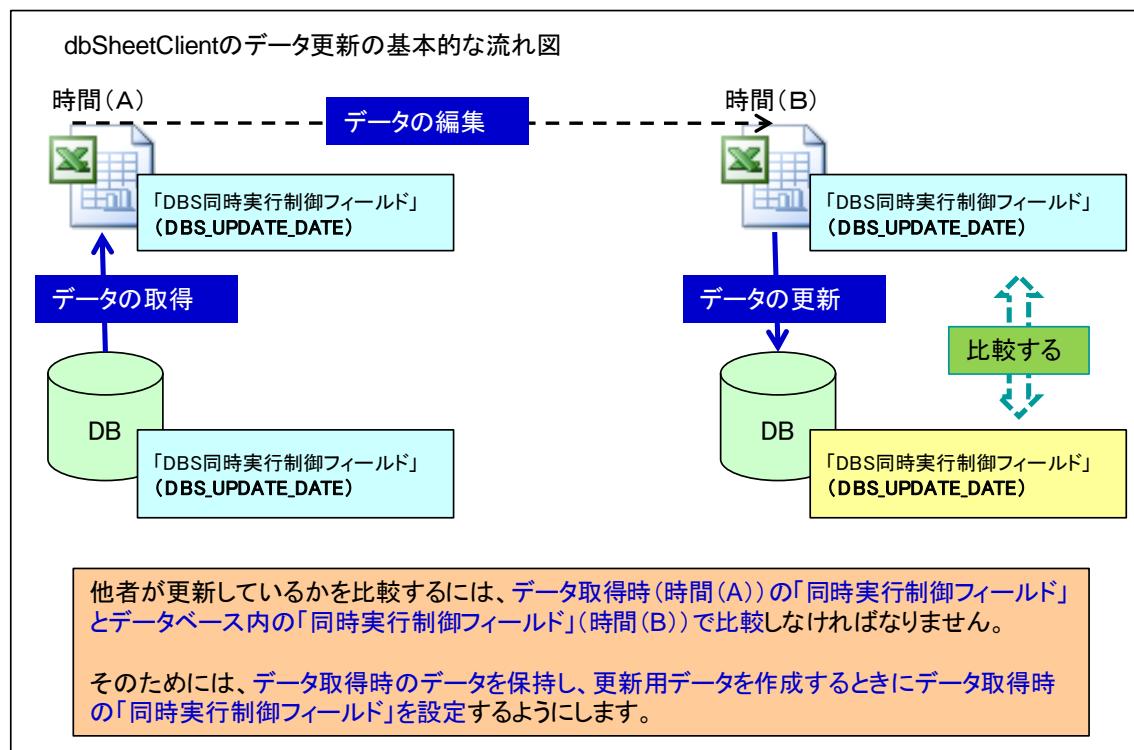
4.3 「DBS 同時実行制御フィールド」を利用したデータ更新処理

データ更新時に、他者が先にデータを更新していないかをチェックしなければなりません。そのためには、更新前のデータと更新用のデータを比較する必要があります。

dbSheetClient では更新チェックに、データベースの「DBS 同時実行制御フィールド (DBS_UPDATE_DATE)」と、Excel シート上の更新データに含まれる「DBS 同時実行制御フィールド (DBS_UPDATE_DATE)」のデータの内容を比較しています。

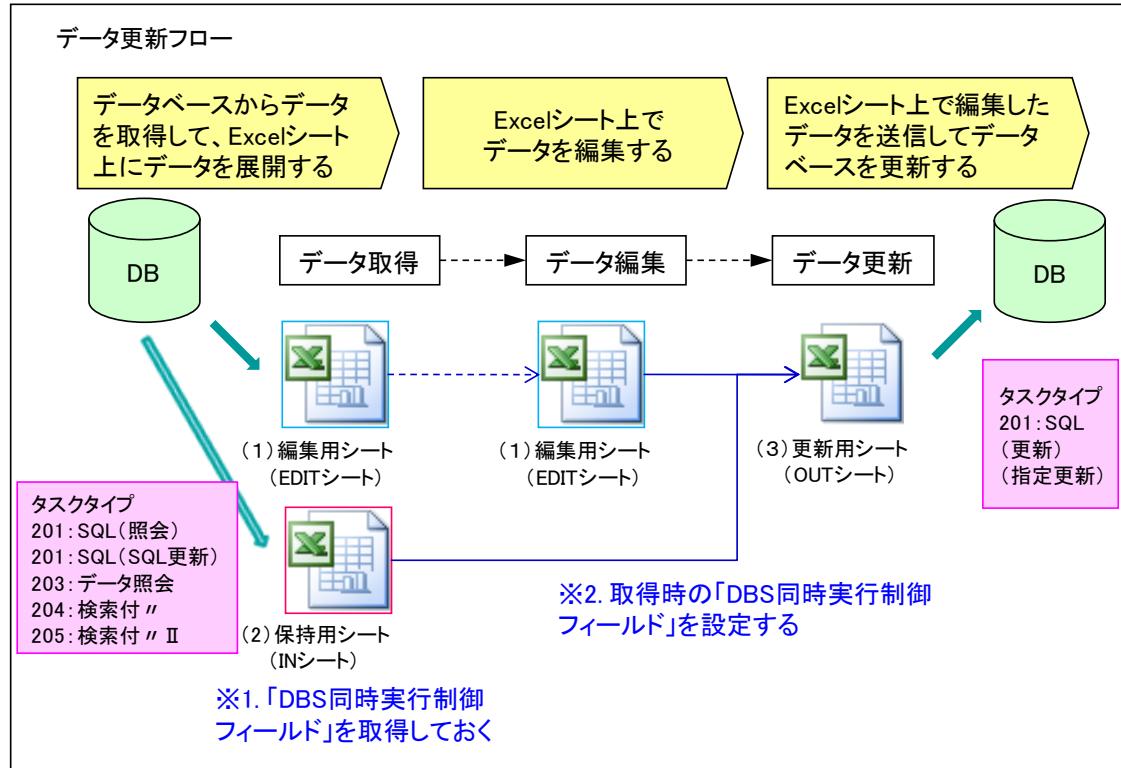
データベースの「DBS 同時実行制御フィールド」が更新していた場合は、他者が先にレコードの更新をしていると判断して、dbSheetClient は更新処理をキャンセルします。

※ 悲観ロックの場合は「DBS_UPDATE_USER」の比較も行います。



4.4 リスト(表)形式の場合のデータ更新方法

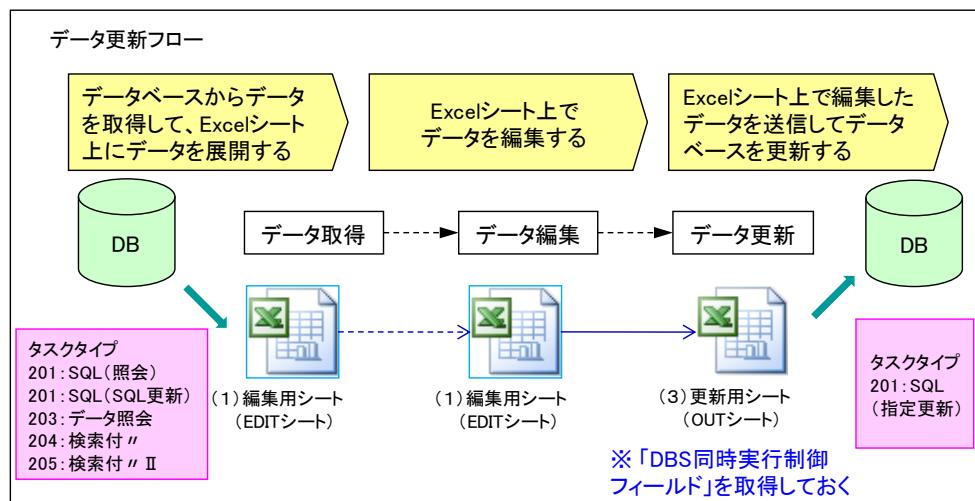
取得した全データではなく、**変更したデータのみを更新する場合**、更新前のデータと**編集したデータを比較**しなければなりません。また、**リスト(表)形式**の場合は、**大量のデータを保持**しなければなりません。そのための方法は、3つのExcelシートを活用することで可能です。



カード(単票)形式の場合やデータ量の少ないリスト(表)形式の場合は、編集用シートのみで更新用データを作成可能です。

第8章 IOTGは、この更新仕様に合わせて、各種定義シートとExcelシートを作成します。

4.5 カード(単票)形式の場合のデータ更新方法



第5章 同時実行制御機能(排他制御)について

同時実行制御機能とは、同時に複数のユーザが同じデータを更新する可能性のあるシステムにおいて、一人のユーザによるデータ更新が他のユーザの更新処理に悪影響を与え、またはデッドロックによるシステム全体が停止することなどを防止するための制御方法です。

dbSheetClient の同時実行制御には、3 つの方式があります。

- ① 楽観的同時実行制御（楽観的ロック方式）
- ② 悲観的同時実行制御（悲観的ロック方式）
- ③ 同時実行制御を行わない

各同時実行制御の説明と定義例などを、各節で説明しています。

5.1 楽観的同時実行制御(楽観的ロック方式)

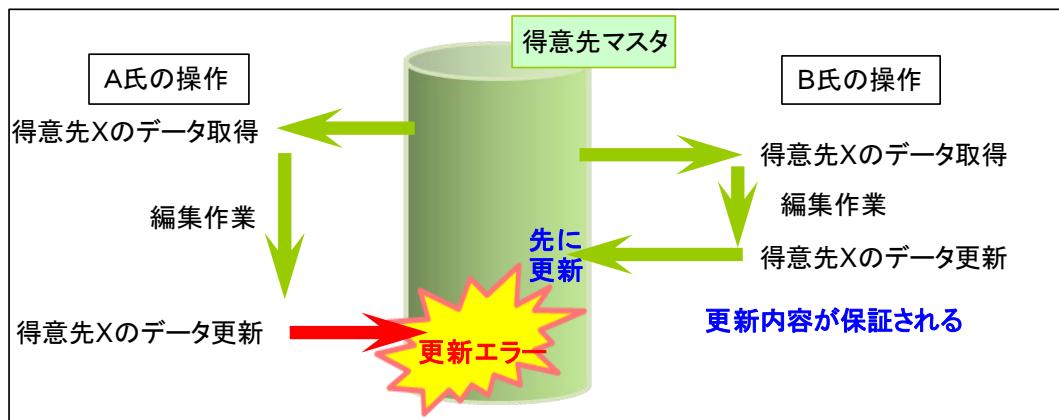
この方式は、ほとんどの Web アプリケーションが採用する排他制御方式です。データを取得するときはデータをロックしません。データを更新するときに、他者がデータを更新していないかチェックし、更新していないときのみ、ロックをかけてデータを更新します。更新終了後にロックを解除します。

【利点】

- ① データの整合性を保証します。
- ② ロック解除待ち状態なる機会や長時間のロック解除待ち状態が大幅に減少します。
- ③ デットロック状態になるリスクを減らせます。

【問題点】

データを取得して編集中に、他者が同じデータを取得して先にデータを更新することができるため、他者が先に更新したときには、自分は更新処理をできなくなります。その場合はデータの再取得と再入力が必要になります。(先勝ち更新)



5.1.1 楽観的同時実行制御のためのデータベースの定義

楽観的同時実行制御を有効にするには、データベースのテーブルの最後に「DBS 同時実行制御フィールド」を定義しておかなければなりません。

なお、「DBS 同時実行制御フィールド」がない場合は、データベースソフトの排他制御機能に依存します。また、「DBS 同時実行制御フィールド」の詳細は、『4.1 DBS 同時実行制御フィールド』を参照してください。

5.1.2 楽観的同時実行制御のための Excel シートの定義

同時実行制御によるデータベースの更新を行うには、更新用の Excel シート内に「DBS 同時実行制御フィールド」とデータ取得時のデータが必要になります。

それは dbSheetClient が、データ更新時に他者が更新したかをデータベースのデータで比較するときに使用するためです。（データ取得時の値を dbSheetClient 側で保持はしません。）そのため、Excel シート内に取得データを保持する必要があります。

「クエリー定義」の「更新処理形態」が“全データ更新”の場合は、取得時の「DBS 同時実行制御フィールド」が必要ですが、“修正データのみ”的場合は、取得時の全データを Excel シート内に保持する必要があります。

それは、Excel シート内の「DBS 同時実行制御フィールド（“DBS_STATUS”）」の値が“2”（追加・更新）、または“99”（削除）がある場合のみ更新処理を行う仕様だからです。

そして、追加や更新の判断は、更新前の全データがなければ、判断することが不可能だからです。そのため、Excel シート内に全取得データを保持する必要があります。

以下は、1 レコードのデータの例です。

① 【照会のみの場合】

《編集用のデータ》				
商品コード	商品名	単位	単価	分類
100	商品100	100	1.000	A

更新しない場合、
「DBS 同時実行制御フィールド」は不要です。

② 【「クエリー定義」の「更新処理形態」が“全データ更新”の場合】

《編集用のデータ》				
商品コード	商品名	単位	単価	分類
100	商品100	100	1.200	B

《取得時のデータ》					
DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER	DBS_UPDATE_DATE	
1	User1	DT2009-09-03 10:12:13.234	User1	DT2009-09-03 10:12:13.234	

《更新用のデータ》								
商品コード	商品名	単位	単価	分類	DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER
100	商品100	100	1.200	B	1	User1	DT2009-09-03 10:12:13.234	User1

“全データ更新”の場合は、「DBS_STATUS」の値を変更する必要がありません。更新用にあるデータを全て更新するようになります。

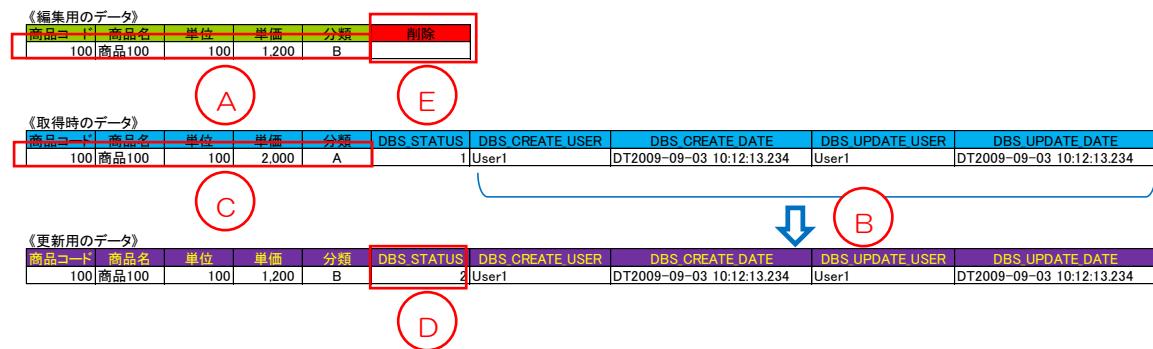


取得時に「同時実行制御フィールド」のデータを、更新用データ側に展開することができます。



編集用のデータを更新用のデータになるように、代入式（数式）を Excel シートに設定します。

③ 【クエリー定義】の「更新処理形態」が“修正データのみ”の場合



“修正データのみ”的場合は、「DBS_STATUS」の値によって、更新や削除の対象データかを判断します。取得時のデータ **C** と編集用のデータ **A** を比較するために、それぞれをセルの左から順番に連結した値で比較するようにします。

(例) 編集用のデータ(Z2) := A2&" :"&B2&" :"&C2&" :"&D2&" :"&E2

取得時のデータ(Z5) := A5&" :"&B5&" :"&C5&" :"&D5&" :"&E5

下記の通り、削除フラグや結合したデータの比較などにより、DBS_STATUS の値を決定します。

D “DBS_STATUS” (F10) := IF(F2="99",99,IF(AND(Z2 = Z5,A2=""),2,IF(AND(Z2=Z5,A2=""),1,"")))

“DBS_STATUS” の意味

“2”：追加・更新対象のデータ

“1”：追加・更新・削除対象外のデータ

“”：追加・更新・削除対象外のデータ（空行）

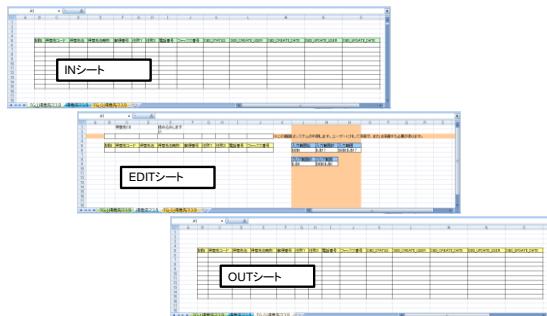
“99”：削除対象のデータ

E に何か設定してある場合、**D** を削除対象（99）にします。

取得時に「同時実行制御フィールド」のデータを、更新用データ側に展開することが可能です。

編集用のデータを更新用のデータになるように、代入式（数式）を Excel シートに設定します。

複数レコードの場合も、基本的な考え方は同じです。ただし、レコード数が多くなると、1枚の Excel シート内では扱いにくくなります。そのため、各データをシート毎に作成するようにします。



取得時のデータは、「IN シート」に展開します。
編集用のデータは、「EDIT シート」です。

更新用のデータは、「OUT シート」になります。

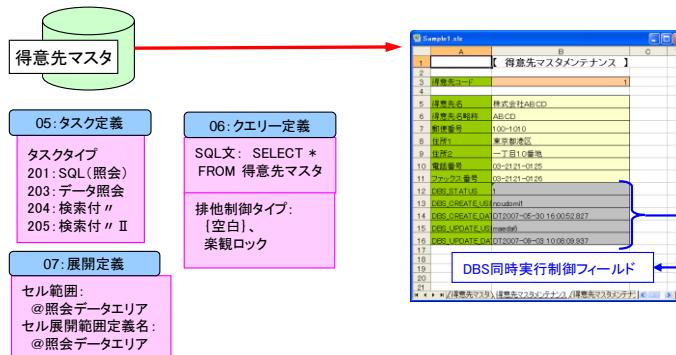
項目のセル位置や並び順は、同じ方がデバッグやメンテナンスしやすいです。

また、項目タイトルの色をサンプルのように変更することをお勧めします。

5.1.3 楽観的同時実行制御のための定義シートの定義例

データベースのデータの更新は、『4.4 リスト（表）形式の場合のデータ更新方法』の図の通り、データベースのデータを取得して Excel シートに展開する場合は、照会用の SQL を発行する定義シートの設定を行います。

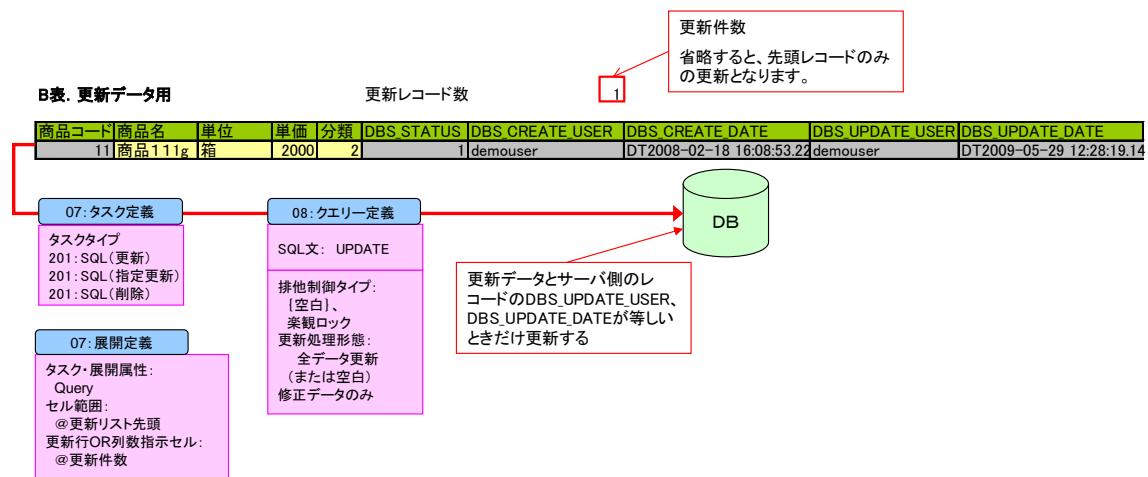
【データ取得用の定義シートの設定例】



クエリー定義の「排他制御タイプ」: 空白または、楽観ロックを指定します。

更新や削除の場合、更新用シートに更新前の「DBS 同時実行制御フィールド」の値が、設定されている必要があります。

【データ更新用の定義シートの設定】



クエリー定義
SQL 文：“UPDATE” 固定です。

排他制御タイプ：“空白または“楽観ロック””を選択します。

更新処理形態：“全データ更新または空白か修正データのみ“を選択します。

展開定義

セル範囲：(必須) 更新のためのデータが保存されているシートの位置を指定します。

更新行 OR 列数指示セル：更新のためのデータレコード数を指定します。

5.2

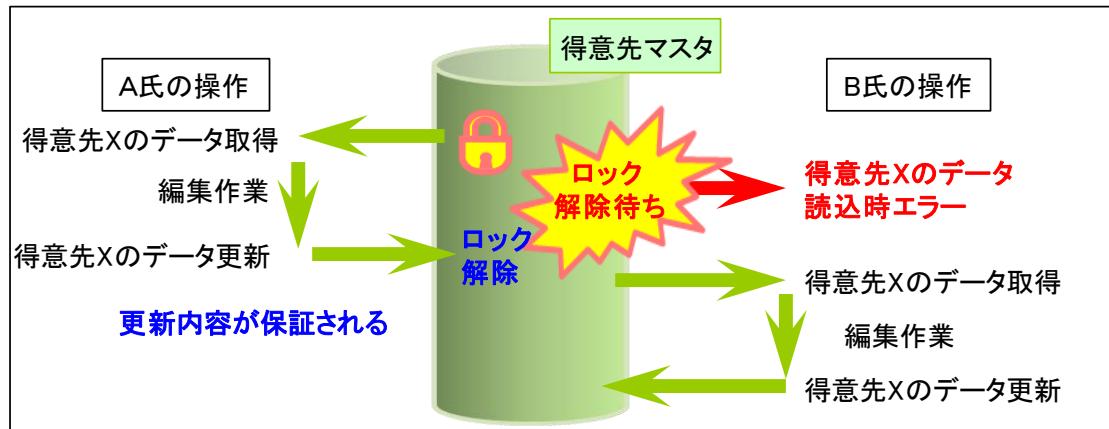
悲観的同時実行制御(悲観的ロック方式)

