

薄まった結果になっていることがクローズアップされた。次年度は今回開発した業務モデルを、本研究と並行して研究を行った電子カルテシステムのエンタープライズモデル開発の to be モデルへ反映させ、さらに精度の高い業務モデルとするとともに、具体的に導入が想定される地域中核病院での機能要件、データ諸元等をもとにコンピューショナルモデルのシステム基盤についての検討を行う。

4. データモデルの開発

分担研究者山本隆一、研究協力者藤咲喜文、研究協力者小西由貴範らが HL7V3 の RIM (参照情報モデル) に準拠した電子カルテシステムのデータモデルの開発方法を検討し、処方を中心にデータモデルの開発を行った。

4. 1 データモデル開発手順の検討

HL7V3 に準拠した電子カルテシステムのデータモデルを実現するために、次の開発手順を考案した。

- (1) データモデル化する対象領域の決定
- (2) 対象領域に対し、対象ドメイン専門家を交えた詳細なユースケースの記述とモデル化 (UML ユースケース図、UML アクティビティ図の作成)
- (3) HL7V3 の知識を基に(2)で作成されたユースケースに対するデータモデル (UML クラス図、HL7V3RIM のクラス名や分類等に対応) の記述
- (4) 以上の成果物をもとにユースケース毎に詳細なシナリオを作成し、相互作用 (アクター間でのやりとり) を洗い出し、具体的にやりとりされるオブジェクト (情報) として整理
- (5) (4)で作成された詳細シナリオに対応する HL7V3 のシナリオ (ストーリーボード) を探し、詳細に対応関係を確認
- (6) 同じく(4)で洗い出されたオブジェクトと HL7V3 で用意しているメッセージ (HMD) との対応関係を確認
- (7) 上記により開発されたデータモデルを検証のため実装設計し、実動作することを確認、必要に応じ上記の中から関連モデルを修正、クラス説明書の記述

(8) 上記により開発されたデータモデルを、電子カルテのための主要ドメインモデルとして整理・蓄積、統合し、トータルな電子カルテシステム用データモデルとする

4. 2 対象ドメインの検討

電子カルテシステムで使用する情報は診療情報から患者管理、業務管理、病院経営のための情報まで、多岐にわたり、その膨大な情報を一度に完成することは不可能である。どのような開発単位に分け、どのような順序で開発していくかも非常に重要である。また、標準化という視点では、既に情報交換のための標準が存在しているドメインを対象とすることが望ましい。また基本的な事項として、開発された単位が、幅広く、有益に使われるためには対象データの客観性 (記録者の主観が入らない、誰にでも受け入れられる) が重要であると考えた。また、今後の開発を進めるにあたり、ある程度の範囲の情報を含むドメインがパイロット開発として有用として考え、処方関連のデータモデルを最初に開発することとした。

4. 3 処方関連データモデルの概要

厚生労働科学研究飯田班「電子カルテ導入における標準的な業務フローモデルに関する研究」で検討・整理されている業務フローモデルと、同作業で得られた情報項目を参考に、改めて処方専門家 (薬剤師資格を持つ3人の研究協力者) とともに処方関連業務を分析し、次の17のユースケース記述を行った。

- (1) 診療を行う (初診時)
- (2) 初診時のインタビューを行う
- (3) 外来患者に薬剤を処方する
- (4) 診療を行う (再診時)
- (5) 再診時のインタビューを行う
- (6) 診療を行う (入院時)
- (7) 入院患者に薬剤を処方する
- (8) 処方に間違いがないことを確認する
- (9) 薬局が調剤を行う
- (10) 院内処方指示せんを受け付ける
- (11) 疑義照会を行う
- (12) 服薬指導を行う
- (13) 継続投与患者に服薬指導を行う

- (14) 処方鑑査を行う
- (15) 投薬を実施する
- (16) 入院患者に配薬する
- (17) 薬局でインタビューを行う

このユースケース記述をもとに、種々の粒度のユースケース図を21種類、概要レベルのアクティビティ図を6種類、ユーザ目的レベルのアクティビティ図を6種類作成した。さらに、データモデル(クラス図)を18種類作成した。また、ユースケース図に対応したユースケースシナリオを4種類作成し、対応するシナリオ(ストーリーボード)をHL7V3のBallot6パッケージより探し、対応するHMDとともに作成したシナリオ、データモデルと比較検証した(HL7V3Ballot6パッケージになかったHMDはVisioToolおよびRoseTreeなどHL7協会が用意するツールを用いて導出した)。さらに検証後のデータモデルをインターシステムズ社のCasheに実装し、実装上の問題点などを検討、データモデルに反映した。

4. 4 まとめと今後の課題

一連の検討・開発を通して、今回策定した開発手順でHL7V3対応のデータモデルが開発できることが確認できた。

今後の課題としては、研究協力者村上英が並行して進めているCDA(Clinical Document Architecture)と呼ばれる診療情報モデルに準拠した形での退院時要約ならびに診療情報提供書のデータモデル開発の成果を、本データモデルに反映させていくことがあげられる。また、引き続き、電子カルテ動向、ならびに電子政府などに対応した診療諸記録の電子化動向を鑑みながら、基本に戻り診療録データの性質、利用方法(特にオーダに伴う自動鑑査)等を検討して、有用性の高いデータモデルを標準として成長させ、維持・管理させることが必要である。特に、早急に必要最小限範囲を策定し、その部分の実現と普及を図ることが重要である。

5. 処理モデルの開発

分担研究者手島文彰らが、地域中核病院における電子カルテシステムを想定し、

RM-ODP(Reference Model・Open Distributed Processing)に沿って、処理モデルの開発を行った。

5. 1 処理モデルの作成手順

処理モデルは、以下の作業手順で作成した。

(1) 論理コンポーネントモデルの作成

論理コンポーネントモデルは、業務フローモデル(To-Beモデル)と情報モデルから導出される。まず、UMLのアクティビティ図で記述された業務フローモデルにおいて、システム化するプロセスにプロセスロール「Performer」を付加する。次に、「Performer」を付加したプロセスについて、具体的な処理手順を洗い出し、サブプロセスを定義する。サブプロセスを構成する個々の処理単位がプロセスコンポーネントとなる。次に、プロセス(機能)コンポーネントがその処理で使用する情報を操作するためのエンティティ(データ)コンポーネントを、データモデルから導出し、プロセスコンポーネントに接続した形で記述する。

実際のシステム開発においては、画面設計により論理コンポーネントが大きな影響を受けることが多く、ユーザインタフェースモデル作成の作業と処理モデル作成の作業は密接に連携しながら進められる。

なお、今回の処理モデル作成作業では、業務フローモデルに記述されたプロセスだけではなく、実際に稼働している病院情報システムの成果物も参照した。また、データモデルを並行して開発したため、業務フローモデル、データモデル、処理モデルで整合がとれていないところがある。また、ユーザインタフェースモデルについては、今回は特別な考慮をしていない。

(2) 論理コンポーネントのドメイン分割

今回は2つのドメインを想定する。1つのドメインは、エンドユーザと対話しながら処理を行うものであり、もう1つのドメインは周辺システムとデータ交換を行うものである。このようなドメイン分割は、多くの電子カルテシステムで採用されている。

(3)コンポーネントのノードへのマッピング

エンドユーザと対話しながら処理を行うドメインにおいては、J2EE に代表される 3 階層型アーキテクチャを採用することとした。また、周辺システムとデータ交換を行うドメインに対しては、メッセージングシステムを介した非同期型システム連携アーキテクチャを採用することとした。図 5-1 に、システム全体のアーキテクチャを示す。

次に、選択したアーキテクチャに従って、論理コンポーネントを配置していく。具体的には、3 階層型のアーキテクチャを採用したドメインにおいては、まず論理コンポーネントをクライアント側に配置するのか、サーバ側に配置するのかを決めていく。そして、サーバ側に配置した論理コンポーネントについては、さらにファサード、ビジネスロジック、エンティティの 3 種類に色分けしておく。ファサードとは、クライアントからの要求を受け、ビジネスロジックを担う論理コンポーネントに様々な処理を依頼するものである。ビジネスロジックとは、クライアントから送られてきた入力データやデータベースに格納されたデータを使用して、各種判断・比較・計算などの処理を行うものである。エンティティとは、データベースからのデータ取得やデータ格納を担うものである。

同様に、非同期型システム連携アーキテクチャを採用したドメインにおいては、論理コンポーネントをイベント送信側に配置するのか、イベント受信側に配置するのかを決めていく。

(4)論理コンポーネントモデルの拡充

アーキテクチャを考慮して配置された論理コンポーネントに対して、情報システムの基盤として必要な機能や法規格に準拠するための機能を追加していく。

なお、今回の処理モデル作成作業では、J2EE プラットフォームを利用することが想定されており、J2EE プラットフォームがすでに必要な基盤機能を提供していることがわかっていたため、論理コンポーネントの拡充作業は行わなかった。

