

添付資料 C-4

平成17年度日本医師会治験促進センター治験推進研究事業

治験の IT 化の現状と課題（主任研究者：木内貴弘）

委託調査報告書

データセンター側（製薬企業など）の 臨床試験管理システムへの CDISC インターフェイス実装費用の概算試算

平成18年3月

住商情報システム株式会社

目次

1. 目的	1
2. 臨床データ管理システムの概要	1
3. CDISC インターフェース	3
3.1 要求仕様の整理	3
3.1.1 出力対象とする CDISC データモデル	3
3.1.2 入力対象とする臨床データ管理システムの検討	3
3.1.3 その他運用上の要求仕様の検討	3
3.1.4 スコープ（開発対象範囲）の検討	4
3.1.5 想定される問題点・課題の整理	4
3.2 大手製薬企業を対象とした仕組みの検討	5
3.2.1 Clintrial4 とのインターフェース検討	5
3.2.2 Oracle Clinical とのインターフェース検討	6
3.2.3 大手製薬企業向けインターフェース実装イメージ	6
3.3 中小製薬企業・医療機器・臨床研究等を対象とした仕組みの検討	7
3.3.1 Office2003 を使用したインターフェース検討	7
3.3.2 オープンソースの利用	8
3.3.3 中小企業向けインターフェース実装イメージ	8
3.4 システム開発手順	9
3.4.1 基本的なマイルストーン	10
3.4.2 推敲と製作反復プロセスの適用（個別開発プロセス）	10
3.5 作業タスク、開発スケジュール及び概算費用	11
別添 1	CLINTRIAL 説明資料「CLINTRIAL 4.3 の概要」
別添 2	Oracle Clinical 説明資料「Overview of Oracle Clinical」
別添 3	SAS 説明資料「An Introduction to CDISC」
別添 4	MySQL 説明資料「MySQL 技術概要」

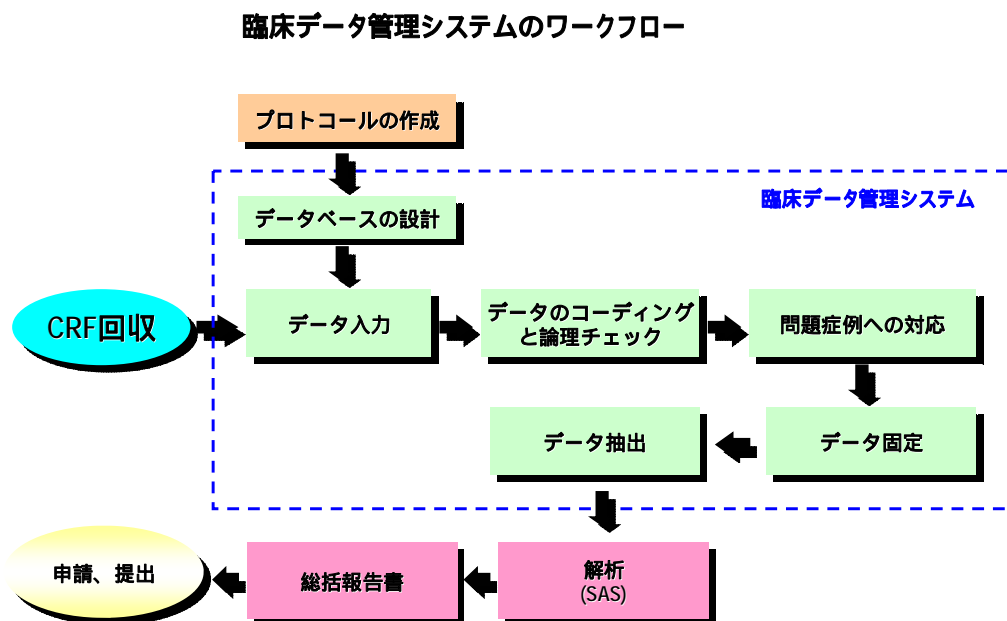
1. 目的

日本国内の製薬企業においては、治験データを取り扱うために臨床データ管理システムを、殆どの企業で導入している。しかしながら、CDISC での申請については、臨床データ管理システムへのインターフェースが推進されていないこともあり、十分整備されていない。今回、日本国内における臨床データ管理システムへの CDISC インターフェースを検討し、今後の臨床データ管理システムに対して標準データの取扱いができるシステムについて試算を行う。

2. 臨床データ管理システムの概要

臨床データ管理システムは、治験データの入力 DB 設計からデータ入力、コーディング、論理チェック、データ固定、データ抽出までを一連で行えるシステムである。

下記に臨床データ管理システムの概要を示す。



主な機能としては、次のような機能が含まれる。

(1) データ定義設計

- ・ プロトコル毎のデータ定義を設定する。
- ・ データ定義作成または修正時に、DB テーブルの自動作成または変更ができ、データ入力時に使用するコード辞書参照用のフォーマットの自動割り当てを行う。

(2)コード辞書管理

- ・コード辞書は、共通またはプロトコール2階層で管理ができる。
- ・MeDRAなどの辞書についてのバージョン管理が可能。

(3)入力画面作成

- ・デフォルトの入力画面の自動作成を行う。
- ・画面レイアウトはユーザ側で自由に編集することが可能。

(4)データ入力

- ・入力設計支援で定義したプロトコル単位のデータ定義別にデータ入力を行う。
- ・データ入力の基本は、ダブルエントリとする。但し、シングルエントリも可能。
- ・一度DB登録を行った後のデータ修正及びDB更新は、履歴、理由の入力を行う。

(5)チェックリスト作成

- ・入力データのチェックリスト出力を行う。

(6)データ固定

- ・DBに登録済み症例データに対して、症例毎データの単位でデータ固定（仮固定）ができる。

(7)テーマ・プロトコール管理

- ・新規テーマの情報を管理できる。
- ・登録済みテーマに新たに発生したプロトコールの情報を管理できる。

(8)役割管理

- ・新規役割に対する処理別のアクセス権を管理し、システム権限を設定する。
- ・登録済み役割に対する処理別のアクセス権の修正を行い、システム権限を判別し、ユーザのシステム権限の付与、削除を行う。また、テーマ・プロトコール別、ユーザ別処理アクセス権の構築を行う。

(9)データの inport/export

- ・臨床検査値データ、CROからのデータなどの一括取り込み機能。
- ・SAS、その他アプリケーションへのEXPORTファイルの作成機能。

以上が、主な臨床データ管理システムの機能概要である。

また、現在、国内外で利用されている主な臨床データ管理システムとしては、

Clintrial（Phase Forward社製）・・・別添1参照

Oracle Clinical（Oracle社製）・・・別添2参照

があり、他国産メーカーのは多種市販されている。

解析パッケージSASについては、別添3を参照

3. CDISC インターフェース

ここでは CDISC インターフェースをコンピュータシステムへ実装想定するにあたり、一先ずどの範囲までを対象とするのか、現状において何が課題なのか、を明確にする。また、CDISC インターフェース構築の開発手順を示す事で、システムの信頼性保障と、費用試算の前提となる作業工程を明らかにする。