データを取得したときから更新が完了するまでロックをかけます。

データの編集から更新完了まで、他のユーザはデータへのアクセスが制限されます。(ロックによる排他制御) また、他者が更新操作を行っていることが分かります。

【利 点】 データ取得後は、データ更新を確実におこなうことができます。

【問題点】 長時間のロック解除待ち状態が続くことや、デットロック状態になることがあります。



Web アプリケーションでは、インターネット経由でサーバからデータをクライアントPCに取得し、データを編集し、データ更新をインターネット経由で依頼します。データを取得したあと、何らかの原因で、ネットワーク回線が切断して更新が完了できなかったり、データを読み込んだまま、長時間更新処理を完了しなかったりするケースが発生します。

このように「悲観的同時実行制御方式」を利用すると、長時間にわたってロック状態が続くため、他のユーザがデータにアクセスすることができなくなる場合がありますので、使用には十分な注意が必要です。

dbSheetClient は、悲観ロックをかけたユーザが、再度データベースの照会を行ったときに、ロックを解除し、再度ロックをするようにします。これは、データの抽出条件の変更などにより、ロック解除漏れを防止するために行います。

また、ユーザがプロジェクトを終了したときに、悲観ロックをかけたユーザのすべてのロックを解除します。

5.2.1 悲観的同時実行制御のしくみ

悲観的ロック方式によるデータの更新では、データ取得時（201.SQL の照会）に、クエリー定義の排他制御タイプで「悲観ロック」を設定して、データ取得しておく必要があります。

「悲観ロック」でデータを取得すると、サーバ側の取得したレコードの DBS_STATUS が全て “3：悲観ロック中” に変更されて、レコードロックが掛かります。

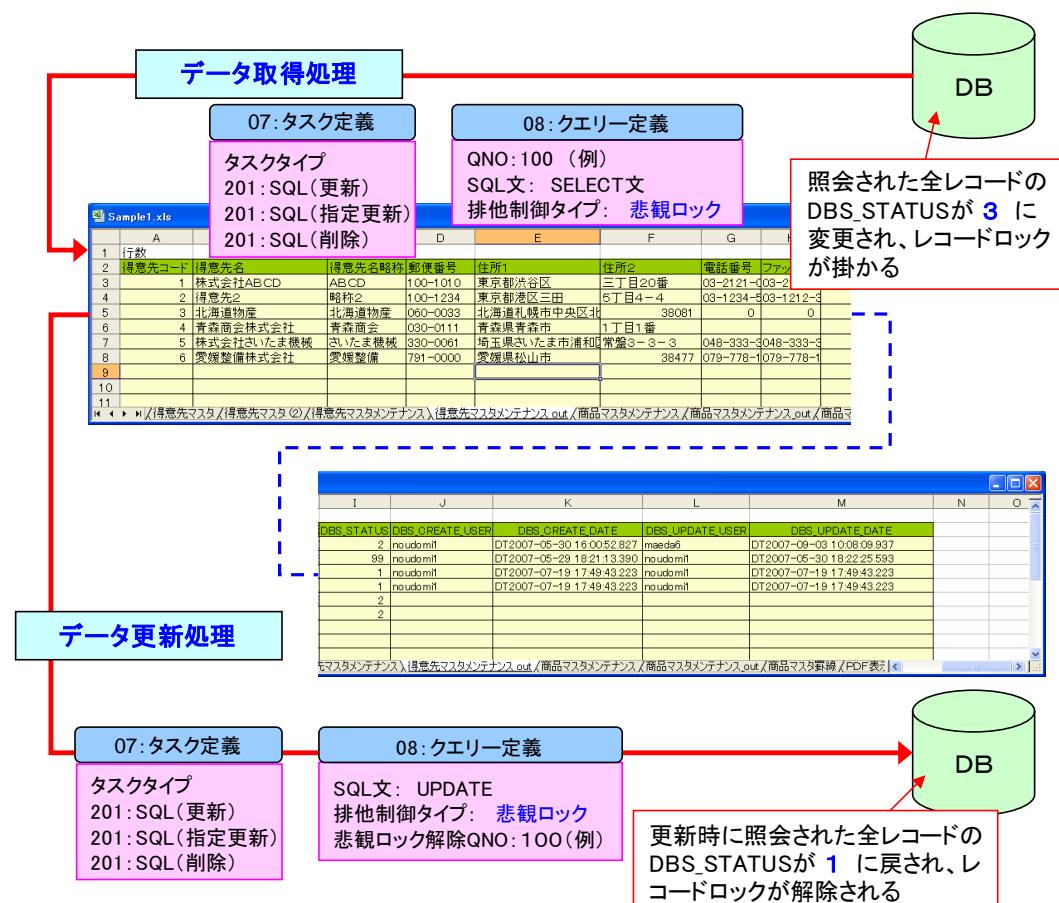
サーバ側のデータの DBS_STATUS が “3：悲観ロック中” のレコードについては、自分以外の他のユーザは参照ができますが、更新はできなくなります。

また、データを Excel シート上で変更して、「全データ更新」や「修正データのみ」でサーバのデータを更新し、悲観ロックを解除するには、必ず、更新用クエリー定義には、「排他制御タイプ」に「悲観ロック」を設定し、「悲観ロック解除 QNO」に、「悲観ロック」でデータを取得したときのクエリーノ（QNO）を指定するようにしてください。

※1. この悲観ロック解除の設定をおこなうと、更新時に全てのレコードの DBS_STATUS の値を “1：更新可能” に戻すことができますが、設定をしないで更新をおこなうと、DBS_STATUS の値は “3：悲観ロック中” のままの状態を保持します。

※2. 更新処理ではなく、201.SQL（照会）でデータ取得時と同じSQL文で「排他制御タイプ」を「悲観ロック解除」に設定することで、悲観ロック解除をおこなうこともできます。

※3. DBS_STATUS の値を “1：更新可能” ⇔ “3：悲観ロック中” したとき、DBS_UPDATE_USER と DBS_UPDATE_DATE の値は更新されますので、**対象データは全データ更新**となります。



5.2.2 悲観的同時実行制御のためのデータベースの定義

『5.1.1 楽観的同時実行制御のためのデータベースの定義』と同じく、悲観的同時実行制御を有効にするには、データベースのテーブルの最後に「DBS 同時実行制御フィールド」を定義しておかなければなりません。

5.2.3 悲観的同時実行制御のための Excel の定義

楽観的同時実行制御と同じく、更新用の Excel シート内に「DBS 同時実行制御フィールド」とデータ取得時のデータが必要になります。

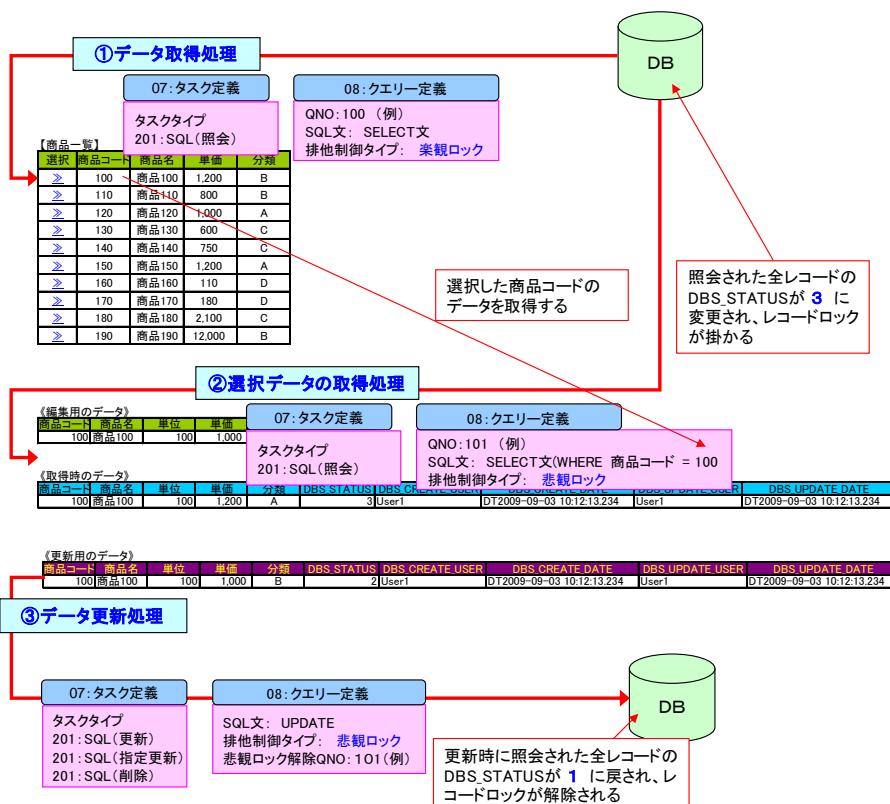
詳しくは、『5.1.2 楽観的同時実行制御のための Excel シートの定義』を参照してください。

5.2.4 悲観的同時実行制御のための定義シートの定義例

悲観的同時実行制御は、データ取得（照会）時にロックをかけますので、大量のデータをかけることで、ロック解除待ちやデッドロックを発生しやすくなります。

そのために、データの修正・更新は、「レコード単位」または「伝票単位」など、最小単位で行うことをお勧めします。（親子構成のテーブルを更新する場合は、デッドロック防止のため、親テーブルからロックをかけることを推奨します。）

削除がない場合は、「クエリー定義」の「更新処理形態」を“全データ更新”とします。削除のある場合は、“修正データのみ”とします。ただし悲観ロック中の更新は、ロック中の全レコードの「DBS_STATUS」を更新しますので、実際は全データ更新をしていることになります。



5.2.5 悲観的同時実行制御の仕様上の注意

dbSheetClient が提供する悲観的同時実行制御は、データ編集作業中のレコードを通知することを目的としています。そのため、「DBS_STATUS」による制御を行っています。

- ① ロックできるのは、単一テーブルのみ

SQL 文は複数のテーブルを扱うことが可能ですが、1 つのクエリー定義でロックできるテーブルは、単一テーブルのみです。

※ 単一テーブル とは、次の状態を意味します。

- JOIN を使用しない
- ORDER BY のテーブルの項目等を定義しない

例)

親子関係の複数テーブルをロックする場合は、親に該当するテーブルのみをロックするようにします。

更新時に、ロック解除するようにします。

【データ取得】

タスク定義	クエリー定義
201:SQL(照会)	SQL 文 テーブル(親) 排他制御タイプ: 悲観ロック QNO:102
201:SQL(照会)	SQL 文 テーブル(子) 排他制御タイプ: 楽観ロック(または空白) QNO:103

【データ更新】

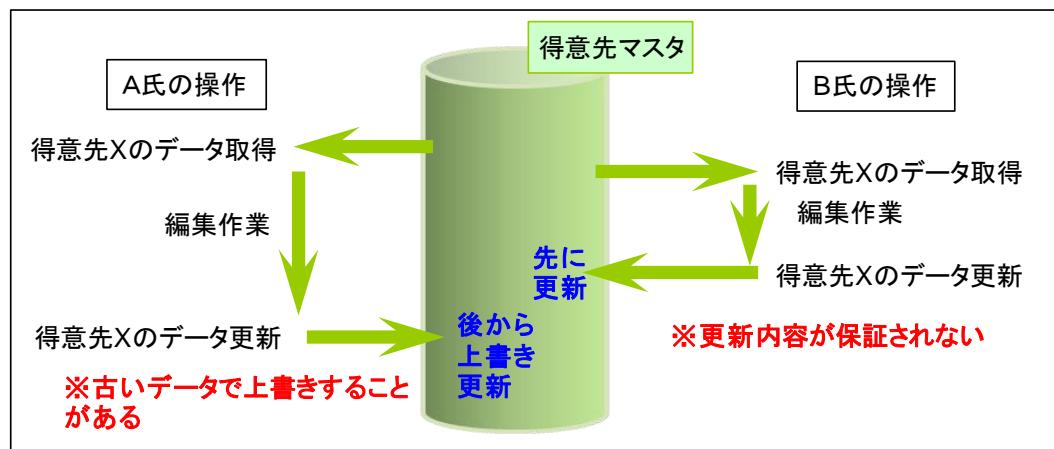
タスク定義	クエリー定義
201:SQL(更新)	SQL 文 テーブル(親) 排他制御タイプ: 悲観ロック解除 QNO:104 悲観ロック解除 QNO:102
201:SQL(更新)	SQL 文 テーブル(子) 排他制御タイプ: 楽観ロック(または空白) QNO:105

5.3 同時実行制御を行わない

dbSheetClient による排他制御を行わない方式です。この場合は、**最新操作の優先（後勝ち更新）**になります。つまり、最後にデータを更新したユーザの更新が常に優先されます。

【利点】 他の誰からもデータの同時実行処理がないことが保証されている場合には実装が簡単です。

【問題点】 共有するデータベースの場合、先に更新したデータの後に上書きすることになりますので、他のユーザの更新データと自分の更新データが混在する可能性があります。そのため更新内容が保証できなくなります。



※ 同時実行制御法式についての記載内容では、マイクロソフト社の MSDN ホームページの「[ADO.NET におけるデータ同時実行制御の概要](#)」に記載されている内容を一部引用しています。

5.3.1 同時実行制御を行わない場合のデータベースの定義

ユーザ毎にデータを編集・更新するような、例えば作業用テーブルなど、一次的に使用するテーブルを更新する場合に向いています。

データベース(テーブル)のフィールド定義					
User ID	NO	商品コード	商品名	単価	分類

ユーザ毎にユニークになるようなフィールド定義をし、**プライマリキー**として登録する

User ID	商品コード

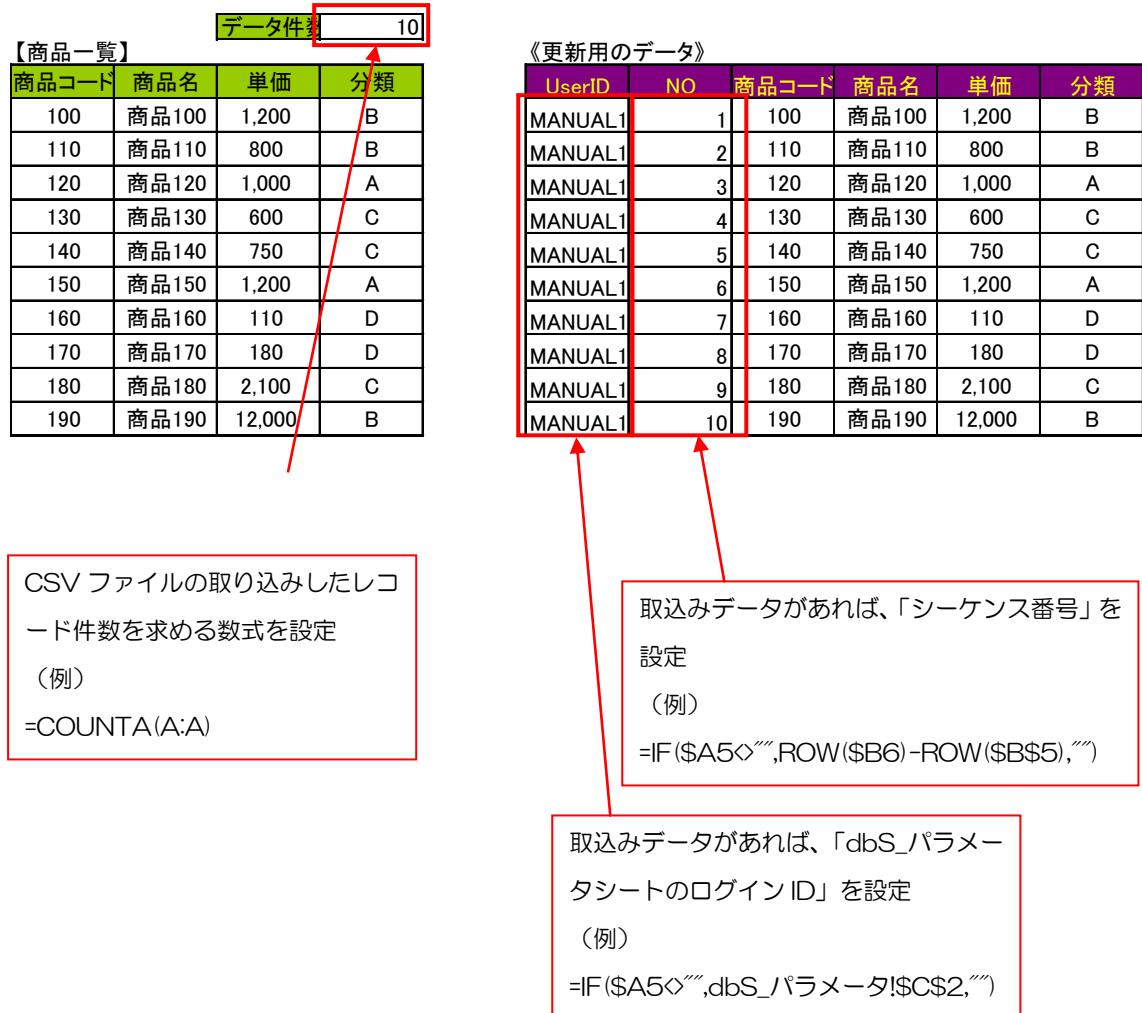
としてもよい

商品マスターの定義と同じ

※「DBS 同時実行制御フィールド」は付加しません。

5.3.2 同時実行制御を行わない場合の Excel の定義

例えば、CSV 入力した商品マスターのデータを登録する前にデータのチェックをするような場合、CSV データの展開用のシートと更新用のシートを作成します。



5.3.3 同時実行制御を行わない場合の定義シートの定義例

タスク処理は、

① CSV ファイルのインポート (一括入出力 (CSV 入力))

② データ更新処理 (SQL (更新))

③ データチェック (SQL (照会))

※ データチェックは、データの整合性や二重登録などのチェックをし、その結果を返すような SQL 文を作成します。



5.4 データ更新のまとめ

5.4.1 更新処理の種類と設定方法

更新処理として、dbSheetClient では「更新」、「指定更新」、「削除」、「SQL 更新」の方法を提供しています。また、楽観的ロック方式と悲観的ロック方式による更新が可能ですが、楽観的ロック方式と悲観的ロック方式については、サーバからデータを取得するときの設定に依存します。

下記の表に楽観的ロック方式と悲観的ロック方式の更新をおこなうためのデータ照会の設定方法と、各種の更新処理の特徴や注意事項をまとめました。

処理形態		タスク定義の主要な定義事項					クエリーリスト定義の主要な定義事項					展開定義の主要な定義事項			
処理	No	タスクタイプ	タスク属性	QNO	DNO	QNO	SQL文	排他制御タイプ	更新処理形態	悲観ロック解除QNO	DNO	シート名	セル範囲	更新行OR列数指示セル	
データ取得 (更新のためのデータを事前に取得する)	①	201.SQL	照会	Q①	D①	Q①	SELECT文を記述する	楽観ロック	---	---	D①	データ取得用のシート名を指定	データを展開する開始セルを指定	---	
	②	"	"	Q②	D②	Q②	"	悲観ロック	---	---	D②	"	"	---	
	③	"	"	Q③	D③	Q③	"	悲観ロック解除	---	---	D③	"	"	---	
	④	"	SQL更新	Q④	D④	Q④	①照会のためのSQL文をSELECT文以外の句を用いて、詳細に記述する。 ②ストアドプロシージャを実行するSQL文を記述する。	---	---	---	D④	"	"	---	
注意事項		1) ①のように、排他制御タイプを「楽観ロック」または省略した時には、SELECT文で取得したサーバ側データのDBS_STATUS(同時実行制御)の値は 1 または 3 のままでです。 2) ②のように、排他制御タイプを「悲観ロック」にした時には、SELECT文で取得したサーバ側データのDBS_STATUS(同時実行制御)の値は 3 に変更され、レコードロックが掛かります。 3) ③のように、排他制御タイプを「悲観ロック解除」にした時には、②で「悲観ロック」をかけたときと同じSELECT文を実行する必要があります。このとき、SELECT文で取得したサーバ側データのDBS_STATUS(同時実行制御)の値は 1 に変更され、レコードロックが解除されます。 4) サーバ側のデータのDBS_STATUS(同時実行制御)の値が 3 のレコードについては、他のユーザは照会はできません。 5) DBS_STATUS(同時実行制御)の値が 3 に変更されたサーバ側のレコードについては、悲観ロックを掛けたユーザが悲観ロック解除をおこなうと 1 に戻り、レコードロックが解除されます。 6) 悲観ロックが掛けられたまま放置されたレコードについては、悲観ロックが掛けられた後、 3時間以上 経過したあとで、誰かがデータを照会したときに強制的に 1 に戻し、レコードロックを解除します。 7) ④のように、タスクタイプ属性に「SQL更新」を用いると、SQL文にストアドプロシージャを実行する文などを記述することによって、データを照会することができます。													