(5)物理コンポーネントモデルの作成

物理コンポーネントは、配置場所ごとに論理コンポーネントを統廃合したものである。具体的には、トランザクション処理などのシステム構成上の制約や、詳細化されたデータモデル(データ構造やデータベース構造)を考慮して論理コンポーネントモデルを再構成する。

物理コンポーネントモデルとしては、基本的には実行単位ごとにコンポーネント構成、コンポーネント間の処理手順、コンポーネントのインタフェースを記述していく。

なお、今回の処理モデル開発作業では、サーバ側の実行単位を中心にモデル作成を行い、クライアント側についてはモデルを作成していない。また、現時点ではデータモデル(データ構造やデータベース構造)が作成中であるため、データモデルの反映作業は行っていない。

5. 2 処理モデルの概要

図 5-2 に、今回開発した処理モデルの作成範囲を示す。処理モデルは、実行単位ごとに、コンポーネント構成、コンポーネント間の処理手順、コンポーネントのインタフェースを記述したものであり、サーバ側に配置された物理コンポーネントについて 417 個のモデルを作成した。図 5-3 にコンポーネント構成の記述例を示す。図 5-4 に処理手順の記述例を示す。図 5-5 にインタフェースの記述例を示す。

5. 3 まとめと今後の課題

処理モデルは、特定のプラットフォームに依存するものではなく、多くのプラットフォームでも実現可能なものとなっている。

今後の課題としては、今回試作した処理モデルをもとに電子カルテシステムの実験システムを開発し、その妥当性を検証する必要がある。また、今回は他のモデルと整合がとれていない部分があり、今後の課題となっている。さらに、今回は開発していないクライアント側の処理モデルについても今後作成していく必要がある。

6. データベース構造モデルの開発

研究協力者藤咲喜丈らが、地域中核病院における電子カルテシステムを想定し、データベース構造の有り方に関する提案を記述した。

6. 1 データベース構造モデルの必要性

データの互換性を担保する上では、電子カルテシステムで使用する情報を格納するデータベース構造の参照モデルを作成し、その参照モデルを反映させた形で実際の電子カルテシステムのデータベースが設計されることが重要である。ここでいうデータベース構造の参照モデルとは、リレーショナルデータベースの「ビュー」に相当する概念である。

一般に、データベースには、将来の拡張を視野に入れ、アプリケーションが必要とする全情報を格納している。そして、データベースはスケーラビリティ、パフォーマンス、可用性などを考慮して最適なデータベース設計が行われ、そのデータベースの配置にも工夫が施される。このため、データの互換性維持のために、データベース構造を共通化することは現実的ではない。そこで、電子カルテシステムのデータベースとして共通的に存在すべき構造を「ビュー」として定義し、それらの「ビュー」に対して外部のアプリケーションから操作できるようにすることで、データの互換性を担保することが可能となる。

6. 2 データベース構造モデルの有り方

電子カルテシステムのデータベース構造の有り方を考えるにあたり、まず電子カルテシステムで管理すべき情報について考察する。

電子カルテシステムで管理すべき情報は患者の治療に伴い発生する診療情報と、患者の治療行為に関連する事務手続きのための諸情報を管理するための患者管理情報に大別される。また、診療情報は診療の流れに沿い次のように4種類に分類され、巡回し発生する。

(1) 観察情報：

問診、診察、検査結果などに加え、診断の際に参照される過去の診療情報、他医療機関等からの紹介状情報などが含まれる。診断に利用するための「事実」の集合体である。

(2) 診断情報：

観察情報を基に、担当医師、医療スタッフがどのような情報をどう利用して、どう判断したかを示す主観情報の集合体である。

(3) 診療計画情報：

診断時に、更にどのような検査が、追加確認が必要か、どのような治療を行うかの計画を扱う情報である。クリティカルパス、診療ガイドラインなどもここで参照する。

(4) 実施記録：

検査の実施に伴い発生するレポートおよび諸記録、治療計画に基づく治療実施記録、看護計画に基づく看護行為実施記録などを扱う。この記録をもとに、更に時間経過を経て、治療効果などとともに観察情報として利用する。

これらの電子カルテシステムで管理すべき情報の概要を図6-1に示し、これをデータベース構造の基本モデルと呼ぶ。

次に、この基本モデルの妥当性を検証するために、基本モデルと電子カルテシステムが提供する代表的な表示ビューとの対応関係について考察する。以下、その代表的な表示ビューと基本モデルとの対応関係を示す。

(1) SOAP 表示ビュー

S (Subjective : 主観データ)

O (Objective : 客観データ)

A (Assessment : 考察)

P (Plan : 計画)

SOAP として有名な診療記述の方法である。基本モデルとの対応関係は次の通りである。

S と O は基本モデルでは観察情報に相当し、そのデータ属性として主観的なものか、客観的なものかが現れる。A は基本モデルでは診断情報に相当する。P は基本モデルでは診療計画情報に相当する。

(2) 看護記録表示ビュー

従来、カルテ（医師の書く診療記録）と、看護記録は似て異なるものと扱われてきたが、観察の視点が違う（カルテ：疾患由来、看護記録：疾患とともに患者の人間性由来）だけであり、状況に応じて入力データ、表示データとも必要な情報が扱えるようにすれば同様に管理可能である。

このように、電子カルテシステムは、様々な医療職に、その利用目的に添った表示ビューを提供しているが、電子カルテシステムのデータベース構造の有り方としては、それらの表示ビューに偏らず、診療データが持つ本来の性質を活かした形で管理されるべきである。ただし、代表的な表示ビューに対しては、スムーズな画面展開を可能とすることは必要不可欠である。

6. 3 まとめと今後の課題

電子カルテシステムを再構築するにあたり、基本である扱いデータの範囲と利用のあり方について検討を行った。引き続き、チーム医療を推進する意味でも、職種を超えて情報を共通管理するとともに、診療データのより望ましい表示ビュー（組み合わせ）を模索していく必要がある。また、この基本モデルと実装の関係も明らかにしていく必要がある。特に、オーダに関する自動監査などを考えた場合に、自動監査（エンジン）が必要とする情報の範囲が従来に比べて格段に広がってきており、それを支えるデータベース構造についての検討が必要である。同時に、データベース構造の有り方を考える上では、セキュリティ管理上の配慮も検討する必要がある。

7. 実行モデルの開発

分担研究者長瀬啓介、分担研究者安光正則、研究協力者大内隆信、山岡紳介らが、筑波大学附属病院向け医療情報システムの開発経験をもとに、電子カルテシステムを構成するための実行モデルの開発を行った。

7. 1 実行モデルの概要

筑波大学附属病院では、J2EE(Java 2 Platform, Enterprise Edition)をベースとした3層アーキテクチャを採用したオーダエントリシステムを開発した。J2EEはエンタープライズシステムを開発するための標準規格であり、各種サーバサイド機能を提供している。J2EE準拠のコンポーネントを利用することで、開発者はネットワークプログラミングなどの複雑な開発作業から解放される利点があり、エンタープライズシステムの業務ロ

ジック部分に集中してシステム開発に専念できる。3層アーキテクチャは、クライアントアプリケーションとデータストアから業務ロジックを独立させることを特徴とするものであり、具体的には、以下の階層から構成される。

- ・プレゼンテーション層（クライアント）
ユーザに対する入出力、およびユーザのリクエストの発行を担当する。
- ・業務ロジック層（ファサード）
クライアントからのリクエストの受信、結果の送信、トランザクション管理、エンティティに対する操作を担当する。
- ・データストレージ層（エンティティ）
モデル内容の保持、ならびに業務ルールの実行を担当する。

実行モデルは、厳密な開発プロセスと実装レベルでの開発支援フレームワークをペアで提供することで、エンタープライズシステム開発時の業務ロジック構築部分に焦点をあて、開発者が業務ロジック実装を簡潔に行えかつ変更要求への迅速に対応できるようにすることを目指している。このために、3階層アーキテクチャにおいて、実行モデルは、分析・設計フェーズで作成されたモデルから各階層に配置するコンポーネント間の関連付けを自動的に生成し、モデルに基づいてコンポーネントを呼び出す仕組みを提供している。

このような実行モデルを導入することにより、以下のような効果が期待できる。

(1) モデル設計から実装までの一貫性

分析・設計フェーズで使用されたモデルからプログラムが自動生成される仕組みは、開発者に本来の業務モデルの開発に専念できる環境を与える。さらにモデルから実装がシームレスに生成されるため、モデルと実装間に食い違いが発生する可能性を軽減させることが出来る。

(2) コンポーネントの拡張性確保

設計したモデルに基づいた実装を実現可能にし、また3層アーキテクチャの各階層に配置されたコンポーネントに対する一定の呼び出し機能が用意されている共通フレームワークを提供している。これらの機能は、コンポ