3.1 要求仕様の整理

インターフェースを構築するということは、すなわち、コンピュータシステム上で入力と出力の対応付け（マッピング）を行うことになる。については、物理的に何種類の紐付けを想定するかによって、システムの実装工数は大きく異なる。以下に、出力対象と入力対象を規定する。

3.1.1 出力対象とする CDISC データモデル

CDISC データモデルは試験の開始から規制当局への申請、データの保管に至るまでの臨床試験データフロー全体をサポートする。具体的には、Operational Data Model(ODM)、Study Data Tabulation Model(SDTM)、Case Report Tabulation Data Definition Specification(CRTDDS)、Laboratory Data Model(Lab)、Analysis Data Model(ADaM)、Standard Exchange of Non-clinical Data(SEND)のモデルから構成される。今回の CDISC インターフェース構築においては、日本 CDISC グループの翻訳した ODM と SDTM を出力対象として以降の要求仕様検討を行う。

3.1.2 入力対象とする臨床データ管理システムの検討

臨床データ管理システムは、試験を実施する企業の規模により大別すると 2 つに分類される。大手製薬会社では、臨床データ更新の頻度、多重度の高い運用であっても、ストレスを感じ無い処理速度や、高度な信頼性等を確保するという観点から、オラクル社の Oracle Database をデータベースとしたシステムが使用されている。具体的な導入パッケージ製品としては、フェーズフォワード社の「Clintrial4」とオラクル社の「Oracle Clinical」の 2 製品に、現状は、ほぼ集約されている。

一方、中小製薬会社や医療機器メーカーにおいては、複数試験を並行して進めるような事態が少ない為、大手製薬会社ほどの、高度な信頼性を不要となる。現状としては、マイクロソフト社のオフィス製品を使用しているケースが非常に多く、具体的には、Excel や Access データベースを上記臨床データを保管する手法に、ほぼ集約されている。

よって、上記の 2 つのパターンに対応する CDISC インターフェース構築を目指す。

3.1.3 その他運用上の要求仕様の検討

前述のとおり、インターフェースシステムの主要な機能は、マッピング機能であるが、実際の運用を想定した場合、それ以外のいくつか機能の必要性も考えられる。まず一つは、XML 自体がテキスト形式である以上、CDISC 標準で XML 化された臨床データを、データベース上に保存する機能、もう一つは、印刷用の証拠書類（Evidence）として PDF 化する

機能が考えられる。

3.1.4 スコープ（開発対象範囲）の検討

前項までの要求仕様整理の内容を踏まえ、システムのスコープを下図に示す。

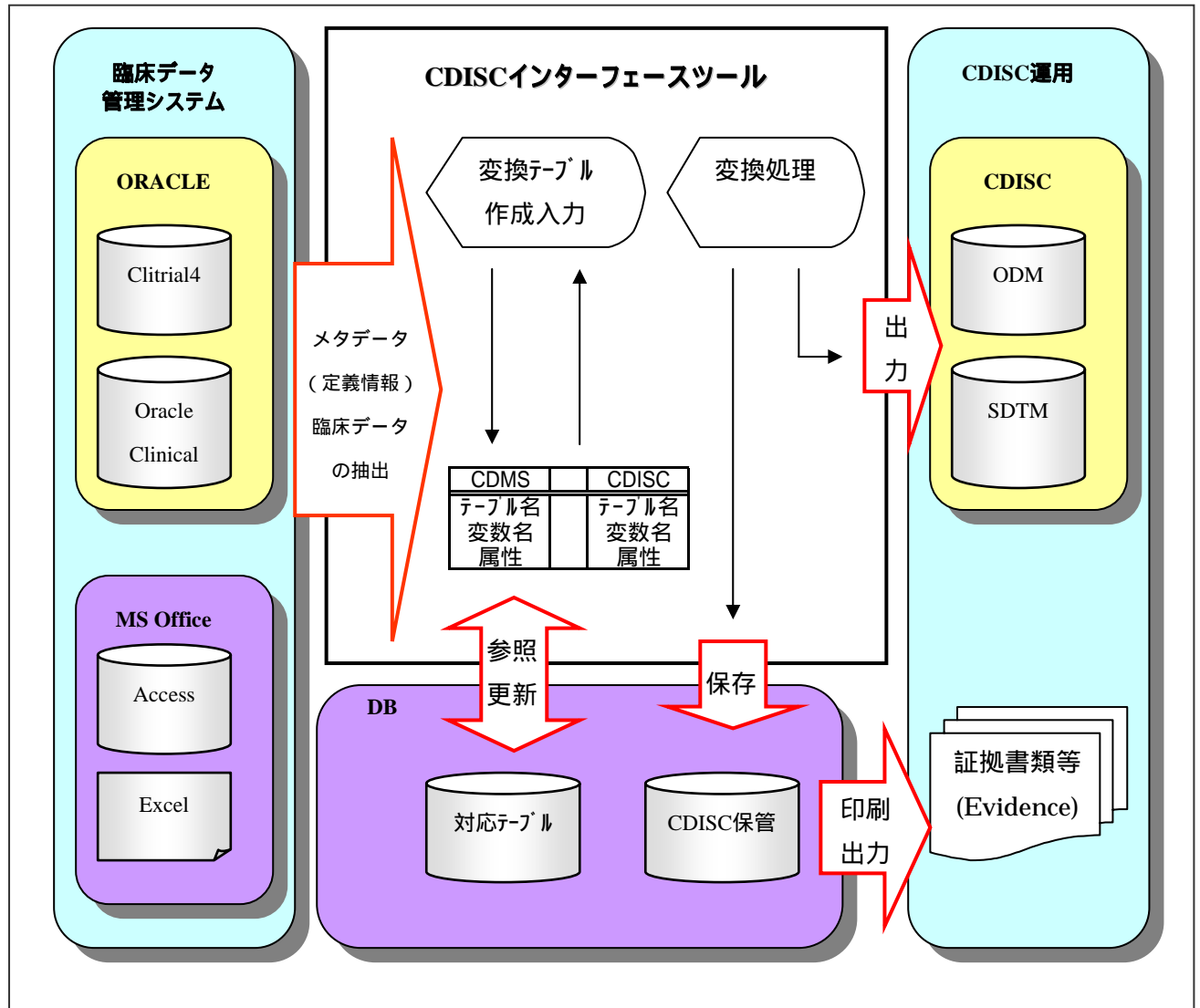


図1：システムスコープ

3.1.5 想定される問題点・課題の整理

上記図1のシステムスコープの実現を目指す上で、現状で想定している解決すべき問題点や検討を要する課題は、以下の通りとなる。

- ・ OracleClinical のデータベース構造資料が非開示である為、現状では定義情報や臨床データを OracleClinical 外部から参照する事が出来ない。製造元（オラクル社）や日本での販売元（シーティーシー・ラボラトリーシステムズ株式会社）との交渉を要する。
- ・ エラートラップ（例外処理：CDISC インターフェースツール上のエラーチェック）で対応するものと、ツール利用前提条件（最低限意識しなければ CDISC インターフェー