処理		No	タスクタイプ	タスク属性	QNO	DNO	QNO	SQL文	排他制御タイプ	更新処理形態	悲観ロック解除QNO	DNO	シート名	セル範囲	更新行OR列数指示セル	
更新	⑤	201.SQL	更新	Q⑤	D⑤	Q⑤	UPDATE	(※悲観ロック)	全データ更新	(※Q②)	D⑤	データ更新用のシート名を指定	更新データの開始セルを指定	更新処理対象のレコード件数がセットされたセルを指定する		
	⑥	"	"	Q⑥	D⑥	Q⑥	UPDATE	(※悲観ロック)	修正データのみ	(※Q②)	D⑥	"	"	"		
指定更新	注意事項	1) タスクタイプ属性が「更新」の時には、データ更新用シート上の更新データは、サーバ側のテーブルの全フィールドとデータタイプで展開されている必要があります。 2) 更新シート上の更新対象となるレコードは、セル範囲で指定された開始位置から、更新行OR列数指示セルで指定されたレコード件数で囲まれた範囲のレコードとなります。したがって、この範囲の途中に空白レコードがあっても、最期のレコードまで更新できるように、レコード件数を調整しておく必要があります。 3) ⑤のように、更新処理形態を「全データ更新」にすると、データ更新用シート上のDBS_STATUSの値がどのような値であっても、更新行OR列数指示セルで指定されたレコード件数を全てサーバに送信して、全データ更新(サーバにデータがないときには新規追加)します。 4) ⑥のように、更新処理形態を「修正データのみ」にすると、データ更新用シート上のDBS_STATUSの値が 2 のレコードを更新または追加レコードとして、 99 のレコードを削除レコードとして、サーバに送信し、サーバのデータを更新します。サーバに送信されるごとに、DBS_STATUSの値は 1 にセッタされます。 5) 更新時には、重複キー違反や同時実行制御の楽観ロック、悲観ロックの違反により、更新されないことがあります。 6) データ照会時に②のように「悲観ロック」でデータを取得したときには、必ず、「排他制御タイプ」に「悲観ロック」を設定し、さらに「悲観ロック解除QNO」に悲観ロックを掛けたQNOを指定して更新してください。悲観ロック解除QNOを忘れるごとに、レコードロックが解除されないレコードが残ることになります。 7) データが更新されると、同時実行制御フィールドの、DBS_UPDATE_USER には更新したユーザ名が、DBS_UPDATE_DATE には更新年月日時刻がセッタされます。 8) サーバのDBS_STATUSの値は、 1 または 3 のみで、 2 や 99 に変化することはできません。														
		⑦	201.SQL	指定更新	Q⑦	D⑦	Q⑦	UPDATE	(※悲観ロック)	全データ更新	(※Q②)	D⑦	同上	更新データの開始セルを指定	更新処理対象のレコード件数がセットされたセルを指定する	
	注意事項	1) タスクタイプ属性が「指定更新」の時には、データ更新用シート上の更新データは、サーバ側のテーブルのフィールドのうち、 主キー と、 変更したいフィールド と、5つの同時実行制御、DBS_STATUS、DBS_CREATE_USER、DBS_CREATE_DATE、DBS_UPDATE_USER、DBS_UPDATE_DATE のフィールドが展開されているだけで、更新をおこなうことができます。 2) 更新の欄で記述した、2)~8) は指定更新でも同じように適用されます。														

処理	No	タスクタイプ	タスクタイプ属性	QNO	DNO	QNO	SQL文	排他制御タイプ	更新処理形態	悲親ロック解除QNO	DNO	シート名	セル範囲	更新行OR列数指示セル
削除	⑨	201.SQL	削除	Q⑨	D⑨	Q⑨	UPDATE	(※悲親ロック)	全データ更新	(※Q②)	D⑨	同上	削除データの開始セルを指定	削除処理対象のレコード件数がセットされたセルを指定する
	⑩	"	"	Q⑩	D⑩	Q⑩	UPDATE	(※悲親ロック)	修正データのみ	(※Q②)	D⑩	"	"	削除処理対象のレコード件数がセットされたセルを指定する
更新処理	注意事項	1) タスクタイプ属性が「削除」の時には、データ更新用シート上の更新データは、サーバ側のテーブルの全フィールドと同じフィールド数とデータタイプで展開されている必要があります。 2) 更新シート上の削除対象となるレコードは、セル範囲で指定された開始位置から、更新行OR列数指示セルで指定されたレコード件数分で囲まれた範囲のレコードとなります。したがって、この範囲の途中に空白レコードがあっても、最期のレコードまで削除できるように、レコード件数を調整しておく必要があります。 3) ⑨のように、更新処理形態を「全データ更新」にすると、データ更新用シート上のDBS_STATUSの値がどのような値であっても、更新行OR列数指示セルで指定されたレコード件数分を全てサーバに送信して、全データを削除（サーバにデータがないときにはエラー）します。 4) ⑩のように、更新処理形態を「修正データのみ」にすると、データ更新用シート上のDBS_STATUSの値が 2 のレコードを更新または追加レコードとして、99 のレコードを削除レコードとして、サーバに送信し、サーバのデータを更新または削除します。サーバに送信されると、サーバ内の、DBS_STATUSの値は 1 にセットされます。 5) 更新時には、重複キー違反や同時実行制御の楽親ロック、悲親ロックの違反により、削除されないことがあります。 6) データ照会時に②のように「悲親ロック」でデータを取得したときには、必ず、「排他制御タイプ」に「悲親ロック」を設定し、さらに「悲親ロック解除QNO」に悲親ロックを掛けたQNOを指定して更新してください。悲親ロック解除QNOを忘ると、レコードロックが解除されないレコードが残ることになります。 7) 削除されたレコードは物理的に削除されます。 8) サーバのDBS_STATUSの値は、1 または 3 のみで、2 や 99 に変化することはありません。												
		⑪	201.SQL	SQL更新	Q⑪	D⑪	Q⑪	①更新のためのSQL文をINSERT、DELETE、UPDATEを用いて詳細に記述する。 ②ストアドロジジャを実行するSQL文を記述する。	---	---	---	D⑪	---	---
SQL更新	注意	1) ⑪のように、タスクタイプ属性に「SQL更新」を用いると、更新のためのSQL文をINSERT、DELETE、UPDATEを用いて詳細に記述したり、ストアドロジジャを実行するSQL文などを記述することによって、独自にデータを更新することができます。												

5.4.2 タスクタイプ属性が「更新」のとき（楽親ロック）

a) 全データ更新

- ① 空白行も含めて更新処理レコード件数は12とする
- ② サーバ側のテーブルの全フィールドを同じ名称で設定する。
- ③ セル範囲（更新データの開始位置）

③	更新処理件数:	12	①	②

商品コード	商品名	単位	単価	分類	DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER	DBS_UPDATE_DATE
1商品1	個	1250	55	1	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-20 17:23:11.35	
2商品2	箱	1300	10	1	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-20 17:25:21.71	
3商品3	箱	1400	11	1	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-08 17:00:32.84	
4商品4	箱	1650	12	1	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-08 13:52:44.95	
5商品5	箱	1750	20	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
6商品6	箱	2000	21	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
7商品7	箱	2000	22	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
8商品8	箱	1260	23	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
9商品9	箱	2000	25	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
10商品10	箱	1280	30	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	

b) 修正データのみ

- ① 空白行も含めて更新処理レコード件数は12とする
- ② サーバ側のテーブルの全フィールドを同じ名称で設定する。
- ③ セル範囲（更新データの開始位置）
- ④ DBS_STATUS の値が2のレコードが更新、99のレコードが削除される

③	更新処理件数:	12	①	②

商品コード	商品名	単位	単価	分類	DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER	DBS_UPDATE_DATE
1商品1	個	1250	55	1	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-20 17:23:11.35	
2商品2	箱	1300	10	2	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-20 17:25:21.71	
3商品3	箱	1400	11	2	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-08 17:00:32.84	
4商品4	箱	1650	12	1	manual1	DT2007-05-30 16:00:52.82	manual1	DT2008-07-08 13:52:44.95	
5商品5	箱	1750	20	99	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
6商品6	箱	2000	21	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
7商品7	箱	2000	22	99	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
8商品8	箱	1260	23	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
9商品9	箱	2000	25	2	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	
10商品10	箱	1280	30	1	manual1	DT2007-05-30 16:00:52.83	manual1	DT2008-07-08 13:52:44.95	

5.4.3 タスクタイプ属性が「指定更新」のとき（楽観ロック）

a) 全データ更新

- ① 空白行も含めて更新処理レコード件数は12とする
- ② サーバ側のテーブルのフィールドのうち【主キー】・【変更したいフィールド】・【5つの同時実行制御フィールド】のフィールドのみの展開で更新できる。
- ③ セル範囲（更新データの開始位置）

更新処理件数: **12** ①

②

商品コード	商品名	単価	DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER	DBS_UPDATE_DATE
1	商品1	1250	1	manual1	DT2007-05-30 16:00:52.827	manual1	DT2008-07-20 17:23:11.35
2	商品2	1300	1	manual1	DT2007-05-30 16:00:52.827	manual1	DT2008-07-20 17:25:21.71
3	商品3	1400	1	manual1	DT2007-05-30 16:00:52.828	manual1	DT2008-07-08 17:00:32.84
4	商品4	1650	1	manual1	DT2007-05-30 16:00:52.829	manual1	DT2008-07-08 13:52:44.95
5	商品5	1750	1	manual1	DT2007-05-30 16:00:52.830	manual1	DT2008-07-08 13:52:44.95
6	商品6	2000	1	manual1	DT2007-05-30 16:00:52.831	manual1	DT2008-07-08 13:52:44.95
7	商品7	2000	1	manual1	DT2007-05-30 16:00:52.832	manual1	DT2008-07-08 13:52:44.95
8	商品8	1260	1	manual1	DT2007-05-30 16:00:52.833	manual1	DT2008-07-08 13:52:44.95
9	商品9	2000	1	manual1	DT2007-05-30 16:00:52.834	manual1	DT2008-07-08 13:52:44.95
10	商品10	1280	1	manual1	DT2007-05-30 16:00:52.835	manual1	DT2008-07-08 13:52:44.95

b) 修正データのみ

- ① 空白行も含めて更新処理レコード件数は12とする
- ② サーバ側のテーブルのフィールドのうち【主キー】・【変更したいフィールド】・【5つの同時実行制御フィールド】のフィールドのみの展開で更新できる。
- ③ セル範囲（更新データの開始位置）
- ④ DBS_STATUS の値が2のレコードが更新、99のレコードが削除される

更新処理件数: **12** ①

②

④

商品コード	商品名	単価	DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER	DBS_UPDATE_DATE
1	商品1	1250	1	manual1	DT2007-05-30 16:00:52.827	manual1	DT2008-07-20 17:23:11.35
2	商品2	1300	2	manual1	DT2007-05-30 16:00:52.827	manual1	DT2008-07-20 17:25:21.71
3	商品3	1400	2	manual1	DT2007-05-30 16:00:52.828	manual1	DT2008-07-08 17:00:32.84
4	商品4	1650	1	manual1	DT2007-05-30 16:00:52.829	manual1	DT2008-07-08 13:52:44.95
5	商品5	1750	99	manual1	DT2007-05-30 16:00:52.830	manual1	DT2008-07-08 13:52:44.95
6	商品6	2000	1	manual1	DT2007-05-30 16:00:52.831	manual1	DT2008-07-08 13:52:44.95
7	商品7	2000	99	manual1	DT2007-05-30 16:00:52.832	manual1	DT2008-07-08 13:52:44.95
8	商品8	1260	1	manual1	DT2007-05-30 16:00:52.833	manual1	DT2008-07-08 13:52:44.95
9	商品9	2000	2	manual1	DT2007-05-30 16:00:52.834	manual1	DT2008-07-08 13:52:44.95
10	商品10	1280	1	manual1	DT2007-05-30 16:00:52.835	manual1	DT2008-07-08 13:52:44.95

5.5 更新処理の注意事項

5.5.1 Null 値を取り扱うときの注意事項

1) DBのNull値をExcelシート上に展開したときの問題点

DB側のNull値のデータをExcelシートに展開すると、空文字“”になって、Excelシート上では空白状態になります。
その結果、Excelシート上にNull値が空文字“”として展開された後では、DB側のデータが、空文字だったのかNull値だったのかを、Excelシート上だけでは、判断することができなくなります。
そのため、dbSheetClient では以下の仕様で、Null値 のデータの取得やNull値での更新をおこなうことができるようにしています。

2) Excelシート上のデータを DB に対して、Null値 で 更新(登録)するための仕様

《V2および、V3. 0. 3. * までのバージョン》

数値型: Excelシート上の空文字“”は 0 で更新(登録)される。
日付型: Excelシート上の空文字“”は 1899.12.30 で更新(登録)される。
文字型: Excelシート上の空文字“”は “” で更新(登録)される。

《V3. 0. 4. 0 以降のバージョン》

【1】クエリー定義の定義方法

クエリー定義で、クエリータイプが「更新」のときに「日付時刻NULL指定」のフィールドに、「NULL指定」を設定する。
※「NULL指定」が設定されていないときには、V2の仕様となるので注意してください。

【2】Excelシート上のデータの設定方法

数値型: Excelシート上の空文字“”または“Null”は Null値 で登録する。 → 空文字“”での登録は不可。
日付型: Excelシート上の空文字“”または“Null”は Null値 で登録する。 → 空文字“”での登録は不可。
“” 1899.12.30 0:00:0時は Null値 で登録する。 → 空文字“”の登録は不可。
文字型: Excelシート上の空文字“”は 空文字“”で登録する。また、“Null”の文字は Null値 で登録する。

【3】文字型のときの注意点(制限事項)

Excelシート上の空文字“”は、空文字“”としてDBに登録し、“Null”が入力されているセルについては、Null値 で登録します。
そのため、DB側から、Null値 のデータを持ってくるときには、SQLの関数 isnull(項目名, "null")などをを利用して、“Null”に変換してExcelシート上に展開したり、Excelシート上の更新シートで空文字“”の時には、“Null”に変換するといった式を設定する必要があります。

したがって、変換しないまま、Excelシート上に展開すると、空文字“”として展開されるため、そのままDB側に登録すると、以下のように、DB側のデータは 空文字“”として登録されてしまいます。

DB側 → Excelシート上 → DB側
[Null値] → “” → “” のように 空文字“”になる。

図で示すと以下のようになります。

データの型 処理	日付時間			文字型(String)			数値型(Numeric)			データの 場所
DBMS	Null	0		Null	””	0	Null	0		DB上
dbSheet	””	0	Null	””	””	0	Null	””	0	Null
										Excel上
DBMS	Null			””		0	Null	Null	0	Null

【注意点】

※1. 文字型(String)の時に、DBからdbSheetClientで展開する時に””になるので、Outシート上でNullに変換するか、SQLのSelect文でNullの時、“Null”をdbSheetClient上のExcelシート上に展開するように設定する必要があります。

※2. Null値の対応のほか、DBのデータ型に対応した、Excelシート上のセル書式の設定については次ページでさらに詳しく説明しています。

5.5.2 更新処理時のデータ型とセル書式設定の注意事項

①DB側のデータタイプとExcelシート上のセル書式設定に関する注意点

DB側では文字型、数値型、日付型、時間型などのデータタイプが設定でき、DB側のデータタイプに応じて、データのREAD/WRITEがおこなわれますが、Excel側では、セルに設定した書式をもとにデータタイプが判断されます。しかしながら、セルの書式を設定しないで、標準のままにしているときや、=数式で他のセルを参照しているときなどには、以下のようにデータタイプが自動的に判定されて、データタイプが変化するので注意が必要です。

①単一セルの場合:

入力されているデータが数字の場合には数字、文字の場合には文字として判定される。

②複数行にわたる連続したセル(列方向)の場合:

入力されているデータの過半数が数字の場合には数字、過半数が文字の場合には文字と判定される

③=数式で参照したときに標準設定のセルに表示される参照値:

1) 参照した値が数字の場合: 数字

2) 参照した値が文字列書式のセルに入力されている数字: 文字型の数字

3) 参照した値が文字: 文字

4) 参照した値が日付: 数値(1900/1/1からの日数)

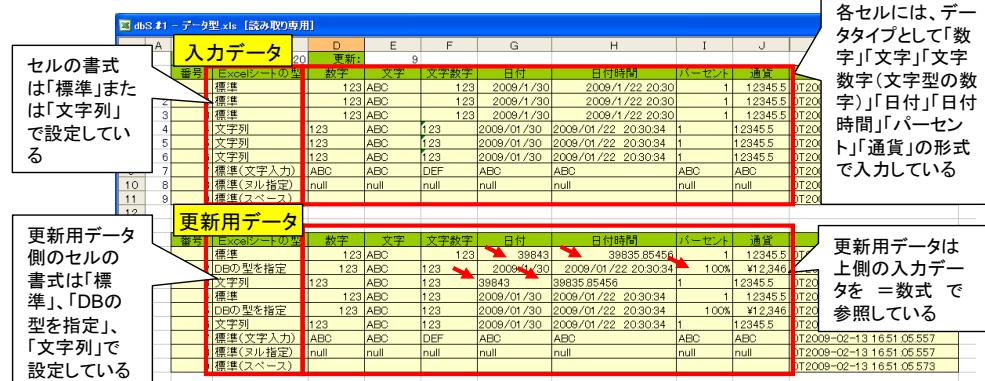
このようなExcelが自動的に処理するデータタイプの判定から、データ更新時に、DB側のデータとExcelシート上のデータのデータタイプのミスマッチによる更新エラーが発生することがあります。

以下に、Excelシート上に入力されたデータが、①標準での更新」と「②Null指定更新」のそれぞれの場合で、1)セル書式の設定の違いによってどのように変化するか、2)Excelのデータを、DB側に更新データとして送った場合にDB側ではデータタイプの違いによってどのようにデータが更新されるか、3)Excel上のデータでDBを更新したときにデータの値が変化する場合、について記載していますので参考にしてください。

②Excel上に入力したデータの変化と、DBの更新結果

1) 標準での更新時 (Null値での更新ができない)

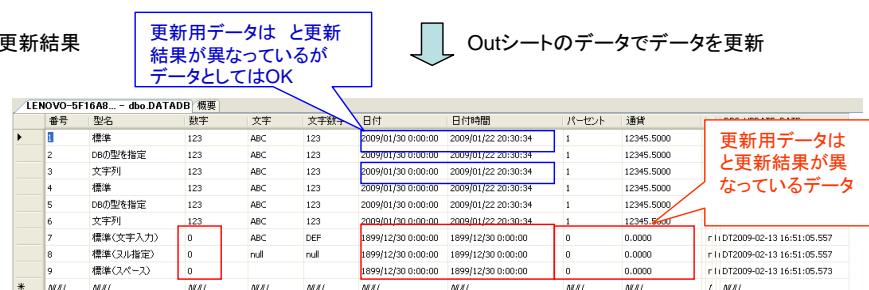
【1】Excelシートのデータ



各セルには、データタイプとして「数字」「文字」「文字数字」「日付」「日付時間」「パーセント」「通貨」で入力している

更新用データは上側の入力データを =数式 で参照している

【2】DBの更新結果

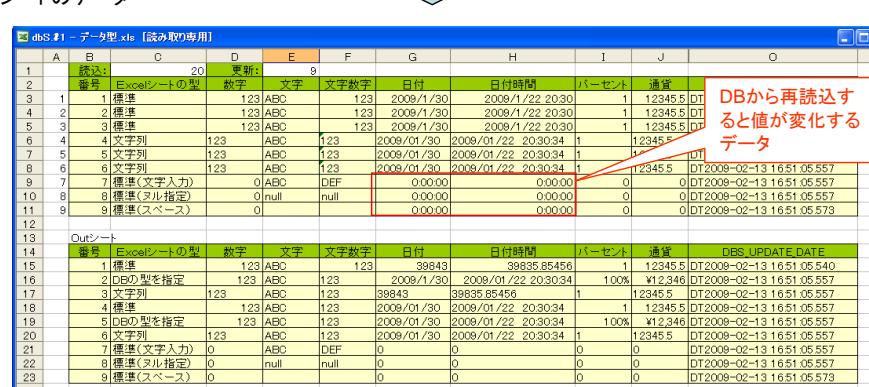


更新用データはと更新結果が異なっているがデータとしてはOK

Outシートのデータでデータを更新

更新用データはと更新結果が異なっているデータ

【3】Excelシートのデータ



DBから再読込むと値が変化するデータ

DBから再読込むと値が変化する

2) Null値更新時 (Null値での更新が可能)

Null値更新時の仕様											
※1:セルの値が“null”的には、Null値で更新されます。											
※2:DBの型が数字のときに、文字を登録したときには、Null値で更新されます。											
※3:DBの型が日付のときに、文字を登録したときには、Null値で更新されます。											

【1】Excelシートのデータ

dbS #1 データ型.xls [読み取り専用]											
セルの書式	入力データ										
	番号	読み取りシートの型	数字	文字	文字数字	日付	日付時間	パーセント	通貨	オプション	DBの型
1	標準	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		DBの型を指定
2	DBの型を指定	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		DBの型を指定
3	文字列	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		DBの型を指定
4	標準	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		DBの型を指定
5	DBの型を指定	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		DBの型を指定
6	文字列	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		DBの型を指定
7	標準(文字入力)	ABC	ABC	DEF	ABC	ABC	ABC	ABC	ABC	ABC	DBの型を指定
8	標準(フル指定)	null	null	null	null	null	null	null	null	null	DBの型を指定
9	標準(スペース)										DBの型を指定
10	標準(スペース)										DBの型を指定
11	9										
12											