ーネットの追加もしくは急なコンポーネント内部の業務項目の追加にも、システムの構造に手を加えることなく、柔軟に対応することが出来る。

7. 2 まとめと今後の課題

筑波大学附属病院で稼働中のオーダエントリシステムでの開発経験を通して、実行モデルの有効性を検証することができた。

今後の課題としては、XML ベースの通信プロトコルをサポートすることで、HL7 など XML ベースプロトコルとの連携を柔軟に行える仕組みを提供すること、ならびにまだ開発者に依存せざるを得ない業務ロジック実装部分でさらなる抽象化の仕組みを提供することがあげられる。

8. 技術モデルの開発

分担研究者佐部太也らが、経済的な観点から情報技術を選択する際の基準をについて記述した。また、電子カルテシステムの要件について、研究協力者岡田康と若原秀幸が記述した。

8. 1 情報技術選択基準

電子カルテシステムを実現する為には、各種の情報技術のなかから適切なものを選択しなければならない。選択を行う為には情報技術が適切に分類されていなければならない。例えば、ある機能を実現する際に用いる技術を選択する際に、特定の製品と規格を同じ視点で比較することは本来できないが、複雑な情報技術を扱う上ではそのような事が頻繁に行われている。情報技術について適切な分類を行うことは困難であるばかりではなく、分類方法についてのコンセンサスが得られなければならない。分類/分類方法を新たに開発するよりは、既に存在し広くコンセンサスが得られているものを再利用することが望ましい。

米国連邦政府が電子政府化政策の過程で開発した連邦エンタープライズアーキテクチャ(Federal Enterprise Architecture: FEA)の中で技術参照モデル(Technical Reference Model :TRM)が策定されており、これを情報

技術の分類/分類方法の基準とするのが望ましいことがわかった。今後の課題として、FEA/TRM を再利用して電子カルテの技術参照モデルを開発する。

情報技術の選択を行う上では、選択対象としての情報技術自体の分析/評価だけでなく、非技術的側面として情報技術を取り巻く環境についても検討しなければならない。とりわけ、電子カルテなどの医療情報システムを支える情報産業において、営利企業が製品やサービスを継続的に開発/提供できる健全な市場が育成されるよう考慮しなければならない。そこで、電子カルテの情報システム取引としての意義、電子カルテがコンポーネントベースとなる事による情報産業への影響などについて検討をおこなった。今後の課題として、非技術的側面についてのモデルを構築し、技術参照モデルとの相互関係を分析する事で情報技術選択についての指針を明確にすることが考えられる。

8. 2 個人情報保護対応

国内においては平成 15 年 5 月 30 日に「個人情報の保護に関する法律」が公布され、公布後 2 年以内に施行されることとなっていたが、平成 17 年 4 月より施行されることが決定されている。

この法律は、「個人情報の有用性に配慮しつつ、個人の権利利益を保護」することを目的にしており、個人情報を有効活用するための法律という位置付けになっている。その上で、個人情報を取り扱う上での基本理念として「個人情報は、個人の人格尊重の理念の下に慎重に取り扱われるべきものであり、その適正な取扱いが図られなければならない。」とうたっている。

医療機関の殆どはこの法律で定められている「個人情報取扱事業者」に該当することになると思われるため、「個人情報取扱事業者の義務」が課せられることとなる。このため、個人情報取扱事業者の義務と、電子カルテシステムの対応要件を整理し、機能要件として明確化することは、ベンダのみならず多くのエンドユーザにとって非常に重要な作業と考えられる。

現在は機能要件の検討に着手した段階であり、今後の検討作業を踏まえて、抽出されたセキュリティ要件を電子カルテシステムのシステム基盤として整理していくことを計画している。

8. 3 電子保存対応

電子カルテシステムの電子保存対応を考えるにあたり、セキュリティの観点から電子保存の機能要件を抽出する必要がある。このため、下記の作業で機能要件を抽出した。

(1)MEDIS-DC ガイドラインの細分化

MEDIS-DC によるガイドラインは、電子保存を行う際の機能要件と管理要件をまとめたものと考えられる。このうち、機能要件については「システムが備えるべき機能」と「システムの機能とは明記せず運用も含めて実現すればよい機能」の2種類が混在して読み取れるが、これらをまとめて機能要件と考える。また、機能実装と実運用の観点から見ると、各機能要件は複数の機能要件から構成されていると読み取れる。そこで、MEDIS-DC のガイドラインの機能要件をアトムレベルに細分化する。

(2)細分化した機能要件のうち、セキュリティに特化されるものを識別する

以上の作業を行った結果、電子カルテシステムの電子保存のためのセキュリティ要件としては、26個の項目を抽出することができた。

今後の課題としては、これらの抽出されたセキュリティ要件を電子カルテシステムのシステム基盤として整理していくことがあげられる。

9. ユニット化とユニット開発フレームワーク

本章は、分担研究者手島文彰、研究協力者佐々木文夫、八川剛志、発知康、山本裕らがユニット化とユニット開発フレームワークについて研究班で検討した結果をもとに記述した。

9. 1 ユニット化の必要性

一般に、電子カルテシステムのソフトウェアを流通させる方法としては、Linuxのようなオープンソースのようなアプローチと、業種別に共通的に使われるソフトウェア部品を整備していくビジネスオブジェクトのようなアプローチが考えられる。オープンソース・アプローチは、コアとなるソフトウェアを維持・改良するような組織がない現状では、実際の普及・展開は難しいと思われる。一方、ビジネスオブジェクト・アプローチでは、既存の電子カルテシステムでも導入することができるようなものでなければ普及・展開しないと思われる。

次に、ソフトウェアの流通を考える上では、ソフトウェアの相互運用性を保証することが重要となる。任意のソフトウェアの相互運用性を保証することは、市場規模を考えても現実的ではない。そこで、あらかじめワークフローを決めておき、そのワークフローをサポートするソフトウェアを決め、それらのソフトウェア間の通信手順と通信方法を決めておくことで、ソフトウェアの相互運用性を保証することが現実的であろう。

以上のことから、電子カルテシステムを構成するソフトウェア間の相互運用性を担保するためには、ワークフローをサポートするソフトウェアと、ソフトウェア間のトランザクション(通信手順と通信方法)を技術仕様書として整備し、実際にソフトウェアの相互接続を試験にて確認することで、ソフトウェアがそのワークフローをサポートすることを保証する仕組みを確立する必要がある。そして、ワークフローをサポートするソフトウェアを、電子カルテシステムのユニットと呼び、ユニットを整備していくことが重要である。

このような考え方は、IHEですでに導入されているものである。IHEでは、当初放射線部門システムを構成する主要なシステム間の相互運用性を担保することを目的として、テクニカルフレームワークと呼ばれる技術仕様書を整備し、コネクタソンと呼ばれる相互接続試験の場を提供することで、IHE準拠のシステムのお墨付きを与えることで、マルチベンダによる放射線部門システムの構築を可能としている。

電子カルテシステムは、放射線部門システムとは異なり単一のシステムとして構築されることが多く、電子カルテシステムを構成するシステム間での相互接続を担保するというような考え方は存在しなかった。しかしながら、データの利用性や他システムとの接続性といった全体最適の観点からは、オープンな設計が求められるようになってきており、電子カルテシステムを構成するソフトウェアをユニットとして定義し、この仮想的な機能モジュール間の相互運用性を担保することで、オープンな設計となっていることは大きな流れとなっていくと思われる。

9. 2 ユニットの概要

ユニットは、トランザクションが発生する仮想的な機能構成単位である。ユニットは、実際の製品システムと一意に対応付けられるものではなく、1つの製品システムの中で複数のユニットが実現される場合もあれば、複数の製品システムで1つのユニットが実現される場合も考えられる。ユニットは、実体があるものではないので、どのようなユニットを定義すべきかが重要となる。このため、多くの関係者の中でのコンセンサス醸成が必要である。

9. 3 ユニット作成手順

以下に、ユニット作成手順を述べる。なお、今回は、今まで電子カルテシステムを開発してきた経験を通して得られた病院業務・運用に関する知識を十分活用して、典型的なワークフローを定義し、それぞれのワークフローで必要となるであろうユニットの定義を行った。

(1) ワークフローを定義する。

まず典型的なワークフローを決める。

(2) ユニットの定義する。

処理モデルにおいて、トランザクションの視点からユニット候補を導出する。

(3) ユニット間のトランザクションを定義する。

導出されたユニット候補をもとに、ユニットを決定し、ユニット間のトランザクションを決定していく。

図 9-1 に、実際に導出されたユニットの事例を示す。

9. 4 ユニット開発フレームワーク

普及促進という観点からは、ユニット開発を支援する開発フレームワークを提供することも重要である。ユニットを構成するコンポーネントは、システムによらず固定的なコンポーネントと、個々のシステムごとに変更する必要がある可変型のコンポーネントの2つに大別される。そこで、あらかじめ固定的なコンポーネントを開発提供しておき、可変型コンポーネントを入れ替えるだけで、ユニットが完成するような開発環境および実行環境を提供したものであれば、ユニット開発フレームワークである。図 9-2 に、ユニット開発フレームワークの概念図を示す。