ツールが機能しない標準化等の規則)で除外するものとのバランスを、どこに合わせるか。ツールの利便性を左右し、またシステムの開発工数に大きく影響を与える為十分な検討を要する。

3.2 大手製薬企業を対象とした仕組みの検討

3.2.1 Clintrial4 とのインターフェース検討

Clintrial4 のデータベース構造はシステム利用ユーザーに公開されている。システムの運用形態によりインスタンス(データベースを起動・停止する単位で、管理者の立場から見ると、1つのデータベース管理単位)を個別に意識する必要はあるが、プロトコル以下の概念は全て Clintrial4 上の定義情報に格納されている為、データの抽出をインターフェースツール上の機能として実装可能と判断出来る。具体的な定義情報としては以下の通りとなる。

- ・ プロトコル情報格納テーブル：CTS.CTS_PROTOCOLS
- ・ パネル情報格納テーブル：CTSDD.PANEL
- ・ コード情報格納テーブル：CTSCODES.CODE_INDEX
- ・ 項目情報格納テーブル：CTSDD.ITEM

なお、物理的なデータ構造については、以下の E - R 図(Entity Relation Diagram)に示す。

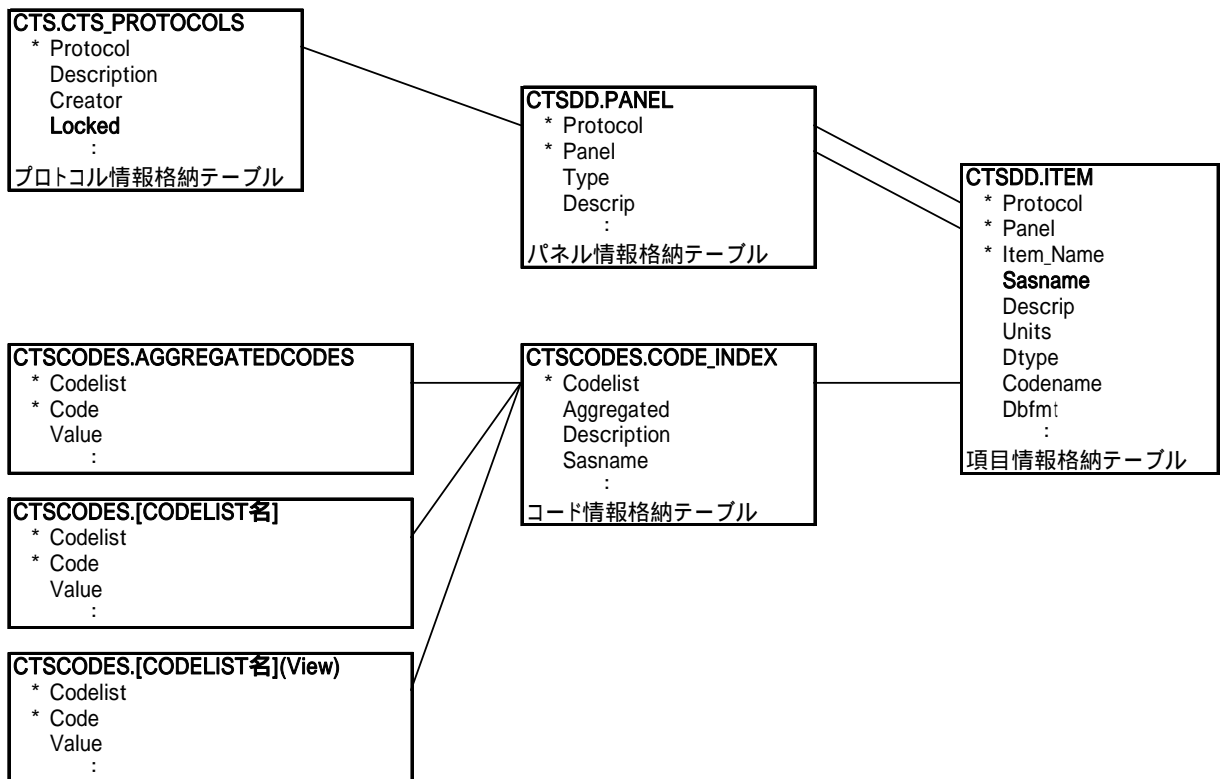


図2 . Clintrial4 メタ情報 E - R

Clintrial4 はオラクルデータベースから見た場合、上記図 2 の定義情報保有する一つの処理単位となる。ついては、オラクルデータベース上に参照権限を持つユーザーを別途付与し、そのユーザーを経由して定義情報をたどり、対象とするプロトコールの症例データを抽出出来ると考える。

但し、Clintrial4 において暗号化設定（外部から参照を不可とするオプション設定）が成されている場合、抽出は不可能となる。

3.2.2 Oracle Clinical とのインターフェース検討

Oracle Clinical のデータベース構造は、前述の通り、原則公開されていない。については Clintrial4 のように、外部からダイレクトにデータ抽出する事は出来ない。とり得る手段としては、Oracle Clinical に標準的に実装されている出力機能「Data Extract View Builder」を使用し、Oracle Clinical 側から SAS データセット等のインターフェースファイルや外部から参照可能な View を生成する事となる。また、「Data Extract View Builder」では、以下のような出力が可能であることが公開されている。

- ・ Default view
- ・ Custom view(One-DCM View、 Cross-DCM View)
DCM : Data Collection Module (Oracle Clinical 上の用語で調査項目の集合体を示す)
- ・ In-Study-Unions
- ・ Cross-Study-Unions
- ・ SAS view
- ・ SAS Data set(Table)
- ・ SAS PROC Report(ASCII text file)

上記の内容について詳細な情報が開示されていない事から、今回のシステムスコープにおいて、相応しい手段を判断するには至らない。ここでは、調査や評価の対象とすべき項目を述べるに止める。

3.2.3 大手製薬企業向けインターフェース実装イメージ

全ての大手製薬会社の統計解析部門や臨床開発部門においては、SAS システムが導入されている。よって、大手製薬企業向けインターフェース実装にあたっては、SAS データセットのファイル形式を介したデータ交換が可能と考えられる。加えて、SAS には標準的な XML エンジンだけでなく PROC CDISC(ODM は製品化済み、SDTM にも近い将来対応予定) や ODM Viewer などの CDISC 支援機能も実装されている。これらの機能はダブルバイト文字での検証が必要ではあるが、大手製薬会社向けのインターフェース検討において、実行環境の選定にあたり、十分なアドバンテージと言える。SAS データセットを介した実装案を下図 3 に提示する。

なお、実際のシステム実装にあたっては、各ソフトウェアのバージョンやソフトウェア内で必要とされるプロダクトやオプションについても明らかにする必要がある。具体的には以下のソフトウェア製品が対象となる。