各セルには、データタイプとして「数字」「文字」「文字数字(文字型の数字)」「日付」「日付時間」「パーセント」「通貨」で入力している

更新用データ側のセルの書式は「標準」または「DBの型を指定」、「DBの型を指定」、「文字列」で設定している

更新用データは上側の入力データを = 数式で参照している

【2】DBの更新結果

更新用データは と更新結果が異なっているがデータとしてはOK											
Outシートのデータでデータを更新											
Null指定更新をおこなうと、正しくNullとして行使される											
r1 DT2009-02-13 16:56:37.947 r1 DT2009-02-13 16:56:37.947 r1 DT2009-02-13 16:56:37.947											
... NULL											

【3】Excelシートのデータ

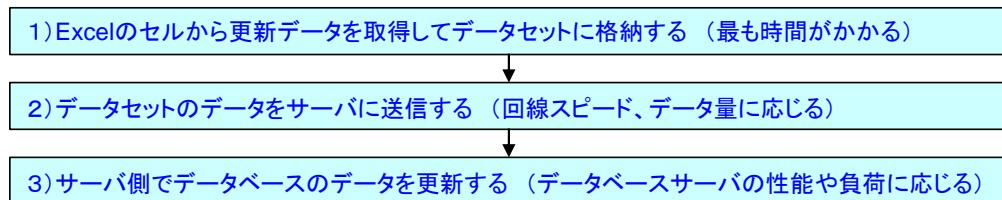
Outシート											
番号	読み取りシートの型	数字	文字	文字数字	日付	日付時間	パーセント	通貨	DBS UPDATE DATE		
									D	E	F
1	標準	123	ABC	123	39843	38835.85456	1	12345.5	DT2009-02-13 16:56:37.947		
2	DBの型を指定	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
3	文字列	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
4	標準	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
5	DBの型を指定	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
6	文字列	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
7	標準(文字入力)	ABC	DEF	ABC	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
8	標準(フル指定)	NULL	NULL	NULL	NULL	NULL	NULL	NULL	DT2009-02-13 16:56:37.947		
9	標準(スペース)	NULL	NULL	NULL	NULL	NULL	NULL	NULL	DT2009-02-13 16:56:37.947		
10											
11											
12											
13											
14	Outシート										
15	標準	123	ABC	123	39843	38835.85456	1	12345.5	DT2009-02-13 16:56:37.947		
16	DBの型を指定	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
17	文字列	123	ABC	123	39843	38835.85456	1	12345.5	DT2009-02-13 16:56:37.947		
18	標準	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
19	DBの型を指定	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
20	文字列	123	ABC	123	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
21	標準(文字入力)	ABC	DEF	ABC	2009/1/30	2009/1/22 20:30:34	1	12345.5	DT2009-02-13 16:56:37.947		
22	標準(フル指定)	NULL	NULL	NULL	NULL	NULL	NULL	NULL	DT2009-02-13 16:56:37.947		
23	標準(スペース)	NULL	NULL	NULL	NULL	NULL	NULL	NULL	DT2009-02-13 16:56:37.947		
24											

DBから再読み込むとNull値は空文字になる

③Excel上のデータでDBを更新したときに、データの値が変化する場合

No.	入力データの型	Excel側のセル書式の設定		データベース側	結果		備考
		編集用シート	出力用シート		データ型	V3.0.18.0以前	
1	文字	標準	標準	文字型	○	○	文字型として登録されます。
2	文字	標準	標準	数字型	○	○	文字が入力された場合には 0 として登録されます。 Null指定をおこなうと Null で登録されます。
3	文字	文字型 YYYY/MM/DD	文字型	日付型	○	○	日付データとして正常に登録されます。
4	数字	標準	標準	文字型	低速:OK 高速:NG	○	【Ver3.0.18以前】 ①低速更新時はOK(更新データ項目数が100未満) ②高速更新時はデータの個数が文字と数字の多い方の型に、Excel側の書式が自動的に設定されるため数値が多いときにNGで、空白になりました。 【Ver3.0.19.0以降】 いずれでもOKになるように、プログラムの変更をおこないました。
5	数字	標準	標準	数字型	○	○	数字型として登録されます。
6	数字	標準	標準	日付型	○	○	日付データとして正常に登録されます。
7	混合 (文字or数字)	標準	標準	文字型	低速:OK 高速:NG	○	【Ver3.0.18以前】 ①低速更新時はOK(更新データ項目数が100未満) ②高速更新時はデータの個数が文字と数字の多い方の型に、Excel側の書式が自動的に設定されるため数値が多いときにNGで、空白になりました。 【Ver3.0.19.0以降】 いずれでもOKになるように、プログラムの変更をおこないました。
8	混合 (文字or数字)	標準	標準	数字型	○	○	文字が入力された場合には 0 として登録されます。 Null指定をおこなうと Null で登録されます。
9	混合 (文字or数字)	標準	標準	文字型	○	○	出力用シートに数式として =TEXT(. . . , YYYY/MM/DD) の書式変換をおこなっておくと、日付文字になりますが、数式を利用しないと、数字の文字列で登録されます。
10	日付	標準	標準	文字型	○	○	出力用シートに数式として =TEXT(. . . , YYYY/MM/DD) の書式変換をおこなっておくと、日付文字になりますが、数式を利用しないと、数字の文字列で登録されます。
11	日付	数値型	標準(数字)	数字型	○	○	数字として登録されます。
12	日付	日付型(YYYY/MM/DD)	日付型	日付型	○	○	日付データとして正常に登録されます。

5.5.3 更新処理時の処理スピードについて



データ件数(レコード件数) × フィールド数 < 100(※1) の時には、更新処理時間に違いは出ませんが、データ件数が100件、1000件と件数が増えると、Excel側で設定したデータの配列方式やデータタイプの設定によって、処理スピードに大きな違いが出てきます。

(※1. データ件数10件、フィールド数9個のときは 90
データ件数5件、フィールド数30個 のときは 150 と計算します)

1000件ほどのデータ更新をおこなったときに、2) と 3) については、一般的な環境では、それぞれ数秒程度で終わりますが、1) の処理については、Excel側の更新データの設定方法の違いによって、数秒で終わるケースと、数分以上かかるケースがあります。

dbSheetClientでは、1)の時間をなるべく短縮するために、**高速データ転送処理(高速処理)**を利用していますが、そのためには、Excel側で更新データを以下のように設定しておく必要があります。設定を間違ったり、条件に合わない場合には、**通常データ転送処理(低速処理)**で処理をおこなうため、10倍以上の時間を要しますので、十分注意してください。

(※ただし、データ件数(レコード件数) × フィールド数 < 100 のときには、高速処理の方が遅くなるケースがあるため、低速処理を実行しています。)

1) 更新データは、行方向に配列すること。

列方向のデータ更新には10倍以上時間がかかります。

→ 列方向に並んでいるデータは、一旦、タスクタイプ410.コピー(汎用パラメータ2の「行列の入替え」を使用)処理などを利用すると、行方向に並べ替えることができます。

2) Excel側の更新データの直前の行に、データベースのフィールド名を設定しておくこと。
指定更新 のときには、一部のフィールドだけの設定でもOKです。

→ フィールド名を間違えると、低速処理になります。

3) Excel側の最後のフィールド名の直後には、空白セルを必ず入れること。

※フィールド名の直後に別の文字などが続くと遅くなります

商品コード	商品名	単価	DBS_STATUS	...	DBS_UPDATE_USER	DBS_UPDATE_DATE	更新行数	10
1	商品1	1250	1	...	maeda	DT2008-07-20 17:23:11.357		
2	商品2	1300	1	...	maeda	DT2008-07-20 17:25:21.717		
3	商品3	1400	1	...	maeda	DT2008-07-08 17:00:32.843		

※フィールド名の直後には、一つ以上の空白セルを空けてください。

商品コード	商品名	単価	DBS_STATUS	...	DBS_UPDATE_USER	DBS_UPDATE_DATE	更新行数	10
1	商品1	1250	1	...	maeda	DT2008-07-20 17:23:11.357		
2	商品2	1300	1	...	maeda	DT2008-07-20 17:25:21.717		
3	商品3	1400	1	...	maeda	DT2008-07-08 17:00:32.843		

4) 日付型のデータのときには、Excel側の書式設定も、必ず、日付型にしておくこと。

→ Excel側の書式が「文字型」のときには、低速処理になります。

参考:処理時間ログファイル dbS_ProcTimeLog(***).log に更新処理がおこなわれたときのログを書き込んでいますので、高速処理が実行されたか低速処理が実行されたかの確認ができます。

第6章 レスポンス改善のための対策

6.1 起動時間や実行時のレスポンス改善のための対策

dbSheetClient を起動してから、ログイン画面の表示、プロジェクトの選択から実行画面が表示されるまでには、次の処理がおこなわれています。

6.1.1 プロジェクトの起動までの処理と、起動時間短縮のための対策

起動処理に時間がかかる場合には、時間のかかっている処理の実行ログファイルを参考にして、以降に記述いたしました対策を施すことにより起動時間を改善・短縮させることができます。

処理	概要	遅くなる主な原因	対策
dbSheetClient のバージョンチェック	クリックワンスで起動された時にはバージョンの状態をチェックする。CD 版で起動された時はチェックしない。	ネットワーク環境、PC のパワーなどの関係でクリックワンスの認識に時間がかかる。	起動の遅い PC はクリックワンスではなく CD 版で動作を比較。
ネットワーク状態のチェック	ネットワークの接続状態をチェックする。DBS サーバを確認する。	①名前解決ができない。 ②プロキシサーバでチェックがかかっている。 ③インターネットセキュリティなどのチェックに時間がかかる。	①DNS などの環境を調べる。 ②プロキシサーバの設定を見直す。 ③キュリティなどの設定を見直す。
クライアント構成のチェック	クライアントのシステム環境をチェックする。	①初めての起動時には、端末情報の登録とシステム環境のダウンロードが必要なため時間を要します。 ②ネットワークのダウンロード処理スピードが遅い。	①2 回目の起動を行い、時間の差があるかをチェックする。 ②可能であればネットワークのスピードを上げる。
サーバ構成のチェック	サーバのライセンス状況をチェックする。	①名前解決ができない。 ②プロキシサーバでチェックがかかっている。 ③インターネットセキュリティなどのチェックに時間がかかる。	①DNS などの環境を調べる。 ②プロキシサーバの設定を見直す。 ③キュリティなどの設定を見直す。
プロジェクトの直接起動	ログイン画面を表示しないで、プロジェクトを起動します。 終了時はログインメニューを表示しないで終了します。		コマンドラインからの自動ログイン機能を使用するとログイン処理をスキップすることで起動時間を短縮できます。 ※CD 版コマンドラインからの起動方法 <実行版>[dir]dbSheetClient.exe c <開発版>[dir]dbSheetDevelop.exe pid=99,uid=abcdefg,upw=**** [dir](標準インストールフォルダ)は C:¥dbSheetClient¥DevelopV5¥bin¥ pid=プロジェクト ID、uid=ユーザ ID、upw=パスワード) ※クリックワンスからの起動方法はカスタマイズが必要ですのでお問い合わせください

【次ページに続く】

【前ページより続く】

処理	概要	遅くなる主な原因	対策
ログイン時事前コンパイル処理	dbSheetClient をスタートアップ起動し、コンパイル処理をして自動終了します。	初回起動時に ① .NetFramework2.0 の起動 ②アプリケーションのコンパイルを行うため、最初の起動に時間がかかります。	スタートアップに下記のパラメータを記述したショットカットを作成します。 ※CD 版コマンドラインからの起動方法 <実行版>[dir]dbSheetClient.exe mod=startup [dir](標準インストールフォルダ)は C:\dbSheetClient\Develop\V5\bin\
ログインユーザ認証	ユーザ認証を行い、サーバからログインユーザが実行可能なプロジェクトリストを取得する。	①DBS サーバの処理時間がかかっている。 ②ネットワークの処理スピードが遅い。	DBS サーバの資源の確認または、DBMS の環境を確認する。
プロジェクト情報のダウンロード	選択されたプロジェクト定義情報をサーバからダウンロードする。	①プロジェクト定義のサイズが大きくダウンロードに時間がかかる。 ②ネットワークの処理スピードが遅い。	サーバシステムでプロジェクト定義情報の保存を「許可」にすると、ローカル PC に保存された定義情報から起動させることができるようになり、ダウンロード時間を0にすることができます。(既定は「禁止」)
DB 定義、ブック定義、メニュー定義の読み込み	プロジェクト定義情報から DB 定義、ブック定義、メニュー定義をメモリに格納する。	DB 定義、ブック定義、メニュー定義の行数が多い。	メニュー項目が少ないほどメモリに格納する時間が少くなり、スピードは改善します
Excel の起動	Excel を起動する。	①Excel の起動に時間がかかる。 ②システムのメモリ資源が少ない。	環境設定で、「Excel をキャッシュする」を ON にすると、ログイン画面が表示される前に一度キャッシュされるので、プロジェクト選択後の処理は改善されます。
Excel ブックのオープン	メニュートップのブックをサーバからダウンロードしオープンする。	①ブックサイズが大きい。 ②ネットワークのダウンロードの処理スピードが遅い。	①ブック定義のホスト指定を「DBS サーバ」にして、ローカル PC に Excel ブックをキャッシュすると、次回の起動からサーバのブックが更新されるまで、サーバからダウンロードしないでキャッシュからブックを開くようになります。 ②ブックサイズを小さくする。 1) 編集を繰返しあなつてると、ブックサイズが異常に大きくなる場合があります。このような場合には、一度データを取込んだ行などにデータがなくとも、行削除などをあなつと小さくなる場合があります。 2) ブックを分割して、複数のブックで運用する。
開始処理実行 ブックオープンなど	開始終了処理を実行する。	開始処理にクエリー実行などのタスクがたくさん定義されている。	①始処理の定義を最低限にする。 ②タスクタイプ 301.起動モードでタスク定義、展開定義、クエリー定義などをメモリに格納すると、メモリ格納時間は余分にかかりますが、ハードディスクへのアクセス回数を少なくすることができ、相対的にスピードが改善される場合があります。

6.1.2 実行時のレスポンス改善の各種対策方法について

実行時のログファイルを確認し、時間のかかっている処理を判別します。

処理時間のログは、C:\dbSheetClient\Develop\log\dbS_(yyyy-mm-dd) ProcTime999.log のファイルに随時書き込まれます。(9999)は数値で保存します。

■処理時間の確認（計算）方法

ログの内容から、処理時間を簡単に計算するための Excel シート（dbS_ProcTimeLogCheck.xls）を用意していますのでご利用ください。

#	操作	画面
	オンラインヘルプより、ダウンロードのダイアログを表示します。 [保存] を選択します。	
②	任意の保管場所を選択し保存します。 圧縮ファイルがダウンロードされます。 <例>ディスクトップを選択し保存します。	
③	圧縮ファイル（Zip）アイコンを右クリック [すべて展開] を選択。 解凍ウィザードに従って処理を完了させてください。 完了するとフィルダが作成され、フォルダが開きます。 dbS_ProcTimeLogCheck(100215).xls ファイルをクリックします。	
	dbS_ProcTimeLogCheck(100215).xls が実行されます。	
	赤枠のエリア「処理時間ログ内容」に C:\dbSheetClient\Develop\log\dbS_(yyyy-mm-dd) ProcTime999.log の内容をコピーストします。((***)には日付文字が入る：(yyyy-mm-dd は英語表記の日付) その結果。「前処理からの経過時間」の長い処理に課題があることが判断できます。	

6.1.3 Excel の使い方として

大量のデータを Excel に一度に読み込むと、手動でも読み込み時間がかかり遅くなります。

また Excel 自体も重くなります。

そこで Excel でレスポンスを上げるために、例えば、最初の読み込みでは、1～10000 件までを読み込み、次へのボタンを押下すると 10001～20000 件までを読み込むように分割して読み込むことで、読み込みスピードが改善され、Excel が重くなるのを軽減させることが可能です。

またデータが分類コードなどが利用可能な場合、その分類の範囲で読み込むようにすることでも Excel への読み込むスピードの改善が可能です。そのような設計をお勧めします。

第7章 機能別動作原理と定義について

7.1 ハイパーテスク実行の原理と定義方法

【処理の概要】

dbSheetClient では、通常タスクはメニュー やボタンをクリックしたときに実行できますが、ハイパーテスク実行の設定をおこなうと Excel シート上に配置した文字や図（オートシェイプや画像）からタスクを直接実行することができるようになります。

7.1.1 ハイパーテスク実行の動作原理

ハイパーテスク実行は、Excel のハイパーリンクのイベントおよびセレクションチェンジイベント（選択されているセルが別のセルに変更したときのイベント）を dbSheetClient が取得し、イベントをキャッチしたのち、選択されているセルの文字列を取得して、ボタン名かタスク NO かを判断して、該当するタスクを実行しています。

したがって、ハイパーテスク実行を設定するには、

- 1) セルまたは図にハイパーリンクを設定し、2) のセルにリンクさせる
- 2) セルに実行したいタスクの「ボタン名」か「タスク NO」を入力する

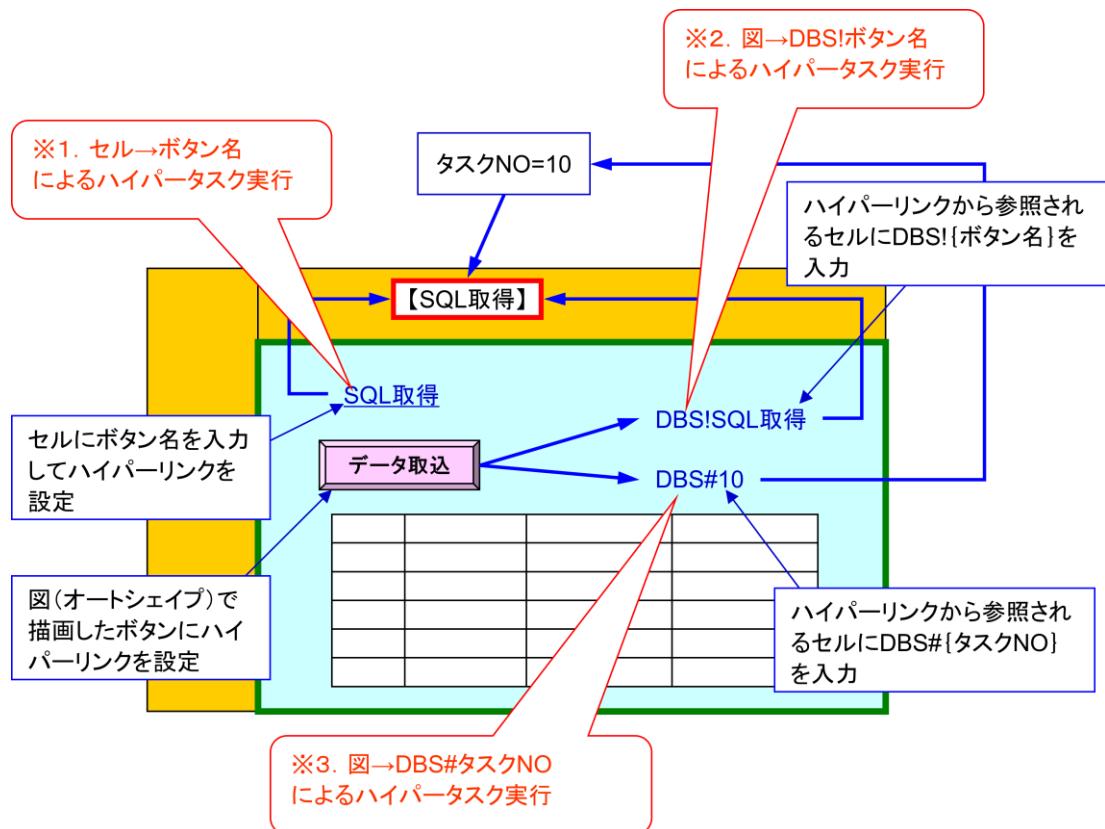
の2段階の設定が必要になります。また、ハイパーテスク実行の設定方法は3通りあります。

※1. セル→ボタン名 によるハイパーテスク実行

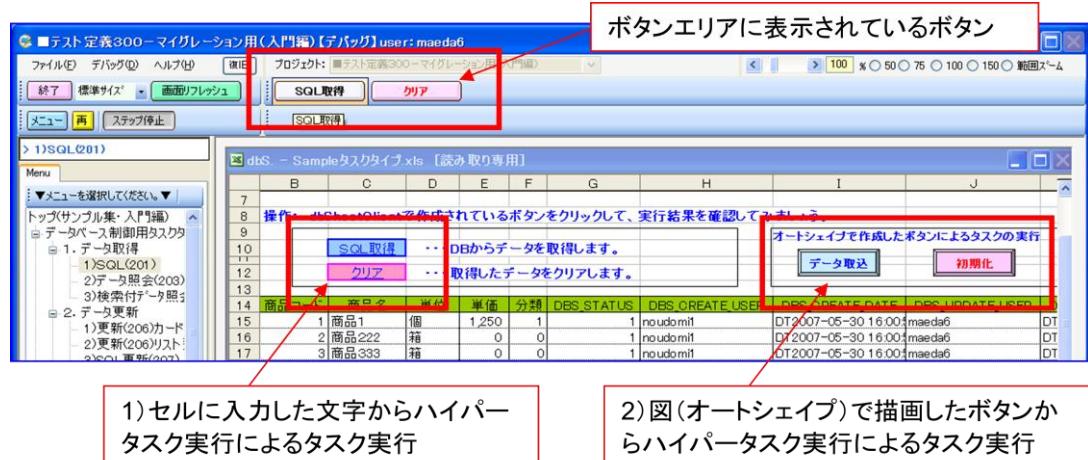
※2. 図→DBS!ボタン名 によるハイパーテスク実行

※3. 図→DBS#タスク NO によるハイパーテスク実行

以下に図を示します。



7.1.2 ハイパーテスク実行の定義方法



図では、ハイパーテスク実行の実際の設定方法として、以下の2つの例を示しています。

① セルに入力した文字からハイパーテスク実行によるタスク実行

【設定方法】

1. 適当なセルに、実行したいタスクが設定されている「ボタンエリアのボタン名」と同じ文字を入力する。上図では、「SQL 取得」と「クリア」を入力しています。
2. セルを右クリックして、「ハイパーリンク」を選択し、以下のように、ハイパーリンクを設定します。
リンク先 → 「このドキュメント内」
セル参照 → 「SQL 取得」や「クリア」入力したセルを指定（名前定義後、定義名で指定也可）
以上の設定で、ハイパーリンクを設定したセルをクリックすると、ボタンに定義されているタスクが実行されます。