このようなユニット開発フレームワークを整備していくことで、電子カルテシステムの開発経験の少ないベンダでも、初期投資を抑えながら他社を差別化するようなユニットの開発が可能となり、マルチベンダ型の電子カルテシステムをさらにエンドユーザーにとって魅力あるものにする原動力になると期待される。

D. 考察

以下のように、研究結果に関する考察を加えた。

(1) EHR モデルの開発動向をまとめたことで、本研究で採用する手法が世界的標準化動向の中で適切な位置付けにあることを検証できた。

(2) モデルおよびコンポーネントを用いた電子カルテシステム開発の枠組みとしては、①モデルからコンポーネントを導出するまでの開発手順を実用段階まで具体化した。②モデルおよびコンポーネントを参照・利用するために EA アプローチが有効であり、これまで本研究で採用した手法との親和性が高いことを検証した。

(3) モデルを用いた電子カルテシステムの開発に関しては、①開発したモデル（データモデル、処理モデル）は UML(Unified Modeling Language、統一モデル記述言語)

で記述されており、第三者でも容易に理解することができるだけでなく、他のモデルとの比較を行う上でも強力な道具になることが確認された。②実行モデルを整備することで、分析・設計フェーズで作成されたモデルとコンポーネントとのシームレスなつながりを実現できることが確認された。③電子カルテシステムを開発するための技術的基盤として、モデルを拡充し参照モデルとして提供することで、モデルを用いた電子カルテシステムの開発が効率化されることが認識された。

(4) モデルの評価として、①作成した電子カルテシステムの業務機能モデルを、今後ユーザサイドの評価を得てより質の高いものにすることが望ましい。②HL7V3のRIM(参照情報モデル)に準拠して電子カルテシステムで使われる情報項目についてのデータモデルを試作した。この作業を通じて、HL7V3のRIM(参照情報モデル)は電子カルテ用参照モデル開発の強力な拠り所となることを確認すると共に、HL7が臨床文書アーキテクチャを参照情報モデルに融合する作業中なので、この作業への参加を強化する必要性を確認した。③試作した実装モデルは非常に可視性が高く、また機能の層別が明快に行われているために、システム設計において強力な拠り所となることが認識された。

(5)コンポーネント流通の観点から、①大きな粒度での流通を考えることが重要であり、それをユニットと定義した。②ユニット間の相互利用性を保証する仕組みを整備していく必要があることが確認された。③ユニットの効率的な開発を促進するためには開発フレームワークの整備が有効であることを検証した。④リファレンスインプリメンテーションを開発することで、ユニット流通が促進されることが期待できる。

(6)本研究の周辺技術やコンテンツに関して、①電子カルテへの要求事項についてはJAMI、各医学会、ユーザ団体との協力が必要である。②HL7のどのメッセージを選ぶかは重要な決定であり、HL7メッセージの適用を検討あるいは検証・普及しているIHE-JやJAHISの各委員会との協力が不可欠である。③病名マスタ、医薬品マスタなど使えるマス

タ入手可能であることが重要であり、厚生労働省、JAMI(日本医療情報学会)等との連携強化が必須である。④情報項目はHL7参照情報モデル準拠を基本とするが、日本で使われている情報項目の整理は不可欠であり、JAMIやユーザ団体、個別に見識・情報を蓄積している先進ユーザとの連携が重要である。⑤新しい情報技術に関してはebXMLやOMG、CBOP、INTAPなどのIT技術団体との連携が必要である。

E. 結論

以上の結果からモデルおよびコンポーネントを用いた電子カルテシステムの開発は以下のように、電子カルテ普及の大きな推進力になり得るものと考えられる。

(1)電子カルテシステム導入時のデータ互換性およびシステム間の相互運用性を保証するための仕組みとして、EA(Enterprise Architecture)アプローチを導入することは有効であり、これにより中長期的な視点から全体最適を考慮した電子カルテシステムの計画的な導入が可能となり、電子カルテシステム導入のための投資を最適化できることが期待される。

(2)EAアプローチを導入し、参照モデルや組織として共有すべきルールや標準を技術的基盤として整備していくことにより、今後開発が望まれるモデルおよびコンポーネントをベンダ間およびユーザ間で共用されることが可能となり、これにより電子カルテシステム導入時にその時点におけるベストプラクティスを具現化でき、ユーザ側の情報格差を解消することに大きく貢献することが期待される。(3)電子カルテシステムを構成するユニットを流通単位とし、ユニット間の相互運用性をIHE的なアプローチで保証することにより、高品質確保、導入容易化、短納期化、低コスト化を誘導し、電子カルテシステムの導入促進に大きく貢献することが期待される。

(4)電子カルテ導入のために当研究では直接扱っていない課題に関して、以下のような各種の活動との連携が重要である。①電子カルテへの要求定義に関するユーザ関係団体との連携、②適用するメッセージに関する、関

係研究班や各団体との協力、③情報項目に関するユーザ団体、先進ユーザとの連携、④個人情報保護やセキュリティ技術ほかの先端ITに関する、関係研究班やJAMI、INTAP、OMG との連携。

F. 健康危険情報

本研究では直接国民の生命・健康に危険を及ぼす内容を扱わないため、本項目には該当しない。

G. 研究発表

1. 論文発表

- (1)高田 彰, 中井 幹爾, 永井 肇, 手島 文彰, 成松 亮, 藤咲 喜丈: モデルとコンポーネントを利用した電子カルテシステム開発, 医療情報学 23(Suppl.):83-84, 2003.
- (2)藤咲 喜丈, 成松 亮, 大林 正晴: HL7V3 の RIM を用いた電子カルテシステム用情報モデルの開発, 医療情報学 23(Suppl.):94-95, 2003.
- (3)手島 文彰, 永井 肇, 中井 幹爾: コンポーネント型電子カルテシステムの開発フレームワーク, 医療情報学 23(Suppl.):89-93, 2003.
- (4)長谷川 英重: 標準的電子カルテシステム向け HL7 レポート, 医療情報学 23(Suppl.):96-97, 2003.
- (5)村上 英: 診療文書国際標準 HL7 CDA(Clinical Document Architecture), 医療情報学 23(Suppl.):100-101, 2003.
- (6)大内 隆信, 安光 正則, 網野 貴文, 五十嵐 徹也, 高田 彰, 長瀬 啓介, 大野 国弘: ユースケースによるリスク管理型ソフトウェア開発手法を用いたオーダリングシステムの構築, 医療情報学 23(Suppl.):98-99, 2003.
- (7)高田 彰, 長瀬 啓介, 大内 隆信, 網野 貴文, 五十嵐 徹也: 病院情報システム開発での UML の利用による、要求要件に関する円滑なコミュニケーションの実現, 医療情報学 23(Suppl.):85-86, 2003.
- (8)長瀬啓介, 高田彰, 五十嵐徹也, 大内隆信, 網野貴文, 大野国弘: Java2Enterprise Edition を用いた推論エ

ンジンをもつ病院情報システムの開発, 医療情報学 23(Suppl.):225-227, 2003.

- (9)大野国弘, 長澤勲, 梅田政信, 長瀬啓介, 高田彰, 五十嵐徹也: 臨床判断支援システムのための知識ベースの開発, 医療情報学 23(Suppl.):490-494, 2003.

2. 学会発表

- (1)高田 彰, 中井 幹爾, 永井 肇, 手島 文彰, 成松 亮, 藤咲 喜丈: モデルとコンポーネントを利用した電子カルテシステム開発, 第 23 回医療情報学連合大会 (第 4 回日本医療情報学会学術大会), 2003.
- (2)藤咲 喜丈, 成松 亮, 大林 正晴: HL7V3 の RIM を用いた電子カルテシステム用情報モデルの開発, 第 23 回医療情報学連合大会 (第 4 回日本医療情報学会学術大会), 2003.
- (3)手島 文彰, 永井 肇, 中井 幹爾: コンポーネント型電子カルテシステムの開発フレームワーク, 第 23 回医療情報学連合大会 (第 4 回日本医療情報学会学術大会), 2003.
- (4)長谷川 英重: 標準的電子カルテシステム向け HL7 レポート, 第 23 回医療情報学連合大会 (第 4 回日本医療情報学会学術大会), 2003.
- (5)村上 英: 診療文書国際標準 HL7 CDA(Clinical Document Architecture), 第 23 回医療情報学連合大会 (第 4 回日本医療情報学会学術大会), 2003.
- (6)大内 隆信, 安光 正則, 網野 貴文, 五十嵐 徹也, 高田 彰, 長瀬 啓介, 大野 国弘: ユースケースによるリスク管理型ソフトウェア開発手法を用いたオーダリングシステムの構築, 第 23 回医療情報学連合大会 (第 4 回日本医療情報学会学術大会), 2003.
- (7)高田 彰, 長瀬 啓介, 大内 隆信, 網野 貴文, 五十嵐 徹也: 病院情報システム開発での UML の利用による、要求要件に関する円滑なコミュニケーションの実現, 第 23 回医療情報学連合大会 (第 4 回日本医療情報学会学術大会), 2003.
- (8)長瀬啓介, 高田彰, 五十嵐徹也, 大内隆

信, 網野貴文, 大野国弘:
Java2Enterprise Edition を用いた推論エ
ンジンをも有する病院情報システムの開発,
医療

情報学 23(Suppl.), 2003.