- Oracle Database
- Clitrial
- Oracle Clinical
- SAS

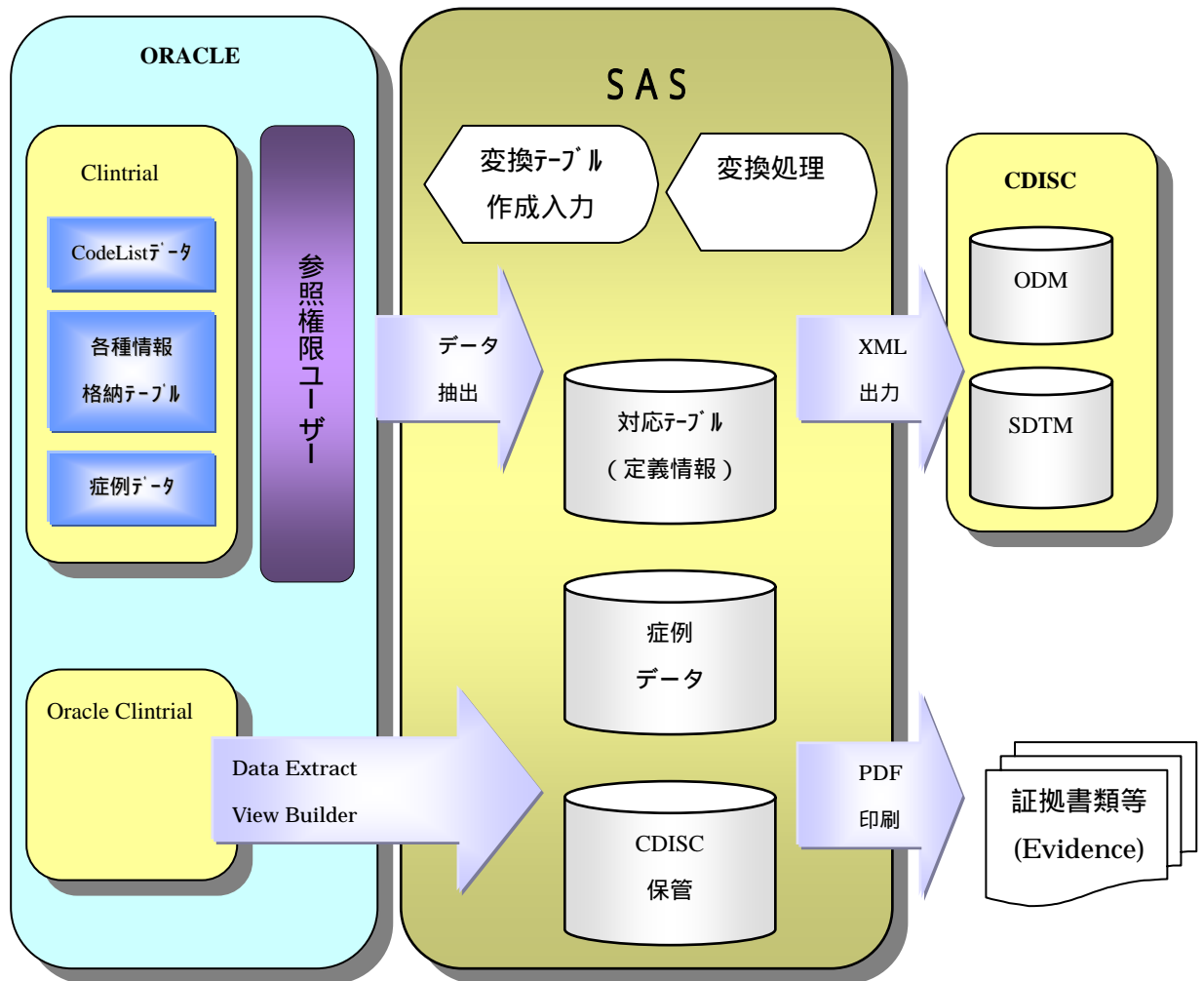


図 3 : 大手製薬企業向けインターフェース実装イメージ

3.3 中小製薬企業・医療機器・臨床研究等を対象とした仕組みの検討

3.3.1 Office2003 を使用したインターフェース検討

マイクロソフト社の Excel や Access データベースも Office2003 のバージョンからは、XML 対応機能が強化されている。Access2003 では W3C (World Wide Web Consortium : WWW で使用されるさまざまな技術標準を定める総本山的な組織で XML 仕様もここで制定された)の標準スキーマ言語 (XML スキーマ) に対応し、Access から XML データを直接エクスポートする際に、XSLT 変換 (任意の XML 文書を読み込んで、それを加工して出力する) が可能

である。また Excel2003 では任意のスキーマ（XML 文書の取り得る構造を記述したもの）が取り込め、また Excel 上のデータを XML 要素に関連付けすることが可能である。

なお、変換テーブル作成等のツールについては、具体的なシステム要求次第ではあるが、システムの親和性の観点から Office2003 にバンドルされる Info Path2003 あるいは Visual Studio を使用したい。

3.3.2 オープンソースの利用

中小製薬企業等を対象とした場合、CDISC インターフェースツールの利用に必要な費用を最小限に止める事も、大変重要となる。については、対応テーブルや CDISC 保管等の DB（図 1：システムスコープを参照）に関しては、SAS や Oracle のように高価な商用製品ではなくオープンソースソフトウェア（ソースコードを公開し、そのソフトウェアの利用、改変、配布を誰もが自由に行う事が出来る）データベースの一つである「MySQL」を利用する。なお、「MySQL」では、無償で使用出来る GPL ライセンス（但し、アプリケーションのソースコードを公開し、誰もが利用可能な状態にしなくてはならない）の採用を予定している。