【注意】

ボタン名と異なった文字の場合にはタスクは起動しません。半角・全角の違いに注意して文字を入力してください。

② 図(オートシェイプ)で描画したボタンからハイパーテスク実行によるタスク実行

【設定方法】

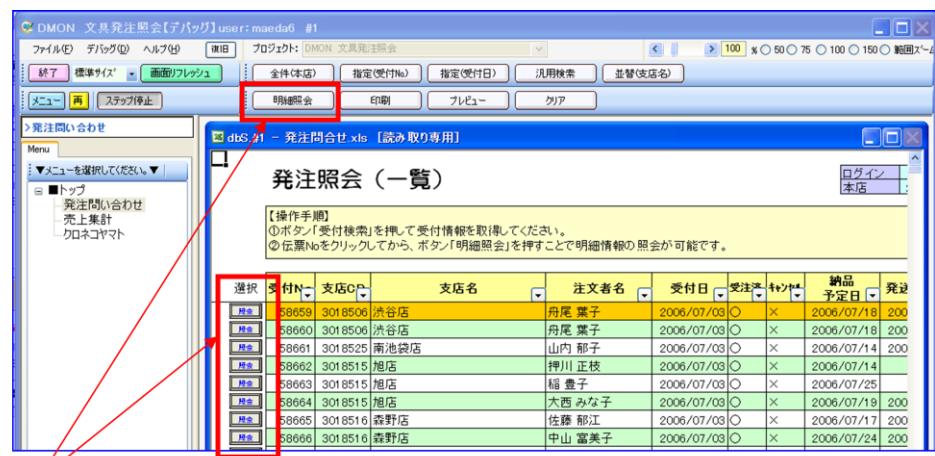
1. 適当なセルに、実行したいタスクが設定されている「ボタンエリアのボタン名」を利用して DBS! {ボタン名} と入力する。上図ではボタン(図)の下にセル I11 に DBS!SQL 取得、J11 に DBS!クリアと入力しています。
2. ボタン(図やオートシェイプ)を描画し、ボタンに表示する適当な文字を入力。上図では「データ取込」と「初期化」を入力しています。文字は、タスクのボタン名と一致させる必要はありません。
3. 図(ボタン)を選択して右クリックして、「ハイパーリンク」を選択し、以下のようにハイパーリンクを設定します。
リンク先 → 「このドキュメント内」
セル参照 → 「DBS!SQL 取得」や「DBS!クリア」と入力したセルを指定
以上の設定で、ハイパーリンクを設定したボタン(図)をクリックすると、ハイパーリンク先のセルにカーソルが移動し、そのセルに入力されているタスクが実行されます。

7.1.3 タスクタイプ:426. ハイパーテスク実行設定 の使用方法

任意のセルに対して自動的にハイパーテルクを設定することでハイパーテスク実行可能にし、またハイパーテスク実行用のボタンの追加・削除をします。

詳細の設定方法については、『タスクタイプリファレンスマニュアル「タスクタイプ426：ハイパーテスク実行設定」』を参照してください。

下図は、ハイパーテスク実行設定を利用して、自動的にハイパーテスク実行用のボタンが追加されている例です。



「明細照会」というボタン名に設定されているタスクを、「照会」というハイパーテスク実行ボタンで実行できるように自動作成しています。このとき、ハイパーテスク実行ボタンは複数連続的に作成することができます。



「戻る」というボタン名に設定されているタスクと、「プレビュー」というボタン名に設定されているタスクを、おのの、「戻る」と「プレビュー」いうハイパーテスク実行ボタンで実行できるように自動作成しています。

7.1.4 ハイパーテスク実行を利用するときの注意点

ハイパーテスク実行は、ハイパーリンクが設定されたセルや、ハイパーリンクが設定された図（オートシェイプや画像）をクリックすることで、タスクを起動するようになりますが、動作原理には以下のようない違があります。

① セルに直接ハイパーリンクが設定されているときの動作

セルがクリックされると dbSheetClient は直ちにハイパーリンクイベントをキャッチすることができる、その時点のセルの文字列を取得して、ボタン名かタスク No かを判断して、該当するタスクを実行します。

② 図（オートシェイプや画像）にハイパーリンクが設定されているときの動作

図（オートシェイプや画像）のハイパーリンクのイベントは、直接 dbSheetClient ではキャッチすることができないため、図（オートシェイプや画像）をクリックしたのち、ハイパーリンクの参照先のセルにカーソルが移動したときのセレクションチェンジイベントをキャッチして、そのセルの文字列からボタン名かタスク NO かを判断して、該当するタスクを実行しています。

したがって、上記の動作原理の違いを踏まえて、以下の点に注意する必要があります。

【 注意 】

- ① 参照先のセルが選択されている状態で、図（オートシェイプや画像）で作成したボタンからのハイパーテスク実行の時には、選択されるセルが変化しないことからセレクションチェンジイベントが発生しないため、タスクを実行することはできません。
したがって、図（オートシェイプや画像）で作成したボタンからハイパーテスク実行した後には、セルの位置を、ジャンプなどのタスクタイプで、ハイパーリンクの参照セルとは違うセルに移動させておくなどの対策が必要です。
- ② 図（オートシェイプや画像）ではなく、DBS!ボタン名、DBS#タスク NO が入力されているセルをマウスなどで選択したときも、セレクションチェンジイベントが発生するため、タスクが実行します。
マウスで簡単にセルをクリックされることがないように、列や行などを非表示にするなどの対策が必要です。
- ③ ボタン名によるハイパーテスク実行は、ボタンエリアにボタンが表示されているときだけ有効になります。そのため、実行権限がそのまま適用されます。
ボタンエリアにボタンが表示されていないときや、実行権限が無くボタンが無効の時には実行できません。
- ④ タスク NO によるハイパーテスク実行を行う前には、タスクタイプ : 304.ハイパーテスク実行制御で「有効」にしておく必要があります。「無効」の状態では実行できません。
なお、タスク NO による実行は、ボタン名のように権限の設定が働かないため、だれでも実行できるので注意が必要です。
- ⑤ ハイパーリンクの参照セルにロックが掛かっていると、タスクを実行することができません。
シートを保護するときには、必ず、セルのロックをはずした状態にしてください。

第8章 IOTG(アイオーティージー)の仕組みと作成手順について

本章では、IOTG(アイオーティージー) 機能の仕組みと作成手順についてについて説明します。

8.1 IOTG(アイオーティージー)の仕組み

IOTG(アイオーティージー) 機能を実行したときの処理を説明します。

8.1.1 IOTGの目的

Web アプリケーションの開発を効率よく進めるために、開発で作成するプロジェクト定義情報や Excel シートを自動的の作成する機能を提供します。

◆作成するシートデザイン：リスト、カード

8.1.2 IOTGの機能

- ① データベースのテーブルからデータを読み、Excel シートに展開するための定義を自動作成
- ② Excel シートのデータをデータベースのテーブルに追加・更新・削除するための定義を自動作成
- ③ データ件数に応じて、行の書式を自動的に変更するための定義を自動作成
- ④ データ最終行以降に、指定数の空行を自動生成するための定義を自動作成

8.1.3 IOTGの使用制限

- ① テーブルの構成に「DBS 同時実行制御フィールド」の項目を含まない場合は作成されません。
- ② 使用する Excel ファイルの拡張子が ”xls” で、項目数が 240 を超えるテーブルの場合は、作成されません。

8.2

基本的な定義シートの作成手順

基本的な定義シートの作成手順について、説明します。

※表示形式は、リスト型とカード型がありますが、本項番での説明は、リスト型を説明します。

◆ 準備

IOTG機能を実行する前に、次の2つの定義を準備します。

- データベース定義
- ブック定義

◆ 手順

IOTG機能を実行する手順について、説明します。

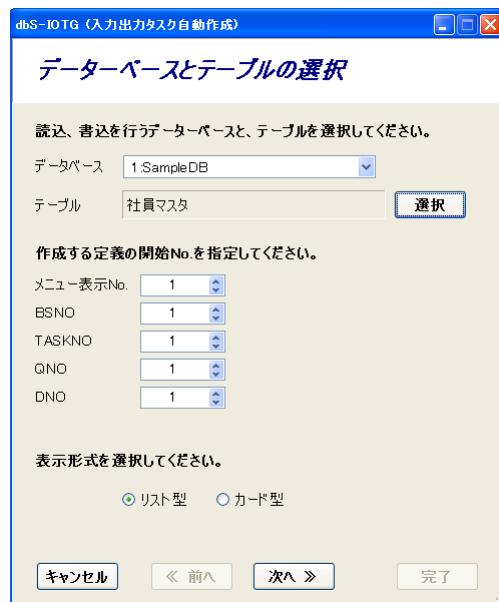
1. IOTG機能の起動
2. データベースのテーブルを選択
3. ブックの選択と各シートの設定
4. メニューとボタンの設定

1. IOTG機能の起動

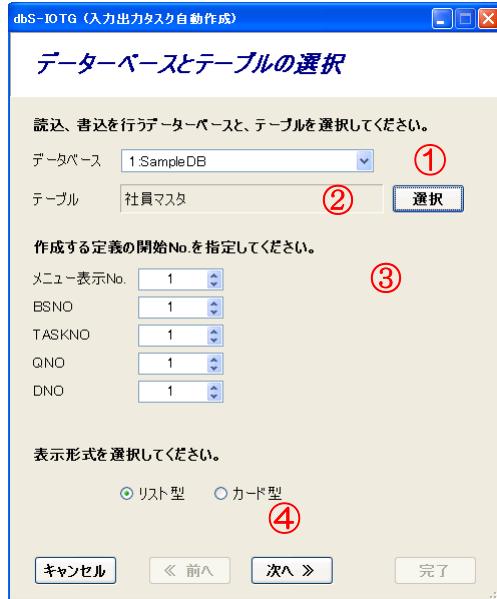


をクリックします。

実行画面が表示されます。



2. データベースのテーブルを選択

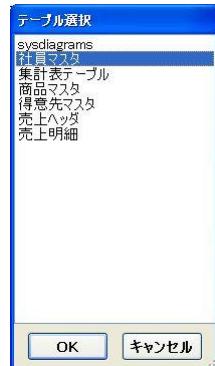


① データベースを指定します。

[▼] データベース定義の一覧が表示されます。

② テーブルを指定します。

[選択] ボタンをクリックすると、テーブルの一覧が表示されます。



③ 定義開始番号を指定します。作成する定義の番号を指定します。

④ 次の設定画面を表示します。

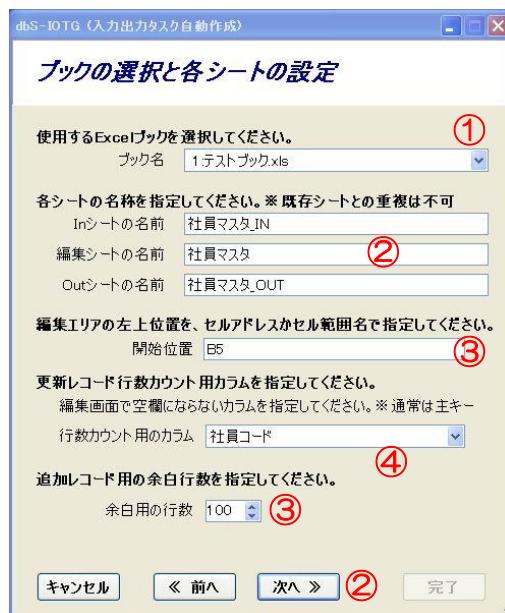
[次へ >>] ボタンをクリックします。

※ 処理を中断するには

[キャンセル] ボタンをクリックします。

3. ブックの選択と各シートの設定

●リスト型の場合



① Excel ブックを選択します。

[▼] ブック定義の一覧が表示されます。

② 追加するシートの名前を指定します。

- In シートの名前
- 編集シートの名前
- Out シートの名前

③ データを配置する開始位置を指定します。

④ データの行数をカウントする項目を指定します。

通常は、主キーの項目を選択します。

[▼] テーブル内の項目の一覧が表示されます。

⑤ 追加レコード用の余白行数を指定します。

編集シートに、罫線などの書式をコピーします。

⑥ 次の設定画面を表示します。

[次へ >>] ボタンをクリックします。

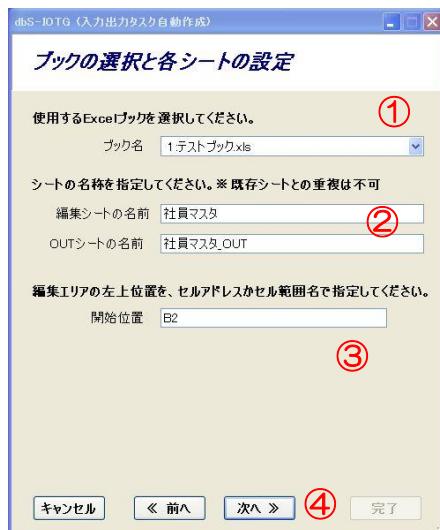
※ 処理を中断するには

[キャンセル] ボタンをクリックします。

※ 前の設定画面を表示するには

[<< 前へ] ボタンをクリックします。

●カード型の場合



① Excel ブックファイルを指定します。

ブックファイルの定義が複数ある場合、【▼】をクリックすると、ブックファイル定義の一覧を表示しますので、そこから選択します。

② 追加するシートの名前を設定します。

- ・編集シートの名前
- ・Out シートの名前

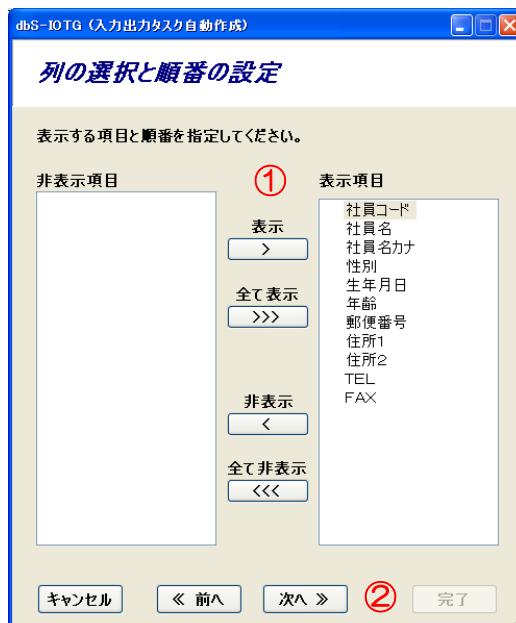
③ データを配置する開始位置（セルアドレスまたはセル範囲名）を設定します。

④ 【次へ >>】ボタンをクリックすると、次の設定画面へ進みます。

【<< 前へ】ボタンをクリックした場合は、前の設定画面へ戻ります。

【キャンセル】ボタンをクリックした場合は、IOTG の実行を中止します。

■項目の設定をおこなう（カード型のみ）



- ① 表示・編集する項目とその順番を設定します。

「非表示項目」と「表示項目」にある項目名をクリックすると、その項目が選択された状態になります。「非表示項目」にある項目を選択した状態で **【表示】** をクリックすると、選択された項目が「表示項目」に移動します。「表示項目」にある項目を選択した状態で **【非表示】** をクリックすると、選択された項目が「非表示項目」に移動します。

【全て表示】 をクリックすると、全ての項目が「表示項目」へ移動します。**【全て非表示】** をクリックすると、全ての項目が「非表示項目」へ移動します。

「表示項目」の中にある項目は Drag Drop 操作の要領で順番を変更することができます。

- ② **【次へ >>】** ボタンをクリックすると、次の設定画面へ進みます。

【<< 前へ】 ボタンをクリックした場合は、前の設定画面へ戻ります。

【キャンセル】 ボタンをクリックした場合は、IOTG の実行を中止します。

4. メニューとボタンの設定



① メニューを追加するか指定します。

チェックボックスをオフにすると、メニュー定義とボタン定義は作成されません。
メニュー名、読み込ボタン名、更新ボタン名の欄へは入力できなくなります。

② 追加するメニュー、ボタンの名前を設定します。

- ・メニュー名
- ・読み込ボタン名
- ・更新ボタン名

③ 設定を完了します。

[完了] ボタンをクリックします。

処理を実行します。

※ 処理を中断するには

[キャンセル] ボタンをクリックします。

※ 前の設定画面を表示するには

[<< 前へ] ボタンをクリックします。

8.2.1 作成されるプロジェクト定義情報

プロジェクトの定義情報に、以下「リスト型の」情報が追加されます。

- ◆ メニュー定義 1行
- ◆ ボタン定義 2行
- ◆ タスク定義 10行 (2組)
- ◆ クエリー定義 3行
- ◆ 展開定義 6行

◆ メニュー定義



The screenshot shows the dbSheetClient menu definition editor. On the left, a tree view shows a 'メニュー' node with a '社員マスタ' child node. On the right, a grid lists menu items with columns for '表示NO', '階層', 'メニュー名', 'BKNO', and 'ブック名'. The grid contains 15 rows of data, with the last row being a summary or placeholder.

メニューに、**社員マスタ** が追加されています。

◆ ボタン定義



The screenshot shows the dbSheetClient button definition editor. On the left, a tree view shows a 'メニュー' node with a '社員マスタ' child node, which has two button nodes: '読込' and '更新'. On the right, a grid lists button definitions with columns for 'BSNO', 'ボタンNO', 'ボタン名表示名', 'TASKNO', 'タスク名', '権限レベル', '定義有効無効フラグ', 'ガイダンス', 'コメント', and 'ボタン'. The grid contains 8 rows of data.