(9)大野国弘, 長澤勲, 梅田政信, 長瀬啓介,
高田彰, 五十嵐徹也: 臨床判断支援シス
テムのための知識ベースの開発, 第23回医
療情報学連合大会 (第4回日本医療情報学
会学術大会), 2003.

H. 知的財産権の出願・登録状況

1. 特許取得

該当案件なし

2. 実用新案登録

該当案件なし

3. その他

なし

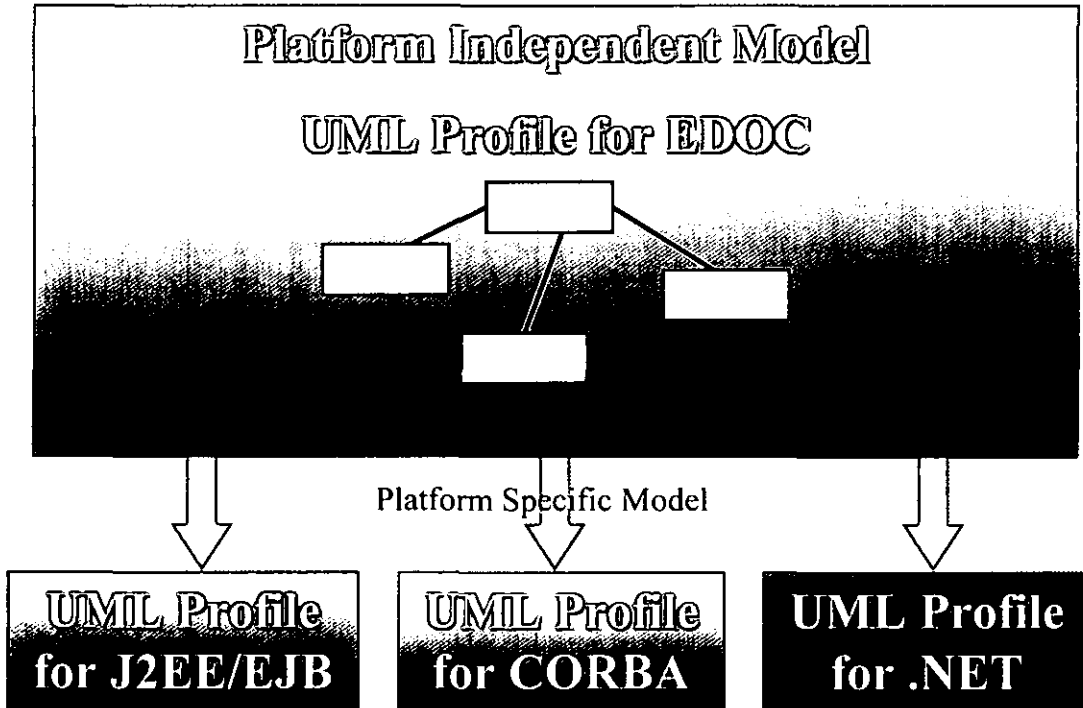


図 2-1 プラットフォーム独立モデルとプラットフォーム特定モデルの関連

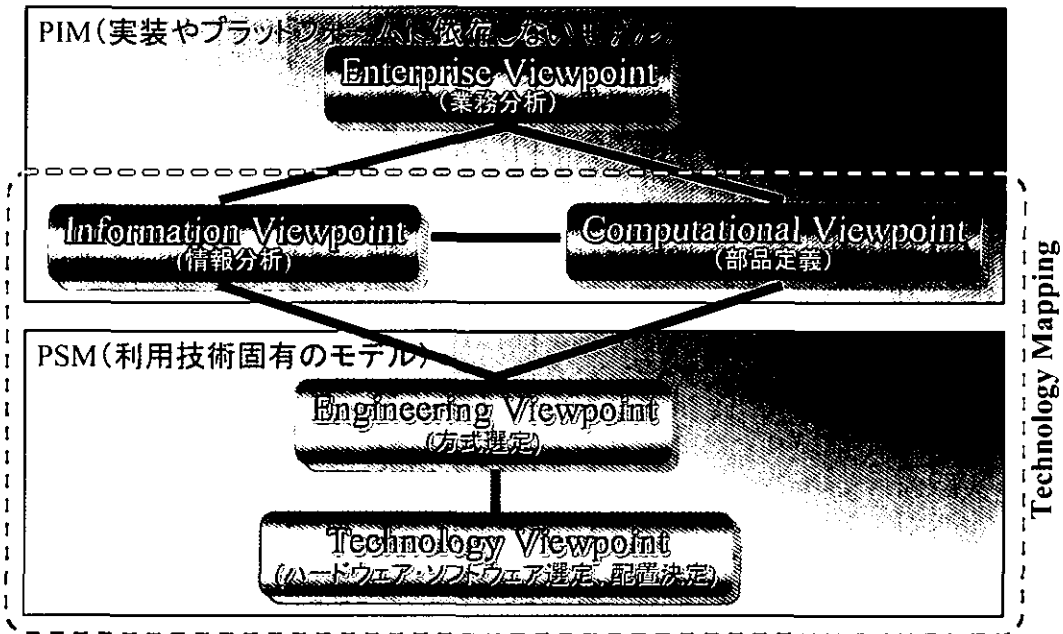


図 2-2 UML Profile for EDOC の全体図

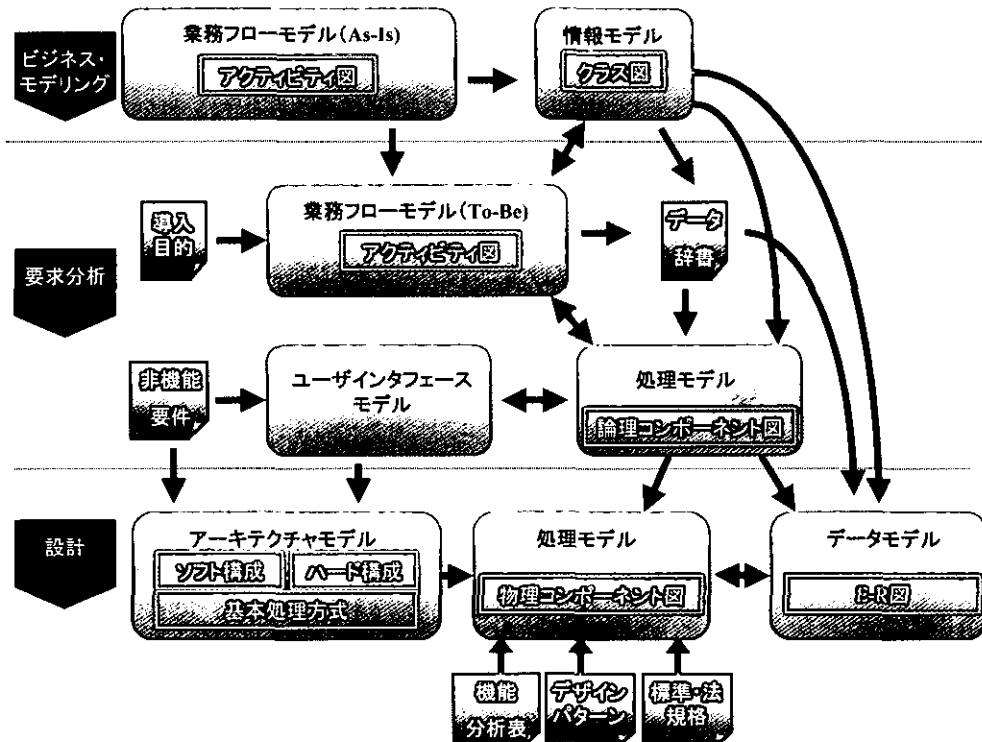


図 2-3 全体の作業プロセス

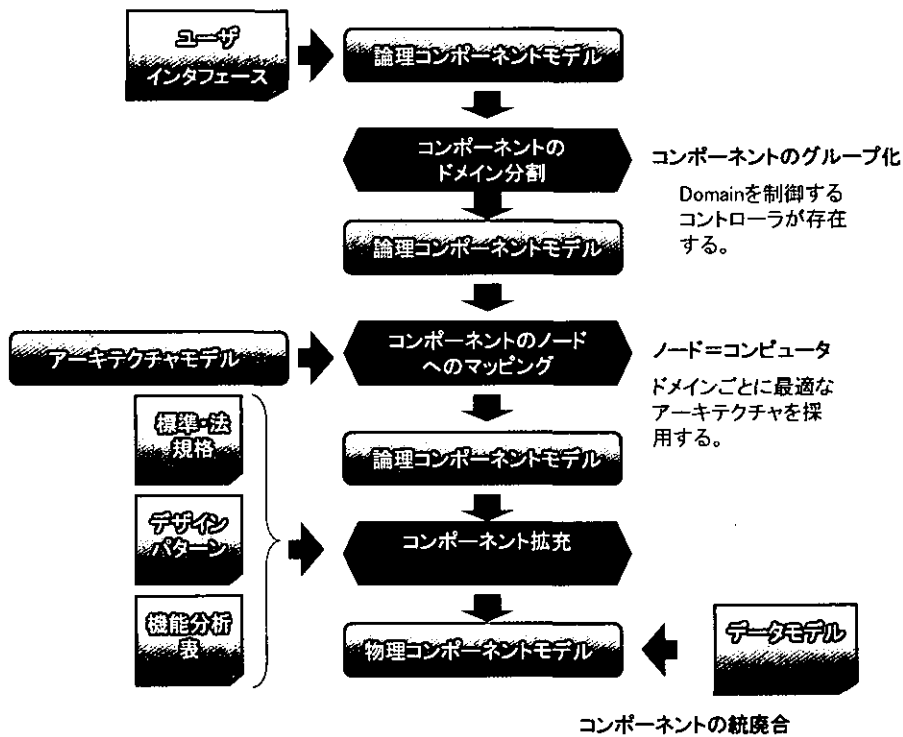


図 2-4 設計フェーズの作業手順

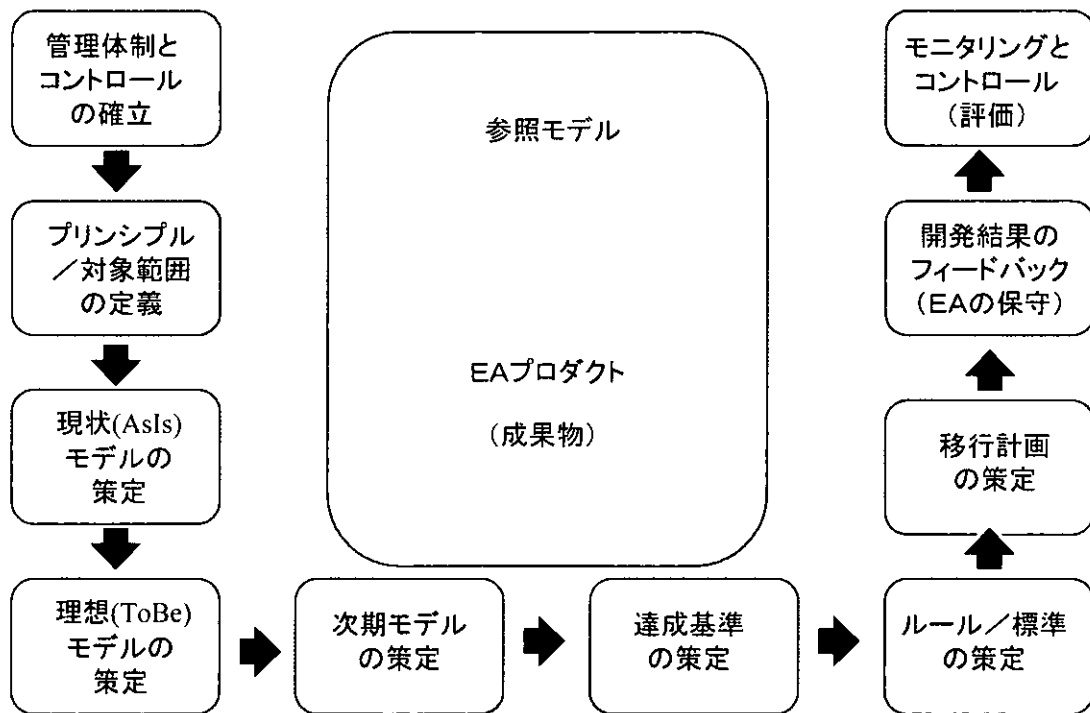


図 2-5 EA のプロセス

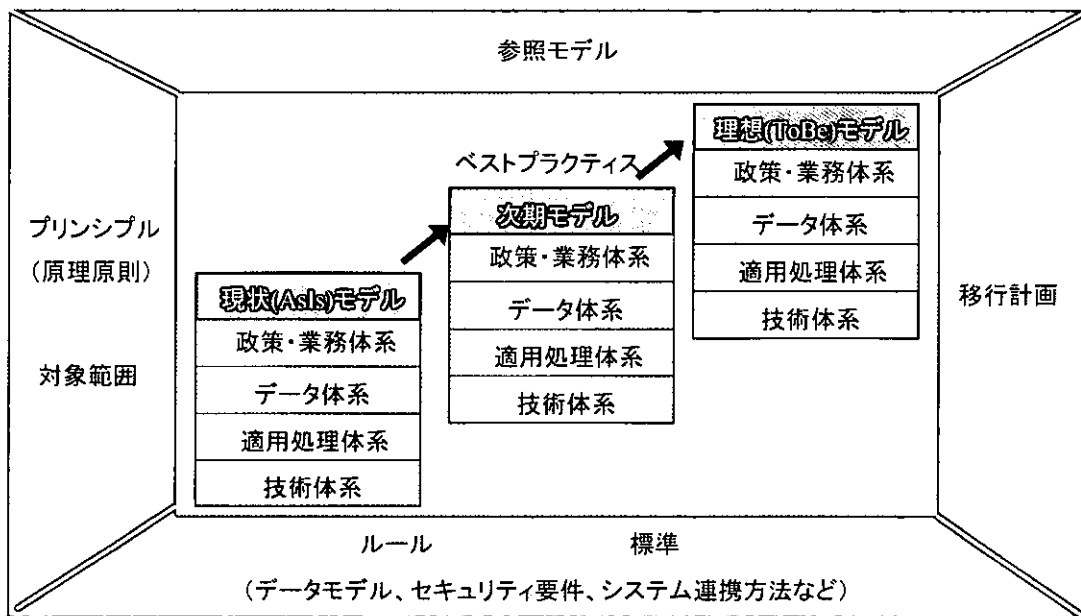


図 2-6 EA のフレームワーク

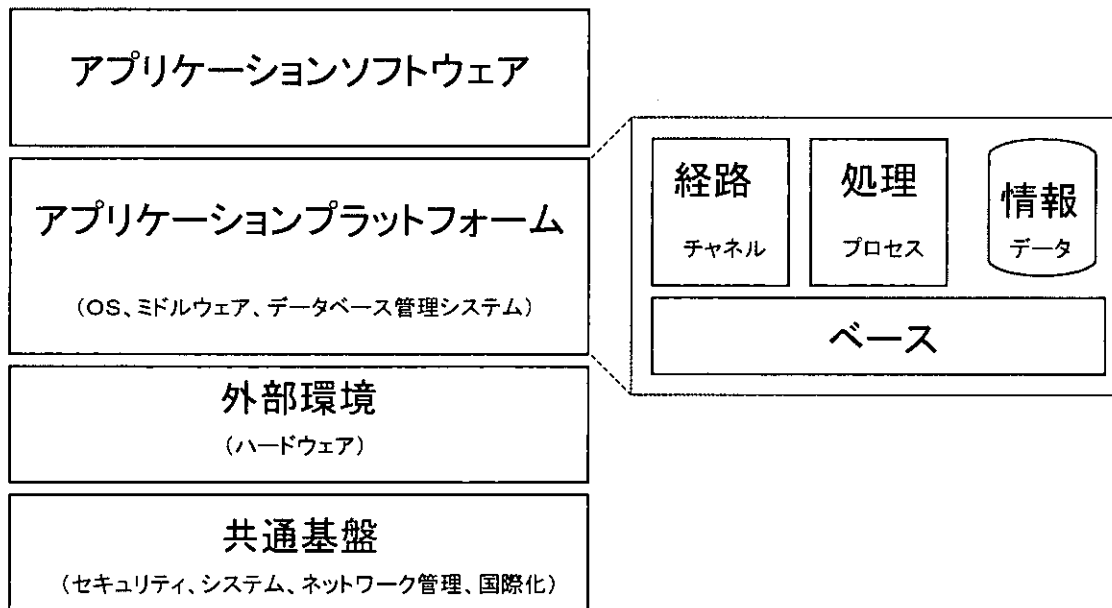


図 2-7 技術参照モデル

政策・業務体系	エンタープライズビューポイント
<ul style="list-style-type: none"> - 業務説明書 - 機能構成図 - 機能情報関連図 - 業務流れ図 	<ul style="list-style-type: none"> - 機能分析表 - 業務フローモデル - 業務フローモデル
データ体系	インフォメーションビューポイント
<ul style="list-style-type: none"> - 情報体系整理図 - 実体関連ダイアグラム - データ定義表 	<ul style="list-style-type: none"> - 情報モデル - データモデル - データ辞書
適用処理体系	コンピューテーショナルビューポイント
<ul style="list-style-type: none"> - 情報システム関連図 - 情報システム機能構成図 	<ul style="list-style-type: none"> - 処理モデル - 処理モデル
	エンジニアリングビューポイント
技術体系	テクノロジービューポイント
<ul style="list-style-type: none"> - ネットワーク構成図 - ソフトウェア構成図 - ハードウェア構成図 	