3.3.3 中小企業向けインターフェース実装イメージ

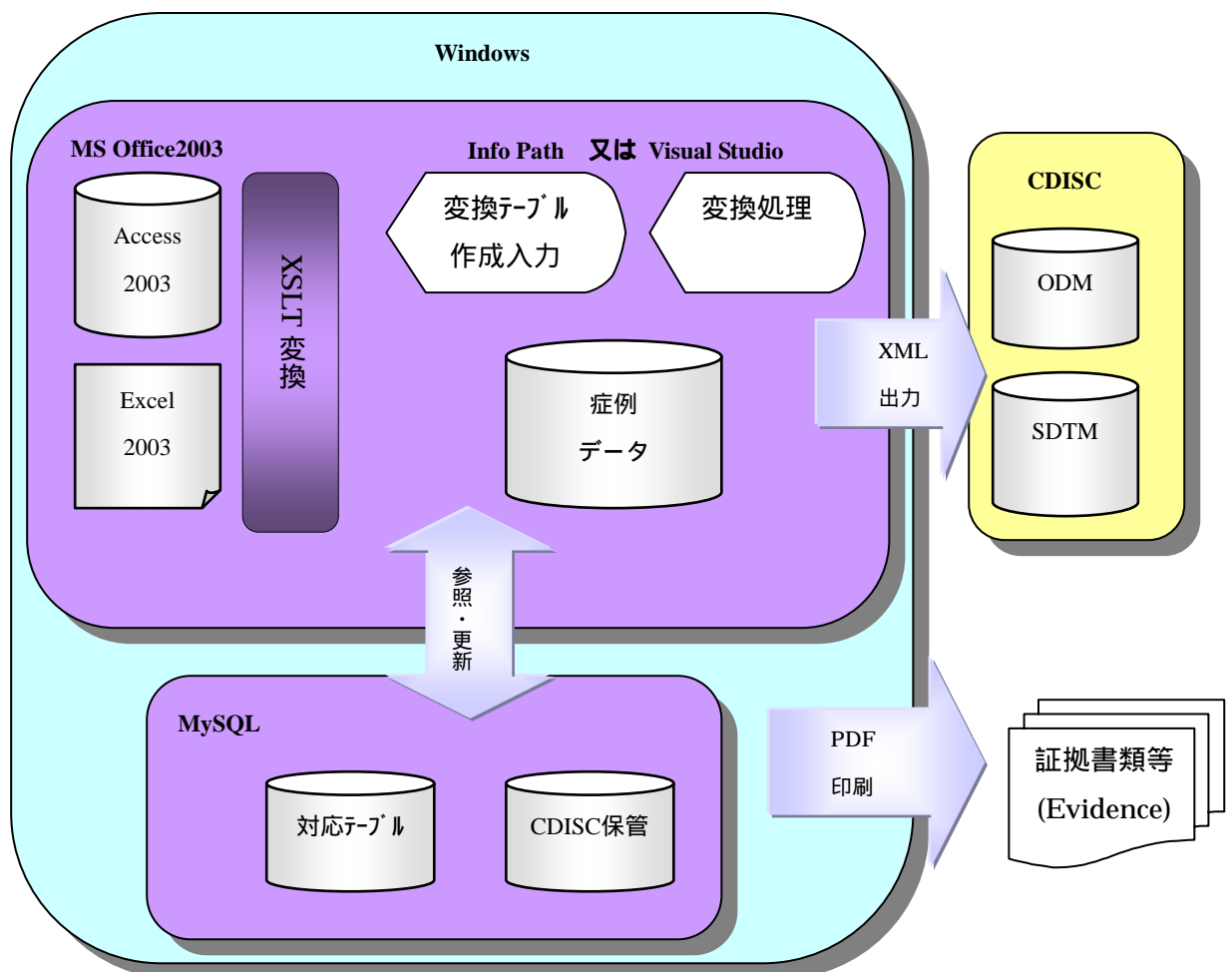


図 4：中小企業向けインターフェース実装イメージ

なお、「MySQL」の製品詳細については、製造元の MySQL AB 社作成の別添 4 を参照されたい。

3.4 システム開発手順

システム開発にあたっては、住商情報システムが開発標準としている SMART プロセスに則り、実施する。(SMARTとは「information Systems design and Management Renaissance project」の呼称)

以下に、SMART プロセスが基本に置く考え方を示す。

(1) ルール重視

計画的で組織的に開発を遂行するための、ルールを重視した進め方を前提とする。つまり、手順や成果物、作業担当をきちんと定めて実施するプロジェクトを想定する。このため、標準的な開発プロセスや成果物、また実施に当たっての技術的ガイドを整備している。

(2) リスク・ドリブン

特にオープン系システムでは技術領域が拡大し、技術面の問題が起りやすく、特にシステム全体に影響を与える基盤(アーキテクチャ)系のリスクが増加している。これにエンドユーザ参加度増に伴う不安定さや短納期・低コスト要求が加わり、これらリスクへの対処が重要課題となっている。このようなリスクがありきの考え方に立ち、一般的に「反復」と呼ばれる進め方を取入れるなど、基本的な方策を織り込んでいる。つまり、手順や成果物、作業担当をきちんと定めて実施するプロジェクトを想定する。このため、標準的な開発プロセスや成果物、また実施に当たっての技術的ガイドを整備している。

(3) 品質重視

一つ一つの工程をきちんと実施し、レビューやテストで都度の品質確保を行う。また、開発ライフサイクル視点からのマイルストーンのクリアとベースライニングによる変更管理など、一般的に「V字モデル」と呼ばれる品質確保の方式やプロジェクト管理方式を採用している。

(4) アーキテクチャ重視

今後のシステム化では、アプリケーションと基盤となるシステム(アーキテクチャ)が渾然一体化し、表裏一体で考える必要がある。また、システム基盤の是非がシステム全体に大きな影響を与える状況になっている。よって、アーキテクチャ設計にも注視したプロセス整備や、早期にアーキテクチャを確定的にした上でAP設計を進め大きな手戻りの危険性を低減する進め方など、アーキテクチャ重視の方策を織り込んでいる。

(5) プロセス設計(個々プロジェクトへの最適化)

標準では、分析から移行・運用に至る開発作業連携を論理的視点で整理し「基本開発プロセス」として定めている。しかし、実際の開発プロジェクトでは多様な条件や状況が存在し、十分な検討をせずに基本開発プロセスをそのまま適用するとすることには危険性が伴う。特に旧来

に比して多様性の高いオープン系システムではそう言える。よって、基本開発プロセスを参照し、各々のプロジェクトに合わせたプロセス設計(最適化)を行うように活用することを前提としている。

(6) エンジニアリング重視

社会的に評価を得ている考え方や手法また技術を活用することが、品質や効率、また信頼性の視点からも重要となる。当標準でもそれらを採用している。しかし単純に組み入れているのではなく、当社にマッチした内容にカスタマイズしている。