ボタンセット **TG_BS1_社員マスタ** が追加されています。

読込 ボタンと、**更新** ボタンが追加されています。

◆ タスク定義

B	C	D	E	F	G
TASKNO	タスク名	処理順NO	タスクタイプNO	タスクタイプ名	タスクタイプ属性
1	1 TG_T_1 社員マスター Read	1	411	クリア	数式と値
2	1 TG_T_1 社員マスター Read	2	201	SQL	照会
3	1 TG_T_1 社員マスター Read	3	201	SQL	照会
4	1 TG_T_1 社員マスター Read	4	417	自動計算	再計算実行
5	1 TG_T_1 社員マスター Read	5	410	コピー	書式: コメント・入力規則・列幅
6	2 TG_T_2 社員マスター Write	1	318	メッセージ表示	エラー判定
7	2 TG_T_2 社員マスター Write	2	410	コピー	数式
8	2 TG_T_2 社員マスター Write	3	417	自動計算	再計算実行
9	2 TG_T_2 社員マスター Write	4	201	SQL	指定更新
10	2 TG_T_2 社員マスター Write	5	320	サブタスク実行	
11					
12					

2つのタスクが追加されています。

• 読込用タスク `TG_T_1 社員マスター Read`

1. 読込領域をクリアします。
2. In シート に、データを読み込みます。
3. 編集シート に、データを読み込みます。
4. 再計算を実行します。
5. 書式をコピーします。

• 更新用タスク `TG_T_2 社員マスター Write`

1. 更新する行がなければ、エラーメッセージを表示し、更新を中断します。
2. 数式をコピーします。（編集シートのデータを参照するなど）
3. 再計算を実行します。
4. データを更新します。
5. 再度、データを読み込みます。

8.2.2 読込用タスク TG_T_1 社員マスター_Read

「リスト型」の説明

1. 読込領域をクリアします。

・タスク定義

	A	B	C	D
1	タイトル	サブ	*	内容
2	TASKNO		*	1
3	タスク名		*	TG_T_1 社員マスター_Read
4	処理順NO		*	1
5	タスクタイプNO		*	411
6	タスクタイプ名		*	クリア
7	タスクタイプ属性			数式と値
8	BKNO		*	1
9	ブック名		*	テストブック.xls
10	シート名			社員マスター
11	DNO			4

・展開定義

	A	B	C	D
1	タイトル	サブ	*	内容
2	DNO		*	4
3	データ展開名			TG_D4 社員マスター_Clear
4	処理順NO			1
5	定義区分			タスク定義
6	タイプ			クリア
7	タスク・展開属性			数式と値
8	定義有効無効フラグ			
9	コメント欄			
10	シート名			社員マスター
11	セル指定属性			間接
12	セル範囲			@社員マスター TG_T1_CopyArea_E
13	シート保護有無			
14	シート保護解除PW			
15	自動計算方法			

・Excel ブック

名前定義 社員マスター TG_T1_CopyArea_E (社員マスター O9)

\$B\$6:\$L\$107 データを読み込む前の領域を指しています。

データを読み込んだ後では、領域は変更します。

2. In シート に、データを読み込みます。

・タスク定義

	A タイトル	B サブ	C * 内容
1	TASKNO		*1
2	タスク名		*TG_T_1 社員マスタ_Read
3	処理順NO		*2
4	タスクタイプNO		*201
5	タスクタイプ名		*SQL
6	タスクタイプ属性		*照会
7	BKNO		*1
8	ブック名		*テストブック.xls
9	シート名		社員マスタ_IN
10	QNO		1
11	クエリー名		TG_社員マスタ_I
12	DNO		1

・クエリー定義

	A タイトル	B サブ	C * 内容
1	クエリー名		*TG_社員マスタ_I
2	クエリータイプ		*照会
3	クエリー属性		*Query
4	DBNO		*1
5	エイリアス名		SampleDB
6	テーブル名		
7	BKNO	旧仕様	0
8	ブック名	旧仕様	
9	取得データ件数セル	旧仕様	
10	SQL文		Select * From [社員マスタ]
11	排他制御タイプ		
12	定義有効無効フラグ		
13	コメント欄		
14			

・展開定義

	A タイトル	B サブ	C * 内容
1			
2	DNO		*1
3	データ展開名		TG_D1_社員マスタ_Q_In
4	処理順NO		1
5	定義区分		クエリー定義
6	タイプ		照会
7	タスク・展開属性		Query
8	データ項目名		
9	数式OR値		
10	定義有効無効フラグ		
11	コメント欄		
12	シート名		社員マスタ_IN
13	セル指定属性		直接
14	セル範囲		@社員マスタ_IN_TG_Q1_I
15	セル展開範囲定義名		
16	シート保護有無		
17	シート保護解除PW		
18	自動計算方法		
19	行列方向	データ行列方向	
20	更新行OR列数指示セル	取得データ件数セル	

・Excel ブック

名前定義 社員マスタ_IN_TG_Q1_I (社員マスタ_IN C6)

Microsoft Excel - テストブック.xls								
社員マスタ_INTG... ▾								
	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5	削除	社員コード	社員名	社員名カナ	性別	生年月日	郵便番号	
6								
7								
8								
9								
10								
11								
12								

3. 編集シート に、データを読み込みます。

・タスク定義

	A	B	C	D
1	タイトル	サブ	*	内容
2	TASKNO		*1	
3	タスク名		*TG_T_1	社員マスタ_Read
4	処理順NO		*3	
5	タスクタイプNO		*201	
6	タスクタイプ名		*SQL	
7	タスクタイプ属性		*照会	
8	BKNO		*1	
9	ブック名		*テストブック.xls	
10	シート名		社員マスタ	
11	QNO		2	
12	クエリー名		TG_社員マスタ_E	
13	DNO		2	

・クエリー定義

	A	B	C	D
1	タイトル	サブ	*	内容
2	クエリー名		*TG_社員マスタ_E	
3	クエリータイプ		*照会	
4	クエリー属性		*Query	
5	DBNO		*1	
6	エイリアス名		SampleDB	
7	テーブル名			
8	BKNO	旧仕様	0	
9	ブック名	旧仕様		
10	取得データ件数セル	旧仕様		Select [社員コード][社員名][社員名カナ][性別][生年月日][郵便番号][住所1][住所2][TEL][FAX] from[社員マスタ]
11	SQL文			
12	掛他制御タイプ			
13	定義有効無効フラグ			
14	コメント欄			

・展開定義

	A	B	C	D
1	タイトル	サブ	*	内容
2	DNO		*2	
3	データ展開名		TG_D2	社員マスタ_Q_Edit
4	処理順NO		1	
5	定義区分		クエリー定義	
6	タイプ		照会	
7	タスク・展開属性		Query	
8	データ項目名			
9	数式OR値			
10	定義有効無効フラグ			
11	コメント欄			
12	シート名		社員マスタ	
13	セル指定属性		直接	
14	セル範囲		@社員マスタ_TG_Q2_E	
15	セル展開範囲定義名			
16	シート保護有無			
17	シート保護解除PW			
18	自動計算方法			
19	行列方向	データ行列方向		
20	更新行OR列数指示セル	取得データ件数セル		

・Excel ブック

名前定義 社員マスタ_TG_Q2_E (社員マスタ C6)

Microsoft Excel - テストブック.xls							
ファイル(F)	編集(E)	表示(V)	挿入(I)	書式(O)	ツール(T)	データ(D)	ウィンドウ(W)
... ファイル	編集	表示	挿入	書式	ツール	データ	ウィンドウ
... ファイル	編集	表示	挿入	書式	ツール	データ	ウィンドウ
社員マスタTG_Q2...							
	A	B	C	D	E	F	G
1							
2							
3							
4							
5	削除	社員コード	社員名	社員名カナ	性別	生年月日	郵便番号
6							
7							
8							
9							
10							
11							
12							

4. 再計算を実行します。

	A	日	C	D
	タイトル	サブ	内容	
1	TASKNO	*	*1	
2	タスク名		*TG.T.1社員マスター.Read	
3	処理順NO		*4	
4	タスクタイプNO		*417	
5	タスクタイプ名		*自動計算	
6	タスクタイプ属性		再計算実行	
7	BKNO		1	
8	ブック名		テストブック.xls	
9	処理条件判定セル	実行判定		
10	定義有効無効フラグ			
11	コメント欄			
12	セル指定属性			
13	汎用パラメータ1	再計算対象		
14	汎用パラメータ2	再計算シート		
15				

5. 書式をコピーします。

・タスク定義

A	B	C	D
1 タイトル	サブ	*	内容
2 TASKNO		*1	
3 タスク名		*TG_T_1 社員マスタ_Read	
4 処理順NO		*5	
5 タスクタイプNO		*410	
6 タスクタイプ名		*コピー	書式・コメント・入力規則・列幅
7 タスクタイプ属性			
8 BKNO		*1	
9 ブック名		*テストブック.xls	
10 シート名			
11 DNO		5	
12 処理条件判定セル	実行判定		
13 定義有効無効フラグ			
14 コメント欄			
15 セル指定属性			
16 汎用パラメータ1	空白セル無視		
17 汎用パラメータ2	行列入替		
18 汎用パラメータ3	ブック間コピー		

・展開定義

A	B	C	D
1 タイトル	サブ	*	内容
2 DNO		*5	
3 データ展開名		TG_D5_E_Copy	
4 処理順NO		1	
5 定義区分		タスク定義	
6 タイプ		コピー	
7 タスク・展開属性		*書式・コメント・入力規則・列幅	
8 定義有効無効フラグ			
9 コメント欄			
10 シート名	社員マスタ		
11 セル指定属性	直接		
12 セル範囲	*@社員マスタTG_T1_OrgLine_E		
13 シート保護有無			
14 シート保護解除PW			
15 自動計算方法			
16 行列方向			
17 先シート名	社員マスタ		
18 先セル指定属性	間接		
19 先セル範囲	*@社員マスタTG_T1_CopyArea_E		
20 先シート保護有無			
21 先シート保護解除PW			

・Excel ブック

名前定義 社員マスタ_TG_T1_OrgLine_E (社員マスタ B3:L3)

Microsoft Excel - テストブック.xls													
社員マスタTG_T1...													
A	B	C	D	E	F	G	H	I	J	K	L	M	
1													
2													
3													※この範囲
4													
5	削除	社員コード	社員名	社員名カナ	性別	生年月日	郵便番号	住所1	住所2	TEL	FAX		
6													
7													
8													
9													
10													
11													
12													

書式をコピーするための、コピー元の領域です。

名前定義 社員マスタ_TG_T1_CopyArea_E (社員マスタ P12)

Microsoft Excel - テストブック.xls															
社員マスタTG_T1...															
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1															
2															
3															※この範囲はシステムが利用します。ユーザーに対して
4															
5	削除	社員コード	社員名	社員名カナ	性別	生年月日	郵便番号	住所1	住所2	TEL	FAX		入力開始行	入力開始列	入力列数
6													6	2	11
7															
8															
9															
10															
11															
12															

書式をコピーするための、コピー先の領域です。

読み込んだデータ量に応じて領域は変化します。

8.2.3 更新用タスク TG_T2_Write

「リスト型」の説明

1. 更新する行がなければ、エラーメッセージを表示し、更新を中断します。

・タスク定義

	A	B	C	D
1	タイトル	サブ	*	内容
2	TASKNO		*	2
3	タスク名		*	TG_T_2社員マスタ_Write
4	処理順NO		*	1
5	タスクタイプNO		*	318
6	タスクタイプ名		*	メッセージ表示
7	タスクタイプ属性		*	エラー判定
8	BKNO		*	1
9	ブック名		*	テストブック.xls
10	シート名		*	社員マスタ_OUT
11	処理条件判定セル	実行判定		
12	定義有効無効フラグ			
13	コメント欄			
14	セル指定属性			
15	汎用パラメータ1	メッセージ1		更新できる行が存在していません。
16	汎用パラメータ2	メッセージ2		
17	汎用パラメータ3	メッセージ3		
18	汎用パラメータ4	メッセージ4		
19	汎用パラメータ5	メッセージ5		
20	汎用パラメータ6	メッセージ6		
21	汎用パラメータ7	メッセージ7		
22	汎用パラメータ8	メッセージ8		
23	汎用パラメータ9	メッセージ9		
24	汎用パラメータ10	メッセージ10		
25	汎用パラメータ11	自動計算		
26	開始判定セル	メッセージNO	*	@社員マスタ_OUT_TG_T2_UpOK

・Excel ブック

名前定義 社員マスタ_OUT_TG_T2_UpOK (社員マスタ AA9)

Microsoft Excel - テストブック.xls												
P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB
1												
2												
3												
4												
5	DBS_UPDATE_USER	DBS_UPDATE_DATE	In結合	Out結合	更新判定	削除判定	既式開始行	既式開始列	既式列数			
6										6	2	
7											23	
8												
9												
10												
11												
12												

更新行数が 0 ならば、値を 1 にします。

値が 1 のとき、タスクは処理を中断します。

2. 数式をコピーします。(編集シートのデータを参照するなど)

・タスク定義

A	B	C	D
タイトル	サブ	内容	
1 TASKNO	*	*2	
2 タスク名		* TG.T_2社員マスタWrite	
3 処理順NO		*2	
5 タスクタイプNO		* 410	
6 タスクタイプ名		* コピー	
7 タスクタイプ属性		数式	
8 BKNO		*1	
9 ブック名		* テストブック.xls	
10 シート名		6	
11 DNO	実行判定		
12 処理条件判定セル			
13 定義有効無効フラグ			
14 コメント欄			
15 セル指定属性			
16 汎用パラメータ1	空白セル無視		
17 汎用パラメータ2	行列入替		
18 汎用パラメータ3	ブック間コピー		
19 開始時メッセージセル			
20 開始時メッセージ		データベースへの更新を行います。	
21			

・展開定義

A	B	C	D
タイトル	サブ	内容	
1 DNO	*	6	
2 データ展開名		TG.D6社員マスタ_O_Copy	
3 定義順NO		;	
5 定義区分		タスク定義	
6 タイプ		コピー	
7 タスク・展開属性		数式	
8 定義有効無効フラグ			
9 コメント欄			
10 シート名		社員マスタ_OUT	
11 セル指定属性		直接	
12 セル範囲		* @社員マスタ_OUT_TG.T2_OrgLine_O	
13 シート保護有無			
14 シート保護解除PW			
15 自動計算方法			
16 行列方向			
17 先シート名		社員マスタ_OUT	
18 先セル指定属性		間接	
19 先セル範囲		* @社員マスタ_OUT_TG.T2_CopyArea_O	
20 先シート保護有無			
21 先シート保護解除PW			

・Excel ブック

名前定義 社員マスタ_OUT_TG_T2_OrgLine_O (社員マスタ_OUT \$B\$3:\$AA\$3)

Microsoft Excel - テストブック.xls												
社員マスタ.OUTT... # =Z12&"."&AA12												
P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB
1							FALSE	FALSE				
2												
3												
4												
5 DBS_UPDATE_USER	DBS_UPDATE_DATE	In結合	Out結合			更新判定	削除判定			数式開始行	数式開始列	数式列数
6										6	2	23
7												
8												
9												
10												
11												
12												
13												
14												
15												

数式をコピーするための、コピー元の領域です。

名前定義 社員マスタ_OUT_TG_T2_CopyArea_O (社員マスタ_OUT AB12)

Microsoft Excel - テストブック.xls												
社員マスタ.OUTT... # =Z12&"."&AA12												
P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB
4												
5 DBS_UPDATE_USER	DBS_UPDATE_DATE	In結合	Out結合			更新判定	削除判定			数式開始行	数式開始列	数式列数
6										6	2	23
7												
8												
9												
10												
11												
12												
13												
14												
15												

\$B\$6:\$AA\$6 数式をコピーするための、コピー先の領域です。

読み込んだデータ量に応じて領域は変化します。

3. 再計算を実行します。

	A	B	C	D
	タイトル	サブ	*	内容
1			*	
2	TASKNO		*2	
3	タスク名		*TG_T_2社員マスタ_Write	
4	処理順NO		*3	
5	タスクタイプNO		*417	
6	タスクタイプ名		*自動計算	
7	タスクタイプ属性		再計算実行	
8	BKNO		1	
9	ブック名		テストブック.xls	
10	処理条件判定セル	実行判定		
11	定義有効無効フラグ			
12	コメント欄			
13	セル指定属性			
14	汎用パラメータ1	再計算対象		
15	汎用パラメータ2	再計算シート		

4. データを更新します。

・タスク定義

A	B	C	D
タイトル	サブ	内容	
1 TASKNO	*	*2	
3 タスク名		*TG_T_2社員マスタ_Write	
4 処理順NO		*4	
5 タスクタイプNO		*201	
6 タスクタイプ名		*SQL	
7 タスクタイプ属性		*指定更新	
8 BKNO		*1	
9 フック名		*テストブック.xls	
10 シート名		社員マスタ	
11 QNO		3	
12 クエリーネーム		TG_社員マスタ_O	
13 DNO		3	
14 処理条件判定セル	実行判定		
15 定義有効無効フラグ			
16 コメント欄			
17 汎用パラメータ20	自動計算		
18 開始時メッセージセル			
19 開始時メッセージ			
20 正常終了時メッセージセル			
21 正常終了時メッセージ		更新処理が正常に終了しました。	

・クエリーディレクティブ

A	B	C	D
タイトル	サブ	内容	
1 クエリーネーム	*	*TG_社員マスタ_O	
2 クエリータイプ		*更新	
3 クエリーディレクティブ		*Query	
4 クエリーディレクティブ属性		*1	
5 DBNO		SampleDB	
6 エイリアス名		*社員マスタ	
7 テーブル名		1	
8 BKNO		テストブック.xls	
9 フック名		UpDate	
10 SQL文			
11 トランザクション処理			
12 排他制御タイプ			
13 更新処理形態		修正データのみ	
14 定義有効無効フラグ			
15 コメント欄			
16 悲観ロック解除QNO			
17 日付時刻NULL指定	NULL指定	NULL	

・展開定義

A	B	C	D
タイトル	サブ	内容	
1 DNO	*	*3	
2 データ展開名		TG_D3社員マスタ_Q_Out	
3 処理順NO		1	
4 定義区分		クエリーディレクティブ	
5 タイプ		Query	
6 タスク・展開属性			
7 データ項目名			
8 数式OR値			
9 定義有効無効フラグ			
10 コメント欄			
11 シート名	更新専用シート名	社員マスタ_OUT	
12 セル指定属性		直接	
13 セル範囲	データ更新開始位置	@社員マスタ_OUT_TG_Q3_O	
14 セル展開範囲定義名			
15 シート保護有無			
16 シート保護解除PW			
17 自動計算方法			
18 行列方向	データ行列方向		
19 更新行OR列数指示セル		@社員マスタ_OUT_TG_Q3_UpCount	

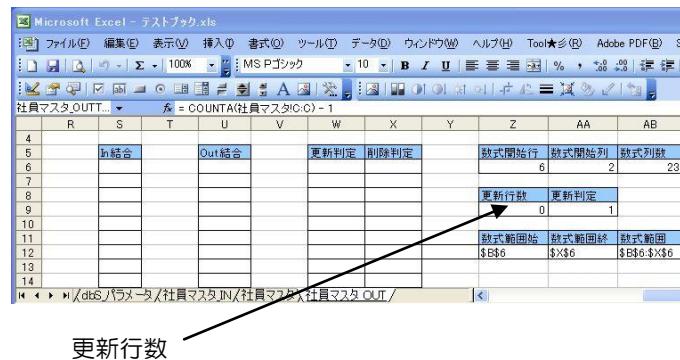
・Excel ブック

名前定義 社員マスタ_OUT_TG_Q3_O (社員マスタ_OUT C6)

Microsoft Excel - テストブック.xls										
ファイル	編集	表示	挿入	書式	ツール	データ	ウィンドウ	ヘルプ	100%	MS Paint
社員マスタOUT... ▾	セル	セル	セル	セル	セル	セル	セル	セル	セル	セル
	A	B	C	D	E	F	G	H	I	J
	4	削除	社員コード	社員名	社員名カナ	性別	生年月日	郵便番号	住所1	住所2
	5									
	6									
	7									
	8									
	9									
	10									
	11									
	12									
	13									
	14									
	15									

更新開始位置

名前定義 社員マスター_OUT_TG_Q3_UpCount (社員マスター_OUT_Z6)



5. 再度、データを読み込みます。

	A	B	C	D
1	タイトル	サブ	*	内容
2	TASKNO		*	2
3	タスク名		*	TG_T_2社員マスター_Write
4	処理順NO		*	5
5	タスクタイプNO		*	320
6	タスクタイプ名		*	サブタスク実行
7	EIKNO		*	1
8	ブック名		*	テストブック.xls
9	シート名			
10	処理条件判定セル	実行判定		
11	定義有効無効フラグ			
12	コメント欄			
13	TASKJUMPN01	サブタスク1	1	
14	TASKJUMPN02	サブタスク2	0	
15	TASKJUMPN03	サブタスク3	0	
16	TASKJUMPN04	サブタスク4	0	
17	TASKJUMPN05	サブタスク5	0	
18	TASKJUMPN06	サブタスク6	0	
19	TASKJUMPN07	サブタスク7	0	
20	TASKJUMPN08	サブタスク8	0	
21	TASKJUMPN09	サブタスク9	0	
22	TASKJUMPN010	サブタスク10	0	

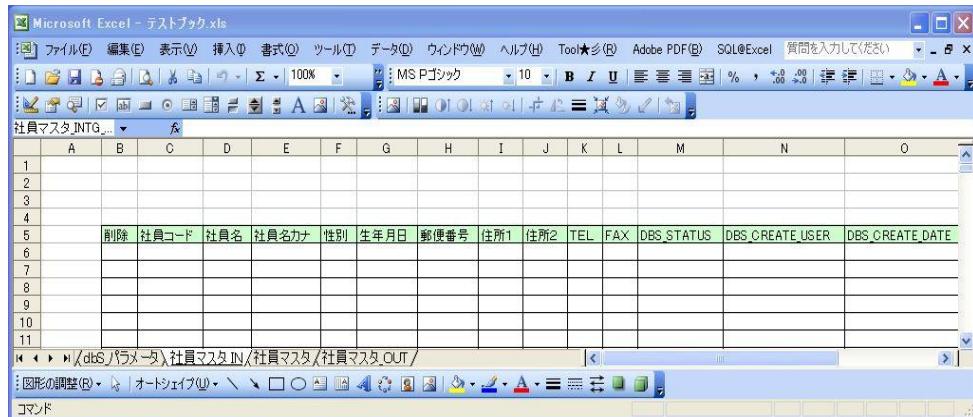
8.2.4 作成されるシート情報

Excel ブック に、以下「リスト型」の3つのシートが追加されます。

- Inシート
 - 編集シート
 - Outシート

◆ レイアウト

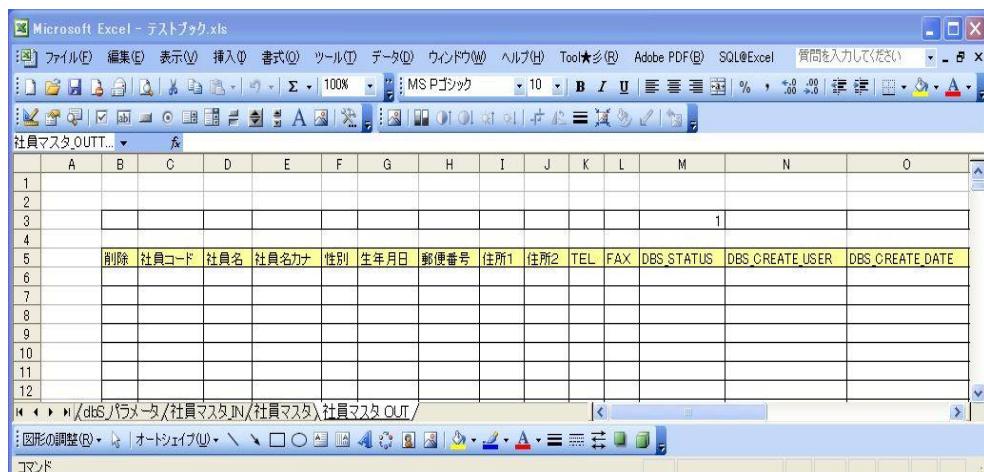
- ### ・Inシート



- 編集シート



- ### ・Outシート



◆ 名前の定義

シート	名前	参照範囲
編集シート	dbs_社員マスタ	=社員マスタ!\$B\$6:\$L\$6
Inシート	社員マスタ_IN_TG_Q1_I	=社員マスタ_IN!\$C\$6
編集シート	社員マスタ_TG_Q2_E	=社員マスタ!\$C\$6
Outシート	社員マスタ _OUT_TG_Q3_O	=社員マスタ_OUT!\$C\$6
Outシート	社員マスタ _OUT_TG_Q3_UpCount	=社員マスタ_OUT!\$Z\$9
編集シート	社員マスタ _TG_T1_CopyArea_E	=社員マスタ!\$P\$12
編集シート	社員マスタ _TG_T1_OrgLine_E	=社員マスタ!\$B\$3:\$L\$3
Outシート	社員マスタ _OUT_TG_T2_CopyArea_O	=社員マスタ_OUT!\$AB\$12
Outシート	社員マスタ _OUT_TG_T2_OrgLine_O	=社員マスタ_OUT!\$B\$3:\$X\$3
Outシート	社員マスタ _OUT_TG_Q3_UpCount	=社員マスタ_OUT!\$Z\$9