図 2-8 UML Profile for EDOC と EA の成果物対応関係

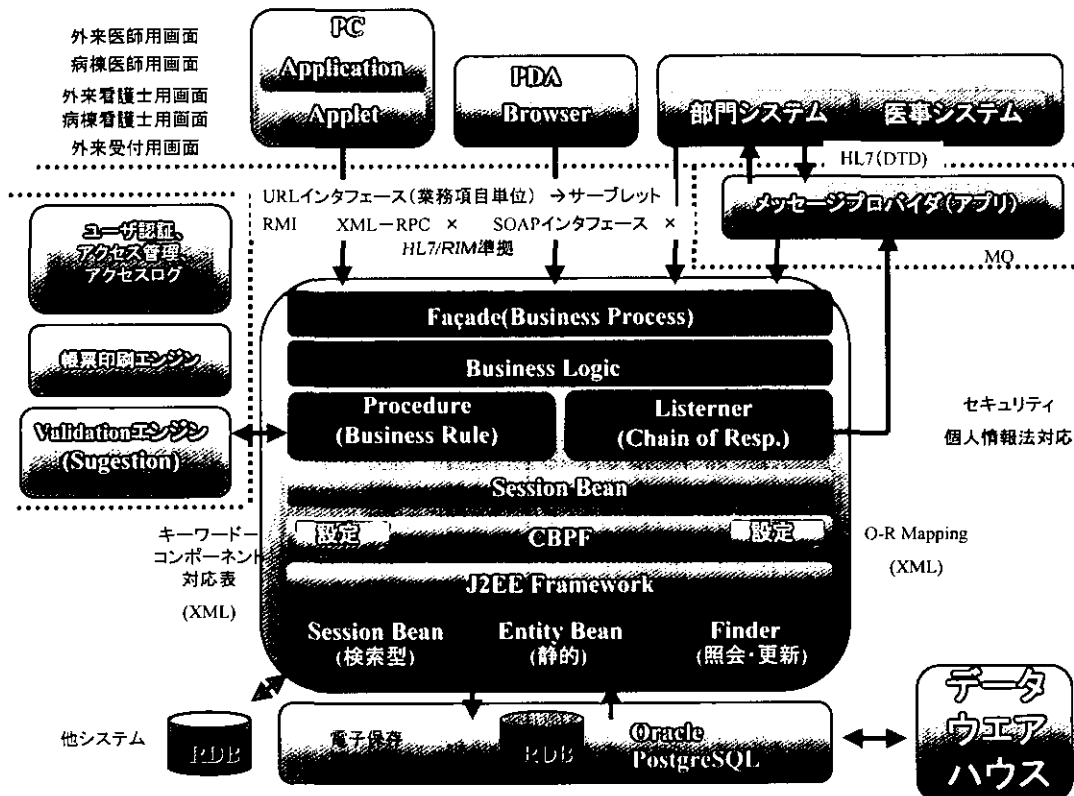


図 5-1 システム全体のアーキテクチャ

- オーダ
 - 注射
 - 処方
 - 処置
 - 検体検査
 - 放射線検査
 - 生理機能検査
 - 病理検査
 - 内視鏡検査
 - 病名
 - 服薬指導
 - リハビリ
 - 透析
 - 給食
 - 服薬指導
 - ホスピタルサービス
 - 看護 (観察指示)
 - 病棟看護
 - 入退院
 - 重症病棟
 - 集中治療室
 - 手術・麻酔
 - 患者所在予定管理
 - 病床管理
 - 予約
 - 外来再診予約
 - カルテ
 - 経過記録情報
 - 退院時要約
 - 診療情報提供
 - 患者情報
 - 患者基本情報
- モデル数: 417