3.4.1 基本的なマイルストーン

前述の基本に置く考え方を盛り込んだ、開発プロセスの基本的なマイルストーンを下図に示す。

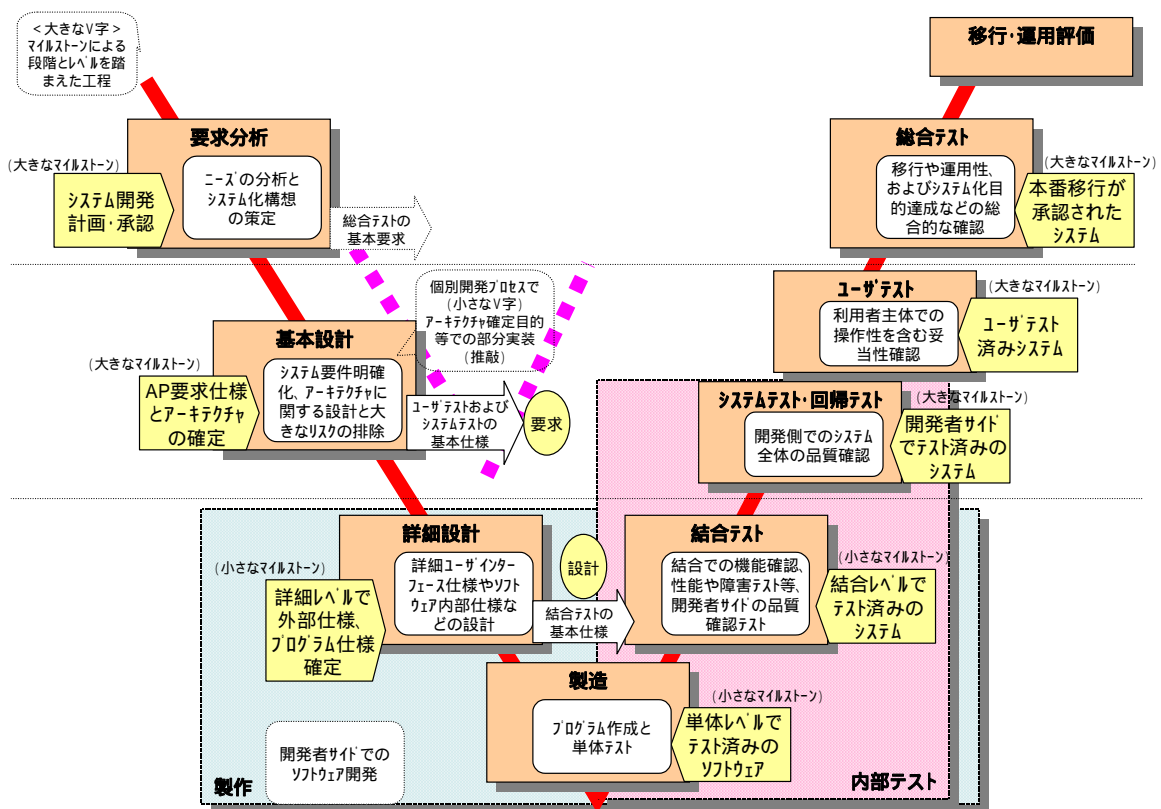


図4：基本的なマイルストーン

3.4.2 個別開発プロセスの適用

個別開発プロセスとは、基本開発プロセスを参照し、各々のプロジェクトに合わせて設計したプロセスを示す。個別開発プロセスには、システム開発に関する基本開発プロセス中のタスクに加え、計画作成やレビューなどの作業ネットワークとして組み込まれるべきプロジェクト管理のタスクも包含される。SMARTでは、個別開発プロセスの代表的なフェーズ構成の例として以下の4つのパターンを提供する。

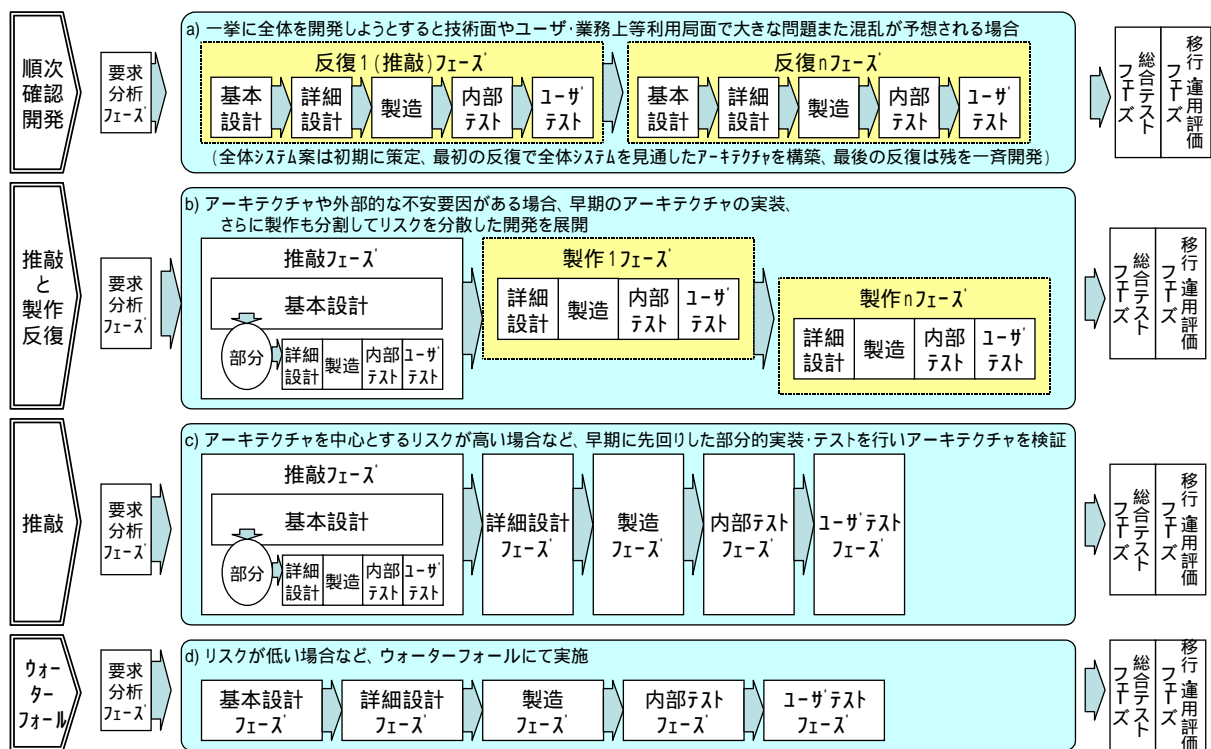


図5：個別開発プロセス代表パターン

CDISC インターフェース実装においては、要求仕様整理で述べたように問題点や課題が見込まれる為、推敲と製作反復プロセスを適用する。

3.5 作業タスク、開発スケジュール及び概算費用

インターフェース開発作業の概略と概算費用、想定工数を下図に示す。なお、対象機能群としては大手製薬企業向け Crintrial インターフェース、Oracle Clinical インターフェース、中小企業向け MS-Office インターフェースとした。

また、それぞれの作業タスクにおける概算を取り纏めた。なお、ハードウェア、ミドルウェア及び保守費用については含まない。

作業タスク、開発スケジュール案、及び概算費用(Clintrial)