◆ リストの定義

• 編集シート

	A	B	C	D	E	F	G
1							
2							
3							
4		削除					
5	削除	社員コード	社員名	社員名力ナ	性別	生年月日	
6							
7							
8							
9							
10							
11							
12							

◆ 数式の定義

• 編集シート

役割	アドレス	数式
入力開始行	N6	6
入力開始列	O6	2
入力列数	P6	11
読込行数	N9	=ROWS(dbs_社員マスタ)
空白行数	O9	100
入力範囲始	N12	=ADDRESS(N6,O6)
入力範囲終	O12	=ADDRESS(N6+N9+O9,O6+P6-1)
入力範囲	P12	=N12&":"&O12

• Out シート

役割	アドレス	数式
編集シート参照	B3~L3	=IF(社員マスタ!B3="","",社員マスタ!B3)
ステータス判定	M3	=IF(X3,99,IF(W3,2,1))
INシートデータ結合	S3	=社員マスタ_IN!C3&":"&社員マスタ_IN!D3&":"&社員マスタ_IN!E3&":"&社員マスタ_IN!F3&":"&社員マスタ_IN!G3&":"&社員マスタ_IN!H3&":"&社員マスタ_IN!I3&":"&社員マスタ_IN!J3&":"&社員マスタ_IN!K3&":"&社員マスタ_IN!L3
Out 結合	U3	=C3&":"&D3&":"&E3&":"&F3&":"&G3&":"&H3&":"&I3&":"&J3&":"&K3&":"&L3
更新判定	W3	=AND(B3="削除",S3<>U3)
削除判定	X3	=AND(B3="削除",社員マスタ_IN!M3<>")
数式開始行	Z6	6
数式開始列	AA6	2
数式列数	AB6	23
更新行数	Z9	=COUNTA(社員マスタ!C:C) - 1
更新判定	AA9	=IF(Z9=0,1,0)
数式範囲始	Z12	=ADDRESS(Z6,AA6)
更新範囲終	AA12	=ADDRESS(Z6+Z9,AA6+AB6-1)
更新範囲	AB12	=Z12&":"&AA12

8.2.5 作成されるプロジェクト定義情報

プロジェクトの定義情報に、以下「カード型の」情報が追加されます。

- ◆ メニュー定義 1行
- ◆ ボタン定義 2行
- ◆ タスク定義 2行
- ◆ クエリー定義 2行
- ◆ 展開定義 15行

◆ メニュー定義



The screenshot shows the dbSheetClient interface for menu definition. The menu bar at the top says 'dbSheetClient 開発版 テクニカルリファレンスマニュアル'. The main window title is '【135】プロジェクト:142【開発】72_IOTGテスト #2' and the sub-tab is '★メニュー定義★'. The toolbar includes buttons for '終了' (Exit), '実行' (Run), 'メニュー' (Menu), 'ボタン' (Button), 'タスク' (Task), 'クエリ' (Query), '展開' (Expand), 'EXCEL', 'SDB', 'GDB', 'バック', 'バック運用', '開始終了', 'ボタン一覧', 'タスク一覧', '展開一覧', 'フォント' (Font), and 'HELP'. The menu structure on the left shows 'メニュー' and '社員マスタ'. The main table on the right lists menu items with columns for 'タイトル' (Title), 'サブ' (Sub), and '内容' (Content). The new item '社員マスタ' is highlighted in green.

A	B	C	D	E	F	G	H	I	J
タイトル	サブ	*	内容						
表示NO	階層	メニュー名	BKNO	ブック名	シート名				
1	1	1	社員マスタ	1	TESTA.xls	社員マ			
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									

メニューに、**社員マスタ** が追加されています。

◆ ボタン定義



The screenshot shows the dbSheetClient interface for button definition. The menu bar at the top says 'dbSheetClient 開発版 テクニカルリファレンスマニュアル'. The main window title is '【135】プロジェクト:142【開発】72_IOTGテスト #2' and the sub-tab is '★ボタン定義★'. The toolbar includes buttons for '終了' (Exit), '実行' (Run), 'メニュー' (Menu), 'ボタン' (Button), 'タスク' (Task), 'クエリ' (Query), '展開' (Expand), 'EXCEL', 'SDB', 'GDB', 'バック', 'バック運用', '開始終了', 'ボタン一覧', 'タスク一覧', '展開一覧', 'フォント' (Font), and 'HELP'. The menu structure on the left shows 'メニュー' and 'ボタン'. The main table on the right lists buttons with columns for 'ボタン名' (Button Name), '表示名' (Display Name), 'TASKNO', 'タスク名' (Task Name), '権限レベル' (Permission Level), '定義有効無効フラグ' (Definition Valid/Invalid Flag), 'ガイダンス' (Guidance), 'コメント欄' (Comment), and 'ボタンデザイン' (Button Design). The new button set 'TG_BS1' is highlighted in green.

A	B	C	D	E	F	G	H	I	J
ボタン名	表示名	TASKNO	タスク名	権限レベル	定義有効無効フラグ	ガイダンス	コメント欄	ボタンデザイン	ボタン
BSNO	ボタンNO	1	読み込	1	TG_T1_Read				
2	1	1	更新	2	TG_T2_Write				
3									
4									
5									
6									
7									

ボタンセット **TG_BS1** が追加されています。

読み込 ボタンと、**更新** ボタンが追加されています。

◆ タスク定義



The screenshot shows the dbSheetClient Task Definition window. The left pane displays a tree structure of tasks under 'TG_BS1' (e.g., TG_T1_Read, TG_T2_Write). The right pane is a grid table with the following data:

B	C	D	E	F	G	H	I	J
TASKNO	タスク名	処理順NO	タスクタイプNO	タスクタイプ名	タスクタイプ属性	BKNO	ブック名	シート名
1	TG_T1_Read	1	203	データ照会画面		1	TESTA.xls	社員マスター
2	TG_T2_Write	1	201	SQL	指定更新	1	TESTA.xls	社員マスター_OUT
3								
4								
5								
6								
7								
8								
9								

2つのタスクが追加されています。

• 読込用タスク `TG_T1_Read`

1. 編集シート に、データを読み込みます。

• 更新用タスク `TG_T2_Write`

1. データを更新します。

8.2.6 読込用タスク TG_T_1 社員マスター_Read

「カード型」の説明

1. 編集シート に、データを読み込みます。

・タスク定義

	A	日	C	D
1	タイトル	サブ	*	内容
2	TASKNO		*	
3	タスク名		*	TG_T_1 社員マスター_Read
4	処理順NO		*	1
5	タスクタイプNO		*	203
6	タスクタイプ名		*	データ照会画面
7	タスクタイプ属性			
8	BKNO		*	
9	ブック名		*	テストブック.xls
10	シート名			社員マスター
11	QNO		*	
12	クエリーネーム			TG_社員マスター_E
13	DNO		*	2

・クエリーネーム

	A	日	C	D
1	タイトル	サブ	*	
2	クエリーネーム		*	TG_社員マスター_E
3	クエリータイプ		*	照会
4	クエリーワイド		*	Query
5	DBNO		*	1
6	エイリアス名			SampleDB
7	テーブル名			
8	BKNO	旧仕様	0	
9	ブック名	旧仕様		
10	取得データ件数セル	旧仕様		Select [社員コード],[社員名],[社員名カナ],[性別],[生年月日],[郵便番号],[住所1],[住所2],[TEL],[FAX]
11	SQL文			
12	排他制御タイプ			
13	定義有効無効フラグ			
14	コメント欄			

・展開定義

日	データ展開名	処理順NO	定義区分	タイプ	データ展開	データ項目名	シート名	レジスト	セル範囲	セル展開範囲
1	DNO									
2	2 TG_D2社員マスター_Q.Edit	1	クエリーワイド	照会	データ	社員コード	社員マスター	直接	@社員マスター_社員コード.TG_1_E_0	
3	2 TG_D2社員マスター_Q.Edit	2	クエリーワイド	照会	データ	社員名	社員マスター	直接	@社員マスター_社員名.TG_1_E_1	
4	2 TG_D2社員マスター_Q.Edit	3	クエリーワイド	照会	データ	社員名カナ	社員マスター	直接	@社員マスター_社員名カナ.TG_1_E_2	
5	2 TG_D2社員マスター_Q.Edit	4	クエリーワイド	照会	データ	性別	社員マスター	直接	@社員マスター_性別.TG_1_E_3	
6	2 TG_D2社員マスター_Q.Edit	5	クエリーワイド	照会	データ	生年月日	社員マスター	直接	@社員マスター_生年月日.TG_1_E_4	
7	2 TG_D2社員マスター_Q.Edit	6	クエリーワイド	照会	データ	郵便番号	社員マスター	直接	@社員マスター_郵便番号.TG_1_E_5	
8	2 TG_D2社員マスター_Q.Edit	7	クエリーワイド	照会	データ	住所1	社員マスター	直接	@社員マスター_住所1.TG_1_E_6	
9	2 TG_D2社員マスター_Q.Edit	8	クエリーワイド	照会	データ	住所2	社員マスター	直接	@社員マスター_住所2.TG_1_E_7	
10	2 TG_D2社員マスター_Q.Edit	9	クエリーワイド	照会	データ	TEL	社員マスター	直接	@社員マスター_TEL.TG_1_E_8	
11	2 TG_D2社員マスター_Q.Edit	10	クエリーワイド	照会	データ	FAX	社員マスター	直接	@社員マスター_FAX.TG_1_E_9	
12	2 TG_D2社員マスター_Q.Edit	11	クエリーワイド	照会	データ	DBS_CREATE_USER	社員マスター_OUT	直接	@社員マスター_OUT_DBS_CREATE_USER.TG_1_E_11	
13	2 TG_D2社員マスター_Q.Edit	12	クエリーワイド	照会	データ	DBS_CREATE_DATE	社員マスター_OUT	直接	@社員マスター_OUT_DBS_CREATE_DATE.TG_1_E_12	
14	2 TG_D2社員マスター_Q.Edit	13	クエリーワイド	照会	データ	DBS_UPDATE_USER	社員マスター_OUT	直接	@社員マスター_OUT_DBS_UPDATE_USER.TG_1_E_13	
15	2 TG_D2社員マスター_Q.Edit	14	クエリーワイド	照会	データ	DBS_UPDATE_DATE	社員マスター_OUT	直接	@社員マスター_OUT_DBS_UPDATE_DATE.TG_1_E_14	

・Excel ブック

名前定義 社員マスター_社員コード.TG_1_E_0 (社員マスター C3)

Microsoft Excel - TEST.xls				
日	データ展開名	処理順NO	定義区分	タイプ
1				
2	削除			
3	社員コード			
4	社員名			
5	社員名カナ			
6	性別			
7	生年月日			
8	郵便番号			
9	住所1			
10	住所2			
11	TEL			
12	FAX			
13				
14				

8.2.7 更新用タスク TG_T_2 社員マスタ_Write

「カード型」の説明

1. データを更新します。

・タスク定義

A	日	C	D
タイトル	サブ	内容	
1 TASKNO	*	*2	
3 タスク名		TG_T_2社員マスタ_Write	
4 処理順NO		*	
5 タスクタイプNO		*201	
6 タスクタイプ名		*SQL	
7 タスクタイプ属性		*指定更新	
8 BKNO		*1	
9 ブック名		*テストブック.xls	
10 シート名		社員マスタ_OUT	
11 QNO		1	
12 クエリー名		TG_社員マスタ_O	
13 DNO		1	
14 処理条件判定セル	実行判定		
15 定義有効無効フラグ			
16 コメント欄			
17 汎用パラメータ20	自動計算		
18 開始時メッセージセル		データベースへの更新を行います。	
19 開始時メッセージ			
20 正常終了時メッセージセル			
21 正常終了時メッセージ		更新処理が正常に終了しました。	

・クエリー定義

A	日	C	D
タイトル	サブ	内容	
1 クエリー名	*	TG_社員マスタ_O	
2 クエリータイプ		*指定更新	
3 クエリー属性		*Query	
4 クエリー属性		*1	
5 DBNO		SampleDB	
6 エイリアス名		*社員マスタ	
7 テーブル名		1	
8 BKNO		TEST.xls	
9 ブック名		UpDate	
10 SQL文			
11 トランザクション処理			
12 排他制御タイプ			
13 更新処理形態		修正データのみ	
14 定義有効無効フラグ			
15 コメント欄			
16 悲観ロック解除QNO			
17 日付時刻NULL指定 NULL指定		NULL	

・展開定義

A	日	C	D
タイトル	サブ	内容	
1 DNO	*	*1	
2 データ展開名		TG_D1_社員マスタ_Q_Out	
3 処理順NO		1	
4 定義区分		クエリー定義	
5 タイプ		*指定更新	
6 タスク・展開属性		Query	
7 データ項目名			
8 数式OR値			
9 定義有効無効フラグ			
10 コメント欄			
11 シート名	更新専用シート名	社員マスタ_OUT	
12 セル指定属性		直接	
13 セル範囲	データ更新開始位置	@社員マスタ_OUTTG_2_O	
14 セル展開範囲定義名			
15 シート保護有無			
16 シート保護解除PW			
17 自動計算方法			
18 行列方向	データ行列方向		
19 更新行OR列数指示セル		1	

・Excel ブック

名前定義 社員マスタ_OUTTG_2_O (社員マスタ_OUT B3)

Microsoft Excel - TEST.xls															
ファイル	編集	表示	挿入	書式	ツール	データ	ウンドウ	ヘルプ	Adobe PDF	質問を入力してください	□	×	×	×	×
社員マスタ.OUT... ▾															
1															
2	社員コード	社員名	社員名カナ	性別	生年月日	郵便番号	住所1	住所2	TEL	FAX	DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER	DBS_UPDA
3															
4															
5															
6															
7															

↑
更新開始位置

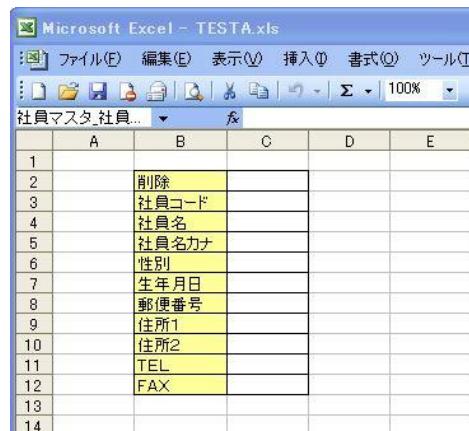
8.2.8 作成されるシート情報

Excel ブック に、以下「カード型」の2つのシートが追加されます。

- ・編集シート
- ・Out シート

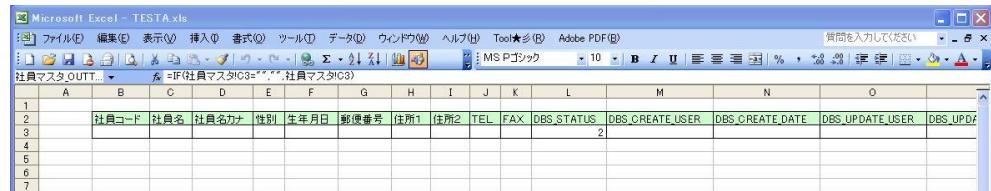
◆ レイアウト

- ・編集シート



	A	B	C	D	E
1					
2		削除			
3		社員コード			
4		社員名			
5		社員名カナ			
6		性別			
7		生年月日			
8		郵便番号			
9		住所1			
10		住所2			
11		TEL			
12		FAX			
13					
14					

- ・Out シート



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1															
2		社員コード	社員名	社員名カナ	性別	生年月日	郵便番号	住所1	住所2	TEL	FAX	DBS_STATUS	DBS_CREATE_USER	DBS_CREATE_DATE	DBS_UPDATE_USER
3												2			
4															
5															
6															
7															

◆ 名前の定義

シート	名前	参照範囲
編集シート	社員マスタ_社員コード_TG_1_E_0	=社員マスタ!C3
編集シート	社員マスタ_社員名_TG_1_E_1	=社員マスタ!C4
編集シート	社員マスタ_社員名カナ_TG_1_E_2	=社員マスタ!C5
編集シート	社員マスタ_性別_TG_1_E_3	=社員マスタ!C6
編集シート	社員マスタ_生年月日_TG_1_E_4	=社員マスタ!C7
編集シート	社員マスタ_郵便番号_TG_1_E_5	=社員マスタ!C8
編集シート	社員マスタ_住所1_TG_1_E_6	=社員マスタ!C9
編集シート	社員マスタ_住所2_TG_1_E_7	=社員マスタ!C10
編集シート	社員マスタ_TEL_TG_1_E_8	=社員マスタ!C11
編集シート	社員マスタ_FAX_TG_1_E_9	=社員マスタ!C12
Outシート	社員マスタ_OUTTG_2_O	=社員マスタ_OUT!\$B\$3

◆ 数式の定義

・Outシート

役割	アドレス	数式
編集シート参照	B3~P3	=IF(社員マスタ!C3="","",社員マスタ!C3)

8.3 読込みのみの場合の定義シートの作成手順

「リスト型」での説明

8.3.1 更新機能の削除

1) 「更新」ボタン を削除

ボタンセットから「更新」ボタンを削除します。

「ツリー表示エリア」の「TG_BS1_社員マスタ」をクリックします。

「定義編集エリア」の「更新」を選びます。

【行削除】ボタンをクリックします。行の背景色が赤になります。



【保存】ボタンをクリックします。

「ツリー表示エリア」の「更新」ボタンと、「定義編集エリア」の「更新」行がなくなります。



2) 「更新」タスクを削除

タスク定義から「更新」タスクを削除します。

「ツリー表示エリア」の「一覧表示」タブをクリックします。



「ツリー表示エリア」の「T_G_T_2社員マスターWrite」をクリックします。



「定義編集エリア」の「T_G_T_2社員マスターWrite」の5行を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。



[保存] ボタンをクリックします。

「ツリー表示エリア」の「更新」ボタンと、「定義編集エリア」の「更新」行がなくなります。



タスク一覧定義から「更新」タスクを削除します。

「ツリー表示エリア」の「タスク一覧」をクリックします。



「定義編集エリア」の「TG_T_2 社員マスタ_Write」を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。



[保存] ボタンをクリックします。

「ツリー表示エリア」の「TG_T_2 社員マスタ_Write」と、「定義編集エリア」の「TG_T_2 社員マスタ_Write」行がなくなります。



3) 「更新」展開 を削除

展開定義から「更新」展開を削除します。

全定義情報表示を、表示 **全** に切り替えます。

「ツリー表示エリア」の「TG_D3_社員マスタ_Q_Out」をクリックします。



「定義編集エリア」の「TG_D3_社員マスタ_Q_Out」を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。

「定義編集エリア」の「TG_D6_社員マスタ_O_Copy」を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。



[保存] ボタンをクリックします。「定義編集エリア」の2行がなくなります。



展開一覧定義から「更新」展開を削除します。

「ツリー表示エリア」の「展開一覧」をクリックします。

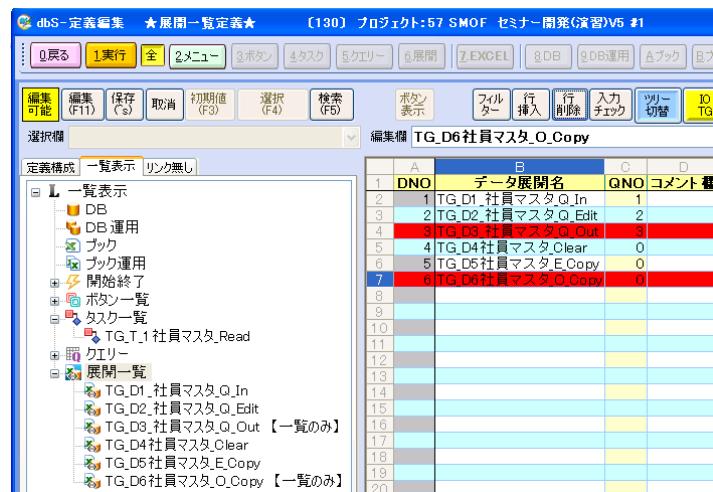


「定義編集エリア」の「TG_D3_社員マスタ_Q_Out」を選びます。

【行削除】ボタンをクリックします。行の背景色が赤に変わります。

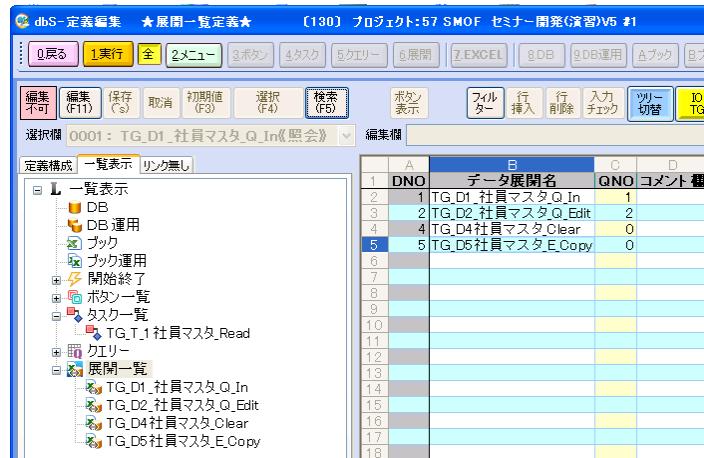
「定義編集エリア」の「TG_D6_社員マスタ_O_Copy」を選びます。

【行削除】ボタンをクリックします。行の背景色が赤に変わります。



【保存】ボタンをクリックします。

「ツリー表示エリア」と「定義編集エリア」から、2行分がなくなります。



4) 更新クエリー を削除

クエリー定義から「更新」クエリーを削除します。

「ツリー表示エリア」の「TG_社員マスタ_O」をクリックします。

A	B	C	D	E	F	G
QNO	クエリーネーム	クエリータイプ	クエリーアクション	DBNO	エイリアス名	テーブル名
1	TG_社員マスタ_I	照会	Query	1	SampleDB	
2	TG_社員マスタ_E	照会	Query	1	SampleDB	
3	TG_社員マスタ_O	更新	Query	1	SampleDB	社員マスタ
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						