図 5-2 処理モデル作成範囲

◇注射オーダ生成

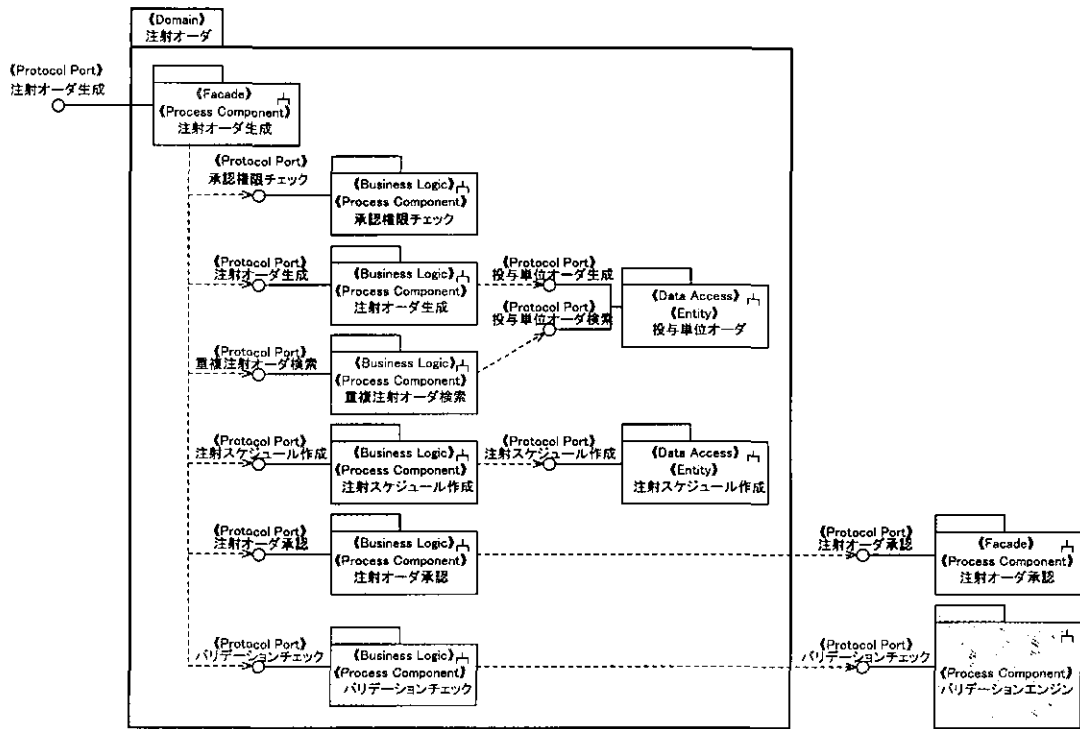


図 5-3 コンポーネント構成の記載例

◇注射オーダ生成

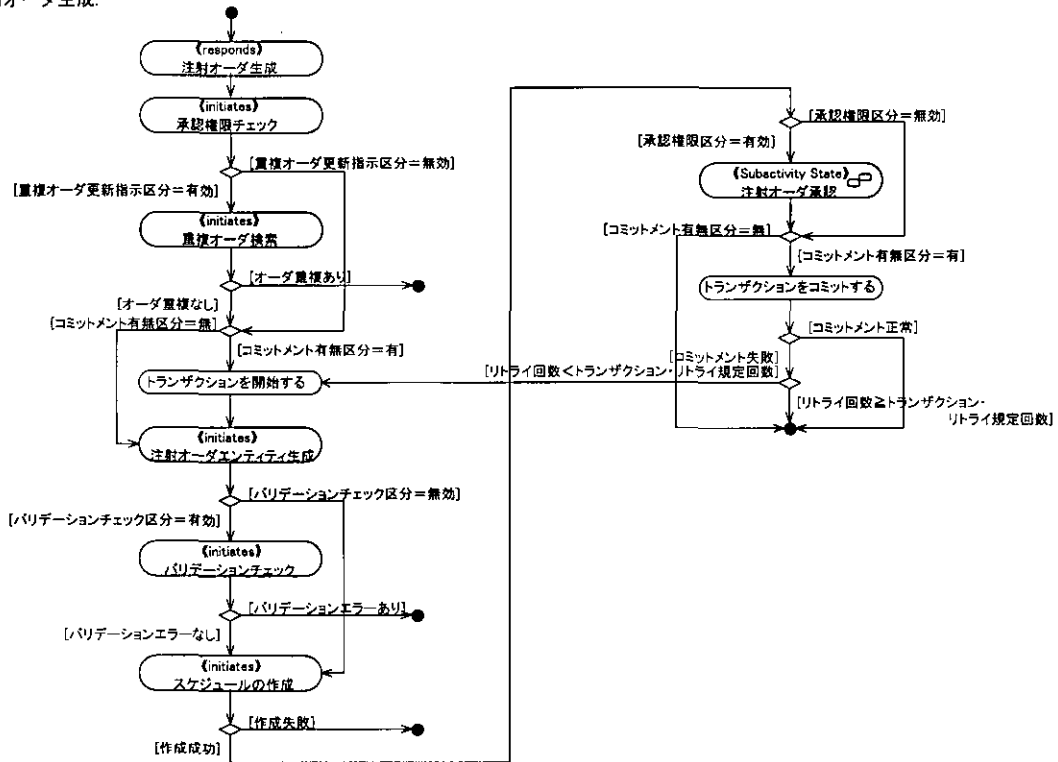


図 5-4 処理手順の記述例

ドメイン名		※入力: 受取 ロケータ, 出力: 出カ ロケータ及び戻値		※クラス名, 属性名及び項目名	
注カデータ		※クラス名, 属性名及び項目名		※クラス名, 属性名及び項目名	
エレメント名(ファサド)	入出力	クラス名	属性	備考	
注カデータ作成	入力		作成者	必須	注カデータを作成する。
			作成日時	必須	
			患者ID	必須	
			オーダー開始予定日時	必須	
			オーダー終了予定日時	必須	
			適料保険 有/無	必須	
			注カ手技	必須	
			注カレポート	必須	
			注カレポート説明区分		
			注カレポート説明コメント		
			注カレポート番号	手技=注カの場合必須	
			注カレポート番号適用区分	手技=注カの場合必須	
			投与速度	手技=注カの場合必須	
			投与順番	手技=注カの場合必須	
			配送部区分		
			投与薬剤		
			混合指示		
			混合指示コメント		
			注カ設備実施予定区分		
			投与指示コメント		
			パリエーションコメント区分	必須	
			コメント有無区分	必須	
			ドラッグクォンタリファイ規定回数	必須	
	出力	投与単位	オーダーID		
			バージョン番号		

図 5-5 インタフェースの記述例

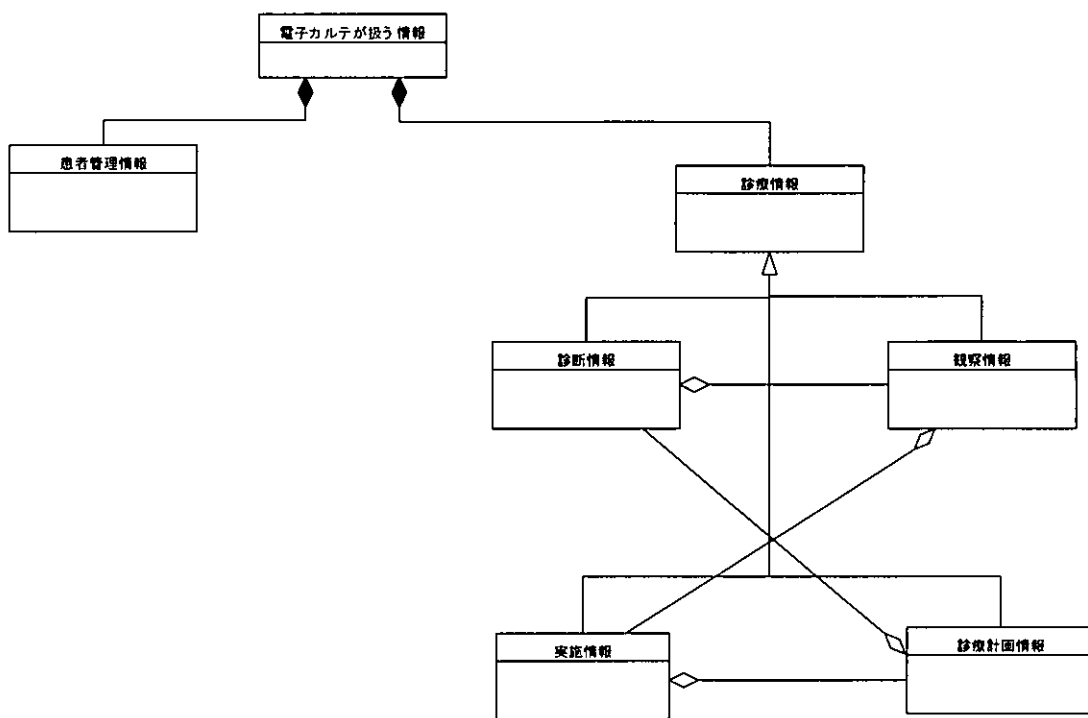


図 6-1 データベース構造の基本モデル

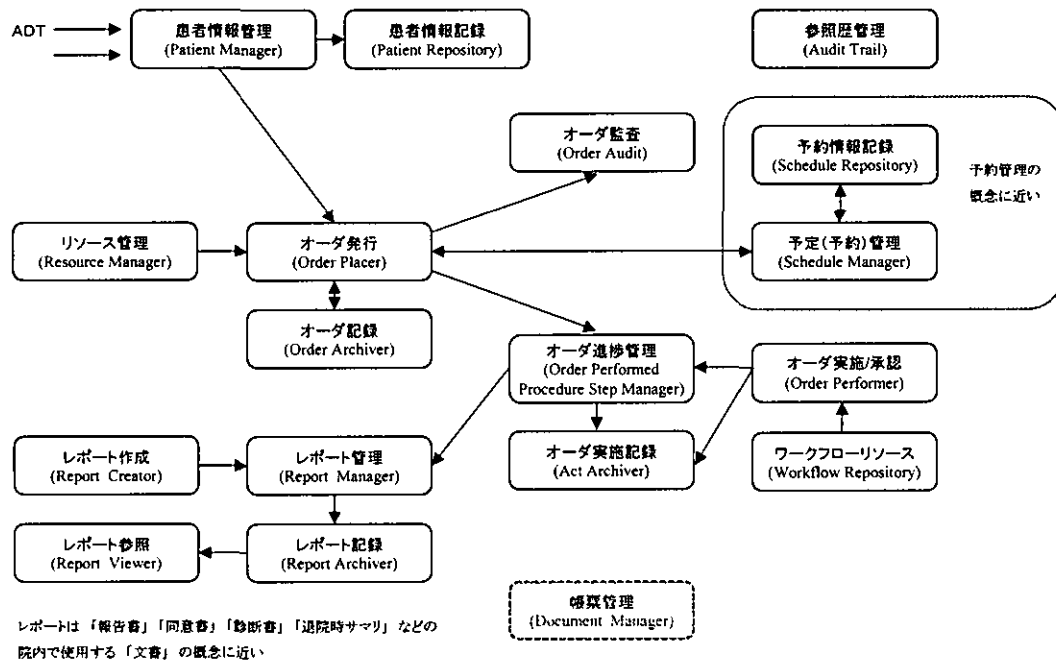
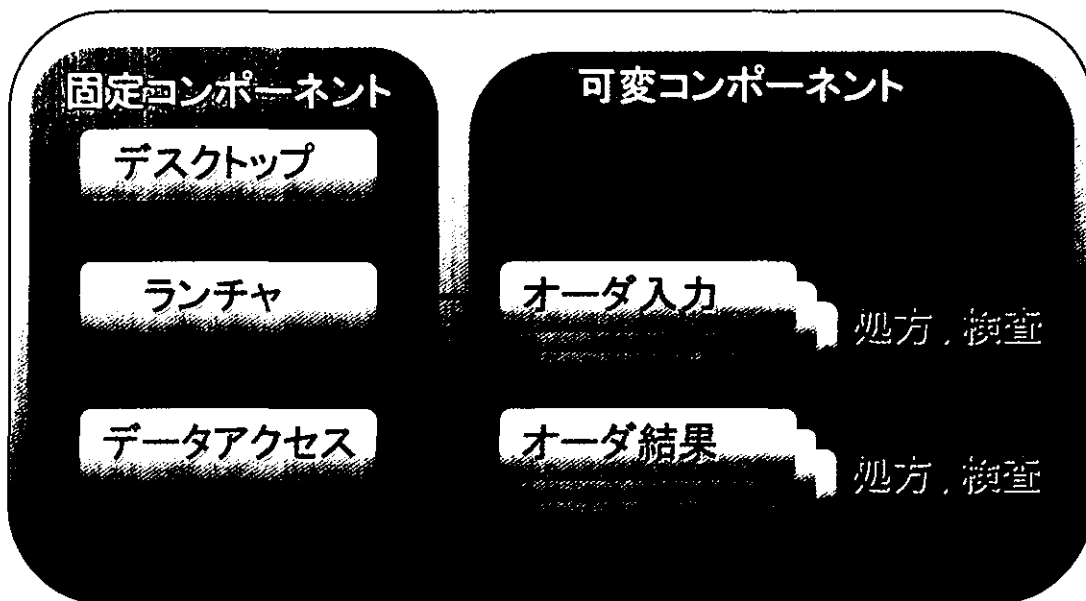


図 9-1 ユニット化の検討例



全体のコンポーネントの動作の仕組みは固定
 可変コンポーネントが交換できる仕組みを提供

図 9-2 ユニット開発フレームワーク