タスク	作業概要	概算費用 (単位：千円)
要求分析フェーズ (Clintrialを対象)		
0. 要求仕様の再確認		
0.1.1 出力対象CDISCモデル検討	日本語化も含めてCDISCモデルやバージョンを検討	20,000 (約10~12人月)
0.2.1 入力対象CDMS検討	CDMSやofficeのバージョン等の検討	
0.3.1 その他運用上の仕様検討	業務上の作業の流れを意識した運用上の必要機能検討	
0.4.1 想定される課題の整理	エラーラップ検討他	
0.5.1 要求仕様書の作成	上記をシステムスコープとした要求仕様書を作成	
0.5.2 要求仕様書のレビュー		
0.5.3 要求仕様書の改定		
0.X.9 要求仕様書の固定		
推敲フェーズ		
1. 推敲フェーズ計画の作成		
1.1.1 推敲フェーズ対象の確定	全体要求仕様の内、先行着手すべき機能を確定。	10,000 (約5~7人月)
1.2.1 役割と責任の取決め	同左	
1.3.1 定例会議の取決め	同左	
1.4.1 プロジェクト管理プロセスの取決め	同左	
1.5.1 推敲フェーズ開発計画の作成	推敲フェーズ作業のWBSを作成する。	
1.5.2 推敲フェーズ開発計画のレビュー	上記WBSのレビュー	
1.5.3 推敲フェーズ開発計画の改定	上記WBSの改定	
1.5.9 推敲フェーズ開発計画の固定	上記WBSの固定	
2. 推敲フェーズの実施		
2.1.1 基本設計	要求仕様書を元にシステムの外部(論理)仕様書を作成	30,000 (約15~20人月)
2.2.1 詳細設計	外部仕様書を元に内部(物理)仕様書を作成	
2.3.1 プログラム開発	同左	
2.4.1 内部テスト	内部(物理)仕様を確認するテスト	
2.5.1 ユーザーテスト	外部(論理)仕様を確認するテスト	
2.6.1 推敲フェーズの評価	同左	
3. 製作フェーズ計画の作成		
3.1.1 製作フェーズ1対象の確定	全体要求仕様の内、開発対象機能を確定。	10,000 (約5~7人月)
3.2.1 役割と責任の取決め	同左	
3.3.1 定例会議の取決め	同左	
3.4.1 プロジェクト管理プロセスの取決め	同左	
3.5.1 製作フェーズ1開発計画の作成	推敲フェーズ作業のWBSを作成する。	
3.5.2 製作フェーズ1開発計画のレビュー	上記WBSのレビュー	
3.5.3 製作フェーズ1開発計画の改定	上記WBSの改定	
3.5.9 製作フェーズ1開発計画の固定	上記WBSの固定	
4. 製作フェーズの実施		
4.1.1 基本設計	要求仕様書を元にシステムの外部(論理)仕様書を作成	40,000 (約20~30人月)
4.2.1 詳細設計	外部仕様書を元に内部(物理)仕様書を作成	
4.3.1 プログラム開発	同左	
4.4.1 内部テスト	内部(物理)仕様を確認するテスト	
4.5.1 ユーザーテスト	外部(論理)仕様を確認するテスト	
4.6.1 製作フェーズ1の評価	同左	
総合テストフェーズ		
5. 総合テスト		
5.1.1 総合テスト計画の検討	テスト計画について、総合的な品質の点から検討する。	20,000 (約10~15人月)
5.2.1 総合テスト実施計画書の作成	検討を基に、テスト計画書を作成する。	
5.3.1 総合テスト実施環境の構築	7.2.1のテスト計画書に基づきシステムテスト実施環境を構築する。	
5.6.1 総合テスト実施データの作成		
5.7.1 総合テストの実施		
5.8.1 総合テスト実施報告書の作成		
5.8.2 総合テスト実施報告書 レビュー承認		
6. システムバリデーション		
6.1.1 バリデーション計画の検討	実施計画について、総合的な品質の点から検討する。	20,000 (約10~15人月)
6.2.1 バリデーション実施計画書の作成	検討を基に、テスト計画書を作成する。	
6.3.1 バリデーション実施環境の構築	バリデーション計画書に基づき実施環境を構築する。	
6.6.1 バリデーションデータの作成		
6.7.1 バリデーションの実施		
6.8.1 バリデーション報告書の作成		
6.8.2 バリデーション報告書レビュー承認		
移行・運用解任フェーズ		
7. ドキュメント作成(運用、操作)		
7.1.1 操作手順書の作成		20,000 (約10~15人月)
7.1.2 操作手順書レビュー		
7.2.1 運用手順書の作成		
7.2.2 運用手順書レビュー		
導入作業		
8. 導入作業		
8.1.1 システムインストール	システムインストール	10,000 (約5~7人月)
8.2.1 IQ/OQ支援	IQ/OQ支援作業	
8.3.1 導入支援サポート	導入時説明、教育、初期設定	
8.4.1 ユーザ教育	操作説明、運用手順説明	
	計	180,000

作業タスク、開発スケジュール案、及び概算費用(Oracle Clinical)

タスク	作業概要	概算費用 (単位：千円)
要求分析フェーズ (Oracle Clinicalを対象)		
0. 要求仕様の再確認		
0.1.1 出力対象CDISCモデル検討	日本語化も含めてCDISCモデルやバージョンを検討	30,000 (約15~20人月)
0.2.1 入力対象CDMS検討	CDMSやofficeのバージョン等の検討	
0.3.1 その他運用上の仕様検討	業務上の作業の流れを意識した運用上の必要機能検討	
0.4.1 想定される課題の整理	Oracle Clinicalの非開示情報やエラーラップ検討	
0.5.1 要求仕様書の作成	上記をシステムスコープとした要求仕様書を作成	
0.5.2 要求仕様書のレビュー		
0.5.3 要求仕様書の改定		
0.X.9 要求仕様書の固定		
推敲フェーズ		
1. 推敲フェーズ計画の作成		
1.1.1 推敲フェーズ対象の確定	全体要求仕様の内、先行着手すべき機能を確定。	10,000 (約5~7人月)
1.2.1 役割と責任の取決め	同左	
1.3.1 定例会議の取決め	同左	
1.4.1 プロジェクト管理プロセスの取決め	同左	
1.5.1 推敲フェーズ開発計画の作成	推敲フェーズ作業のWBSを作成する。	
1.5.2 推敲フェーズ開発計画のレビュー	上記WBSのレビュー	
1.5.3 推敲フェーズ開発計画の改定	上記WBSの改定	
1.5.9 推敲フェーズ開発計画の固定	上記WBSの固定	
2. 推敲フェーズの実施		
2.1.1 基本設計	要求仕様書を元にシステムの外部(論理)仕様書を作成	40,000 (約20~30人月)
2.2.1 詳細設計	外部仕様書を元に内部(物理)仕様書を作成	
2.3.1 プログラム開発	同左	
2.4.1 内部テスト	内部(物理)仕様を確認するテスト	
2.5.1 ユーザーテスト	外部(論理)仕様を確認するテスト	
2.6.1 推敲フェーズの評価	同左	
3. 製作フェーズ計画の作成		
3.1.1 製作フェーズ1対象の確定	全体要求仕様の内、開発対象機能を確定。	10,000 (約5~7人月)
3.2.1 役割と責任の取決め	同左	
3.3.1 定例会議の取決め	同左	
3.4.1 プロジェクト管理プロセスの取決め	同左	
3.5.1 製作フェーズ1開発計画の作成	推敲フェーズ作業のWBSを作成する。	
3.5.2 製作フェーズ1開発計画のレビュー	上記WBSのレビュー	
3.5.3 製作フェーズ1開発計画の改定	上記WBSの改定	
3.5.9 製作フェーズ1開発計画の固定	上記WBSの固定	
4. 製作フェーズの実施		
4.1.1 基本設計	要求仕様書を元にシステムの外部(論理)仕様書を作成	80,000 (約40~50人月)
4.2.1 詳細設計	外部仕様書を元に内部(物理)仕様書を作成	
4.3.1 プログラム開発	同左	
4.4.1 内部テスト	内部(物理)仕様を確認するテスト	
4.5.1 ユーザーテスト	外部(論理)仕様を確認するテスト	
4.6.1 製作フェーズ1の評価	同左	
総合テストフェーズ		
5. 総合テスト		
5.1.1 総合テスト計画の検討	テスト計画について、総合的な品質の点から検討する。	20,000 (約10~15人月)
5.2.1 総合テスト実施計画書の作成	検討を基に、テスト計画書を作成する。	
5.3.1 総合テスト実施環境の構築	7.2.1のテスト計画書に基づきシステムテスト実施環境を構築する。	
5.6.1 総合テスト実施データの作成		
5.7.1 総合テストの実施		
5.8.1 総合テスト実施報告書の作成		
5.8.2 総合テスト実施報告書レビュー承認		
6. システムバリデーション		
6.1.1 バリデーション計画の検討	実施計画について、総合的な品質の点から検討する。	20,000 (約10~15人月)
6.2.1 バリデーション実施計画書の作成	検討を基に、テスト計画書を作成する。	
6.3.1 バリデーション実施環境の構築	バリデーション計画書に基づき実施環境を構築する。	
6.6.1 バリデーションデータの作成		
6.7.1 バリデーションの実施		
6.8.1 バリデーション報告書の作成		
6.8.2 バリデーション報告書レビュー承認		
移行・運用評価フェーズ		
7. ドキュメント作成(運用、操作)		
7.1.1 操作手順書の作成		20,000 (約10~15人月)
7.1.2 操作手順書レビュー		
7.2.1 運用手順書の作成		
7.2.2 運用手順書レビュー		
導入作業		
8. 導入作業		
8.1.1 システムインストール	システムインストール	10,000 (約5~7人月)
8.2.1 IQ/OQ支援	IQ/OQ支援作業	
8.3.1 導入支援サポート	導入時説明、教育、初期設定	
8.4.1 ユーザ教育	操作説明、運用手順説明	
	計	240,000