「定義編集エリア」の「TG_社員マスタ_O」を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。

A	B	C	D	E	F	G
QNO	クエリーネーム	クエリータイプ	クエリーアクション	DBNO	エイリアス名	テーブル名
1	TG_社員マスタ_I	照会	Query	1	SampleDB	
2	TG_社員マスタ_E	照会	Query	1	SampleDB	
3	TG_社員マスタ_O	更新	Query	1	SampleDB	社員マスタ
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						

[保存] ボタンをクリックします。

「ツリー表示エリア」と「定義編集エリア」から、1行分がなくなります。

A	B	C	D	E	F	G
QNO	クエリーネーム	クエリータイプ	クエリーアクション	DBNO	エイリアス名	テーブル名
2	TG_社員マスタ_I	照会	Query	1	SampleDB	
3	TG_社員マスタ_E	照会	Query	1	SampleDB	

8.3.2 読込機能の変更

1) Inシートへの読込タスクを削除

タスク定義から「読込」タスクを削除します。

「ツリー表示エリア」の「TG_T_1 社員マスター Read」をクリックします。



「定義編集エリア」の「処理順 NO 2」を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。



[保存] ボタンをクリックします。「定義編集エリア」の1行がなくなります。



2) インシートへの読み込み展開を削除

展開定義から「読み込み」展開を削除します。

「ツリー表示エリア」の「TG_D1_社員マスター_Q_In」をクリックします。



「定義編集エリア」の「TG_D1_社員マスター_Q_In」を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。



[保存] ボタンをクリックします。「定義編集エリア」で、1行がなくなります。



展開一覧定義から「読み込」展開を削除します。

「ツリー表示エリア」の「展開一覧」をクリックします。



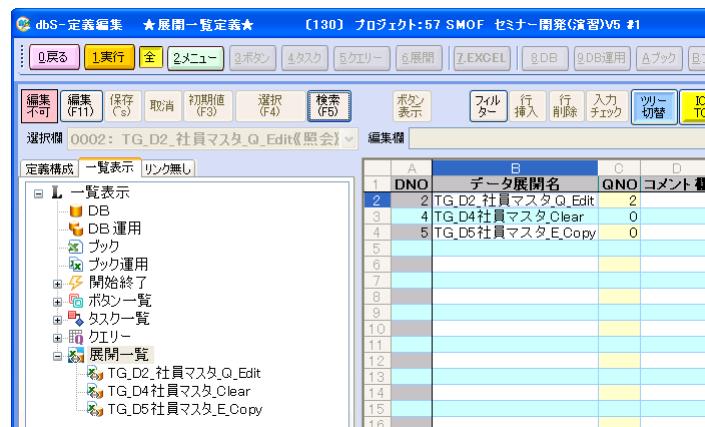
「定義編集エリア」の「TG_D1_Q_In」を選びます。

「行削除」ボタンをクリックします。行の背景色が赤に変わります。



「保存」ボタンをクリックします。

「ツリー表示エリア」と「定義編集エリア」から、1行分がなくなります。



3) タブシートへの読み込みクエリーを削除

クエリー定義から「読み込み」クエリーを削除します。

「ツリー表示エリア」の「TG_社員マスター」をクリックします。



「定義編集エリア」の「TG_社員マスター」を選びます。

[行削除] ボタンをクリックします。行の背景色が赤に変わります。

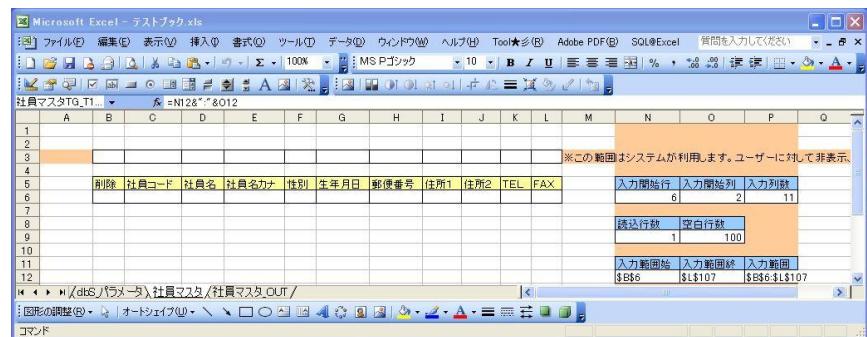


[保存] ボタンをクリックします。

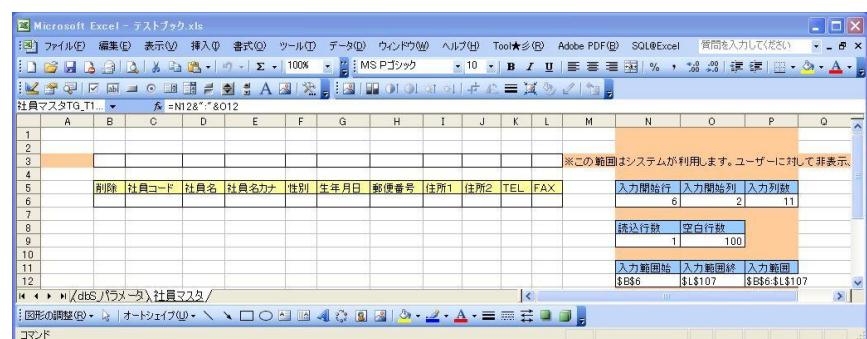
「ツリー表示エリア」と「定義編集エリア」から、1行分がなくなります。



4) In シート を削除



5) Out シート を削除



6) 動作確認

デバッグ実行 をクリックします。



「更新」ボタン が削除されています。

「読み込」ボタン をクリックします。



データを読み込むことができました。

第9章 制限事項とトラブルシューティング

本章では、dbSheetClient の制限事項やトラブルシューティングについて説明します。

9.1 制限事項

(1) システム全般の制限事項

No	機能分類	該当バージョン	制限事項	暫定対応＆補足事項
1	ActiveDirectory 対応	全バージョン	ActiveDirectory の管理下でご利用を予定している場合には、お客様のネットワーク環境や運用形態の調査のうえ、カスタマイズ等が必要になりますので、事前にお問い合わせください。	ActiveDirectory 管理下のグループとユーザを dbSheetClient のリソースとして取込ことができ、ActiveDirectory にログイン済みのユーザについては、dbSheetClient のログイン画面からのログイン・ユーザ認証を省略することができます。
2	ORACLE の対応	全バージョン	VISTA では、ローカル経由での ORACLE 接続は対応していません。VISTA の場合には、必ず WEB サーバ経由で接続する必要があります。XP からローカル経由での ORACLE 接続は可能です。	
3	Excel2003 の初期バージョン使用時の注意	全バージョン	Excel2003 のサービスパックの当たっていない、初期バージョンを使用している場合、タスクタイプ 410.コピーを実行したときにエラーが発生するケースがあります。	Excel2003 SP3をご利用ください。
4	Excel の制限事項	全バージョン	1 つのセルに入力できる文字数は最大 32767 文字ですが、表示および印刷可能な文字数は 1024 文字です。	http://support.microsoft.com/kb/211580/ja

(2) 実行版の制限事項

No	機能分類	該当バージョン	制限事項	暫定対応&補足事項
1	データ照会と更新のデータサイズについて	全バージョン	タスクタイプ「201、203、204、205、230」では一回の実行で照会や更新できるデータサイズの上限値は2GBで超えるとエラーになります。	
2	「DBS サーバ」へのファイルのアップロード時のファイルサイズ	全バージョン	「DBS サーバ」にアップロードできるファイルのサイズの上限は、Web.config で指定されたサイズまでです。 初期値としては32MBに設定しています。	上限サイズを変更する場合は、Web.Config を直接変更します。なお、最大サイズは2GBまで可能です。
3	「DBS サーバ」へのアクセス権限	全バージョン	dbSheetClient では「NT AUTHORITY\NETWORK SERVICE」の権限で、Web サービス経由により「DBS サーバ」内のファイルの上書きと、フォルダやファイルの作成/移動/削除をしています。このため、ファイルやフォルダには「NT AUTHORITY\NETWORK SERVICE」でアクセスできる権限が設定されている必要があります。	タスクタイプ 322.ファイルアップロード、341.サーバフォルダ削除、342.ファイル操作で制限が発生します。
4	Excel2000 の制限	全バージョン	Excel2000 で使用する際には以下の制限が発生します。 1) タスクタイプ 410.コピーで、タスクタイプ属性の⑨数式と数値の書式 ⑩値と数値の書式は使用することができません。 2) 大きなサイズの図を読み込む際にエラーになります。 3) ブック制御で参照用に開いた Excel ファイルを手動（「X（閉じる）」）でクリックした時、警告を表示するが、そのまま閉じてしまいます。	→1)他のタスクタイプ属性を利用してください。 →2)Excel2000 で表示できるサイズの図に変更してください。 →3)Excel2003 以上ならば閉じる操作はチャンセルにされます。
5	グラフシートの扱い	全バージョン	dbSheetClient では、グラフシート（チャートシート）の制御はできません。	Excel にグラフを作成する際に、ワークシート上に作成してください。
6	秒未満の時間の制限	全バージョン	日付時間型の場合、秒未満の値を Excel からデータベースに保存できない制限があります。	

【次ページに続く】

【前ページより続く】

No	機能分類	該当バージョン	制限事項	暫定対応&補足事項
7	URL でリンクされた外部 Excel の起動時の扱い	全バージョン	WindowsXP の環境下で dbSheetClient を実行中に、URL でリンクされている外部 EXCEL ファイルを起動すると、dbSheetClient の Excel エリア内に一旦そのブックが入り込み、その後、別の Excel を起動して、そこにブックを開きますが、その際に、dbSheetClient で編集中の Excel ブックの名称が変更されてしまったり、外部 EXCEL ブック自体の編集ができないくなる現象が発生することが確認できました。	現在、回避処理について、調査中です。
8	開発時の Excel が2003の場合	全バージョン	Excel2002 以下で動作しない関数などがあります。	必ず動作確認をし、動作しない場合は、別関数に切り替える。
9	開発時の Excel が2007の場合	全バージョン	Excel2003 以下で動作しない関数などがあります。	必ず動作確認をし、動作しない場合は、別関数に切り替える。
10	開発時の Excel が2007の場合(2)	全バージョン	Excel ファイルの拡張子が、"xlsx"で、Excel2003 以下の Excel を使用する場合、「Word/Excel/PowerPoint 2007 ファイル形式用 Microsoft Office 互換機能パック」が必要になります。	各クライアントに「 Word/Excel/PowerPoint 2007 ファイル形式用 Microsoft Office 互換機能パック 」をインストールしてください。
11	Excel のシート名について	全バージョン	シート名で「S1」などを使用した場合、EXCEL 側がセル位置と判断しますので、セルアドレスにない名前「XF1」より大きい名前を使用	Excel での制約
12	テキストボックスについて	Excel 2007	Excel2003 で作成した Excel レイアウト上にフォント8のテキストボックスを配置した場合 Excel2003 ではフォントサイズは 8 で表示されるが、Excel2007 ではフォントサイズが 11 となって表示される	Excel での制約
13	シート名について	全バージョン	"で括ったシート名は使用できません。このため、ADDRESS 関数で作成した空白を含むシート名のアドレスは間接指定で使用できません。	Excel での制約

(3) 開発版の制限事項

No	機能分類	該当バージョン	制限事項	暫定対応＆補足事項
1	DB 自動作成	全バージョン	Excel2007 形式で保存されたブックで DB 自動作成をおこなうことはできません。 Excel2003 形式で保存すると DB 自動作成をおこなうことができます。	
2	「DBS サーバ」へのアクセス権限	全バージョン	dbSheetClient では「NT AUTHORITY\NETWORK SERVICE」の権限で、Web サービス経由により「DBS サーバ」内のファイルの上書きと、フォルダやファイルの作成/移動/削除をしています。このため、ファイルやフォルダには「NT AUTHORITY\NETWORK SERVICE」でアクセスできる権限が設定されている必要があります。	開発メイン操作画面の「Excel アップロード」機能で制限が発生します。
3	「データ保守定義」の制限	全バージョン	データ保守の対象テーブルを指定する際、フィールド名に全角カッコ（“）が使われているテーブルはデータ保守定義における「リスト型」の対象外となります。 例）データベース “dbs_addressBook” の中のテーブル「住所録」に、全角カッコ（“）がフィールド名として、名前（姓）や名前（名）のように使用されている場合が該当します。	リスト型でデータ保守を定義する場合は、これらのフィールド名から全角カッコを、除外してテーブルを再定義する必要があります。 カード型で定義する場合には、制限はありません。

9.2 トラブルシューティング

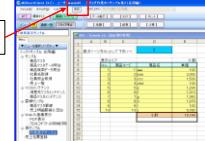
(1) システム全般のトラブルシューティング

No	機能分類	該当バージョン	現象	原因&対応方法
1	インストール時のエラー	全バージョン	クリックワنسの実行版をインストールしようとするとエラーになり、インストールすることができない。	<p>【原因】</p> <p>①ブラウザは IE6.0 以上のみ対応しています。マイクロソフト社以外のブラウザでは正常にインストールすることはできません。</p> <p>②ご使用の PC のセキュリティソフトで制限している場合と設定、セキュリティポリシーの設定で正常にインストールできないことがあります。解決できない場合には弊社までお問い合わせください。</p> <p>【対応方法】</p> <p>「.NetFramework2.0」を個別にインストールするとインストールできるようになります。</p>
2	ログイン時のエラー	全バージョン	PC を起動直後に、dbSheetClient を起動すると、サーバ接続チェック画面が表示されたまま動作しない。	<p>【原因】</p> <p>DBS サーバの確認に時間がかかっているためです。</p> <p>【対応方法】</p> <p>動作しない状態のときには、タスクマネージャから強制終了して、再起動するとすぐに接続できることがあります。</p>
3	ログイン時のエラー	全バージョン	Windows のアップデート後などに、ログイン画面で、「dbSheetClient 用システム管理ファイルが正常に取得できませんでした。・・・'System.EnterpriseServices.Wrapper.dll" またはその依存関係の 1 つが読み込めませんでした。・・・」と表示されて、ログインができなくなる。	<p>【原因】</p> <p>Windows アップデートの際に、「.Net Framework2.0」の一部のファイルが壊れた可能性があります。</p> <p>【対応方法】</p> <p>「.Net Framework2.0」を再インストールすると正常にログインできるようになります。</p>
4	ログイン時のエラー	全バージョン	Excel2000 がインストールされている環境で、Excel2003 用の実行版をインストールすると起動時に、エラーが発生する。	<p>【原因】</p> <p>dbSheetClient が使用する Excel 制御のコンポーネントが Excel2000 用と Excel2003 用とでは違うためです。Excel2000 用の環境には、Excel2000 用の実行版をインストールする必要があります。</p> <p>【対応方法】</p> <p>Excel2003 用の実行版をアンインストール後、Excel2000 用の実行版をインストールすると正常に起動するようになります。</p>

【次ページに続く】

【前ページより続く】

No	機能分類	該当バージョン	現象	原因&対応方法
5	オートフィルタの設定された領域へのデータ展開	全バージョン	Excel シート上でオートフィルタの設定されている領域でデータが絞り込まれている状態のときに、201.SQL (照会)、203.データ照会画面、204.検索付データ照会画面、205.検索付データ照会画面Ⅱ、405.入出力一括 (入力) でデータを展開すると、正常にデータが展開されない現象が発生します。	画面上の「復旧」ボタンを押してください。 スクリーン表示が更新状態に戻り、正常に表示されます。

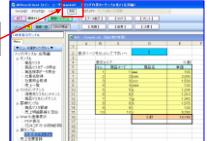


(2) 実行版のトラブルシューティング

No	機能分類	該当バージョン	現象	原因&対応方法
1	Excel のメニュー表示	全バージョン	<p>dbSheetClient の実行画面を表示中、何らかのトラブルが発生して異常終了したのち、Excel を単独で起動すると、メニューの表示がされなくなったり。</p> <p>【原因】</p> <p>Excel のメニュー やツールバーをそのままにしておくと、メニュー やツールバーから Excel を直接操作して、数式の変更や許可されていない操作でデータをコピーするなどといったことが可能になります。そのため、dbSheetClient の実行中は、Excel のメニュー やツールバーは全て隠した状態にしており、実行画面を閉じるときに元に戻しています。</p> <p>何らかの原因で異常終了すると、Excel のメニューを戻す処理がおこなわれないまま Excel のプロセスがメモリ上に残ってしまうため、次に起動したときに Excel のメニューが表示されないというケースが発生します。</p> <p>【元の設定状態に戻す方法】</p> <p>Excel のメニュー やツールバーの状態は、</p> <p>Excel2000 : Excel.xlb、 Excel2003 : Excel11.xlb、 Excel2007 : Excel12.xlb のファイルに格納されます。</p> <p>通常、Windows2000 と WindowsXP では、 C:\Documents and Settings\ユーザー名\Application Data\Microsoft\Excel の下に作られます。</p> <p>このファイルをあらかじめ保存しておき、メニューが表示されなくなったときに上書きすると、元の設定状態に戻すことができます。</p>	<p>【対応方法】</p> <p>①まず、タスクマネージャを起動して、「プロセス」のタブから Excel プロセスを選んで直接削除してください。その後、Excel を起動して、メニューが復旧すればOKです。</p> <p>②Excel のプロセスの削除は、dbSheetClient のログイン画面の「ヘルプ」ボタンをクリックして、「(2) 障害対策」のタブを表示すると、「①Excel のメニュー表示復旧」のボタンがあります。このボタンをクリックすることで、メモリ上に残っている Excel のプロセスを削除することもできます。</p> <p>③Excel のプロセスを削除しても Excel のメニューが復旧しないときには、さらに、「Excel のメニューを復旧する」ボタンをクリックすると、メニューの復旧した Excel が表示されます。そのまま終了して、Excel 起動すると復旧します。ただし、このときには、メニュー やツールバーの状態は、Excel をインストールした直後の初期状態に戻りますので、元の設定状態に戻したいときには、左記【元の設定状態に戻す方法】で、事前に「Excel ツールバーファイル」のコピーを保存しておいてください。</p>

【次ページに続く】

【前ページより続く】

No	機能分類	該当バージョン	現象	原因&対応方法
2	オートフィルタの設定された領域へのデータ展開	全バージョン	Excel シート上でオートフィルタの設定されている領域でデータが絞り込まれている状態のときに、201.SQL(照会)、203.データ照会画面、204.検索付データ照会画面、205.検索付データ照会画面Ⅱ、405.入出力一括(入力)でデータを展開すると、正常にデータが展開されない現象が発生します。	<p>【原因】 オートフィルタがかかったままの状態でそのままデータを展開すると、非表示の行からはデータを展開することができないため、データの開始位置がずれて展開されるためです。</p> <p>【対応方法】 データを展開する前に、項目名+データの範囲をタスクタイプ430.EXCEL コマンドの「データオートフィルタ解除」で一旦、オートフィルタの範囲を解除し、SQL でデータを展開した後に、再度、「オートフィルタ設定」で項目名+データの範囲に対して、オートフィルタを掛けおすことで対応することができます</p>
3	SQL(更新)、SQL(指定更新)	全バージョン	データベース(SQLServer)の型が、“smalldatetime”の場合は、1990/1/1～2079/6/1 分までの範囲時間のみを扱う場合は、Excel で扱う日付は、“1900年”からとなるため、変換工 ラーになります。	時間のみを扱う場合、データベースのデータの型は、“datetime”を使用してください。
4	全て	全バージョン	タスク実行途中でエラーが発生し、スクリーンが無反応になり、データ入力できなくなりました。	<p>画面上の「復旧」ボタンを押してください。</p> <p>スクリーン表示が更新状態に戻り、正常に表示されます。</p> 

§ 付 錄

A. 改版履歴

A. 改版履歷

株式会社ニューコム (c) Copyright 2009-2010 NEWCOM CORPORATION .

本 社 〒330-0061 埼玉県さいたま市浦和区常盤 7-3-16 ジブラルタ生命浦和ビル
TEL : 048-815-8460 FAX : 048-825-5518



株式会社ニューコム