作業タスク、開発スケジュール案、及び概算費用(MS-Office)

タスク	作業概要	概算費用 (単位：千円)
要求分析フェーズ (MS-Officeを対象)		
0. 要求仕様の再確認		
0.1.1 出力対象CDISCモデル検討	日本語化も含めてCDISCモデルやバージョンを検討	10,000 (約5~7人月)
0.2.1 DB検討		
0.3.1 その他運用上の仕様検討	業務上の作業の流れを意識した運用上の必要機能検討	
0.4.1 想定される課題の整理		
0.5.1 要求仕様書の作成	上記をシステムスコープとした要求仕様書を作成	
0.5.2 要求仕様書のレビュー		
0.5.3 要求仕様書の改定		
0.X.9 要求仕様書の固定		
推敲フェーズ		
1. 推敲フェーズ計画の作成		
1.1.1 推敲フェーズ対象の確定	全体要求仕様の内、先行着手すべき機能を確定。	6,000 (約3~5人月)
1.2.1 役割と責任の取決め	同左	
1.3.1 定例会議の取決め	同左	
1.4.1 プロジェクト管理プロセスの取決め	同左	
1.5.1 推敲フェーズ開発計画の作成	推敲フェーズ作業のWBSを作成する。	
1.5.2 推敲フェーズ開発計画のレビュー	上記WBSのレビュー	
1.5.3 推敲フェーズ開発計画の改定	上記WBSの改定	
1.5.9 推敲フェーズ開発計画の固定	上記WBSの固定	
2. 推敲フェーズの実施		
2.1.1 基本設計	要求仕様書を元にシステムの外部(論理)仕様書を作成	20,000 (約10~15人月)
2.2.1 詳細設計	外部仕様書を元に内部(物理)仕様書を作成	
2.3.1 プログラム開発	同左	
2.4.1 内部テスト	内部(物理)仕様を確認するテスト	
2.5.1 ユーザーテスト	外部(論理)仕様を確認するテスト	
2.6.1 推敲フェーズの評価	同左	
3. 製作フェーズ計画の作成		
3.1.1 製作フェーズ1対象の確定	全体要求仕様の内、開発対象機能を確定。	6,000 (約3~5人月)
3.2.1 役割と責任の取決め	同左	
3.3.1 定例会議の取決め	同左	
3.4.1 プロジェクト管理プロセスの取決め	同左	
3.5.1 製作フェーズ1開発計画の作成	推敲フェーズ作業のWBSを作成する。	
3.5.2 製作フェーズ1開発計画のレビュー	上記WBSのレビュー	
3.5.3 製作フェーズ1開発計画の改定	上記WBSの改定	
3.5.9 製作フェーズ1開発計画の固定	上記WBSの固定	
4. 製作フェーズの実施		
4.1.1 基本設計	要求仕様書を元にシステムの外部(論理)仕様書を作成	40,000 (約20~30人月)
4.2.1 詳細設計	外部仕様書を元に内部(物理)仕様書を作成	
4.3.1 プログラム開発	同左	
4.4.1 内部テスト	内部(物理)仕様を確認するテスト	
4.5.1 ユーザーテスト	外部(論理)仕様を確認するテスト	
4.6.1 製作フェーズ1の評価	同左	
総合テストフェーズ		
5. 総合テスト		
5.1.1 総合テスト計画の検討	テスト計画について、総合的な品質の点から検討する。	10,000 (約5~7人月)
5.2.1 総合テスト実施計画書の作成	検討を基に、テスト計画書を作成する。	
5.3.1 総合テスト実施環境の構築	7.2.1のテスト計画書に基づきシステムテスト実施環境を構築する。	
5.6.1 総合テスト実施データの作成		
5.7.1 総合テストの実施		
5.8.1 総合テスト実施報告書の作成		
5.8.2 総合テスト実施報告書レビュー承認		
6. システムバリデーション		
6.1.1 バリデーション計画の検討	実施計画について、総合的な品質の点から検討する。	10,000 (約5~7人月)
6.2.1 バリデーション実施計画書の作成	検討を基に、テスト計画書を作成する。	
6.3.1 バリデーション実施環境の構築	バリデーション計画書に基づき実施環境を構築する。	
6.6.1 バリデーションデータの作成		
6.7.1 バリデーションの実施		
6.8.1 バリデーション報告書の作成		
6.8.2 バリデーション報告書レビュー承認		
移行・運用評価フェーズ		
7. ドキュメント作成(運用、操作)		
7.1.1 操作手順書の作成		10,000 (約5~7人月)
7.1.2 操作手順書レビュー		
7.2.1 運用手順書の作成		
7.2.2 運用手順書レビュー		
導入作業		
8. 導入作業		
8.1.1 システムインストール	システムインストール	10,000 (約5~7人月)
8.2.1 IQ/OQ支援	IQ/OQ支援作業	
8.3.1 導入支援サポート	導入時説明、教育、初期設定	
8.4.1 ユーザ教育	操作説明、運用手順説明	
	計	122,000