

## ソフトウェア設計情報共有化の為の ソフトウェア CALS システムに関する一考察

藤井 義信      清兼 幸雄      中村 正実      宮川 純一

富士通株式会社

〒211 川崎市中原区下小田中 2-12-5 中原ビル

通信ソフトウェアは、社会の多様なニーズに応える為にますます大規模化し、更にユーザーの急速な変化に伴い短期間で開発、提供する必要性が増加している。この為、設計情報を含むソフトウェア資産の有効活用を図り、ソフトウェア開発作業の効率化が急務である。ところが、ドキュメントに関してはその量の膨大さの為もあり、ソフトウェアエンジニアリングとしての手法はほとんど導入されていない。本稿では、文書記述の為に国際標準規約である SGML 及び CALS の考え方をソフトウェアエンジニアリングに導入したソフトウェア CALS によるグローバル開発支援環境について述べる。

## A Study of Software CALS System for Sharing Software Design Information

Yoshinobu Fujii      Yukio Kiyokane      Masami Nakamura      Junichi Miyagawa

Fujitsu Limited

Nakahara BLDG., 2-12-5, Shimokodanaka, Nakahara-Ku, Kawasaki, Kanagawa, 211 Japan

Communication software is now rapidly growing to meet various demands in our society. The ever changing demands from the user are constantly calling for immediate technical response and commercial availability of new software. To develop and supply communication software in a time as short as possible, we should effectively utilize existing software to enhance the efficiency in software development. However, the technique of software engineering has not been applied to sharing of documents partly because the volume of documents involved is usually enormous. The software CALS introduced the concept of CALS to software engineering. This paper explains the global environment for development support as realized with the software CALS. This environment facilitates sharing of design information and efficient document development with application software based on the SGML-the international documentation standards.

## 1. はじめに

通信ソフトウェアは、社会の多様なニーズに応えるため、ますます大規模化し、なおかつ短期間で開発、提供することが求められている。設計情報を含むソフトウェア資産の有効活用を図り、ソフトウェア開発作業の効率化を図ることが急務である。ところが、ドキュメントに関してはその量の膨大さのためもあり、ソフトウェアエンジニアリングとしての手法はほとんど導入されていない。

本稿では、CALCの考え方をソフトウェアエンジニアリングに導入したソフトウェアCALCによるグローバル開発支援環境について述べる。この環境の特徴は、文書記述のための国際標準規約であるSGMLを適用し、文書の標準化及びその特性を利用したアプリケーションにより、設計情報の共有化とドキュメントの開発を効率的に行うことができることである。

## 2. 通信ソフトウェア開発の課題

通信ソフトウェアのような大規模ソフトウェア開発において素早く顧客の要求に応えるためには、品質の保証された既存資産をいかに共有化することが重要になる。現在、通信ソフトウェア開発における設計情報の共有化に関しては、大きく分けて3つの課題が存在する。即ち、(a)複数開発拠点及び複数作業者間における情報の共有化、(b)次期開発のための再利用、(c)設計情報を利用した品質保証のための検証、である。

(a)に関しては、通信ソフトウェア開発は大規模かつ短納期であるため、開発形態は複数拠点に分散し、多人数の開発要員が協同で開発する分散並行開発となっている。この分散並行開発環境下において、スムーズに開発を行うためには、プロジェクト内で共有すべき情報へのアクセスが容易かつ高速に行う必要がある。即ち、空間的に離れた場所からネットワークを介して、セキュリティチェックや構成管理の行われている共有資産のデータベースにアクセスでき、様々な検索に適した情報形態と目的に合致した検索機能を有する必要がある。

(b)に関しては、通信ソフトウェアの開発において

は既存通信サービスの次期システムへの継承の必要性和ソフトウェア規模の大きさから既存プログラムの改造となることが殆どである。類似機能の開発/改造においては、既存機能のうち流用できる部分に関しては再利用すること及び以前開発した製品の生成過程で得られたノウハウを利用することが、効率的かつ正確な開発につながる。しかし現状は、特にドキュメントに関しては既存開発物の資産化は十分に行われているとは言えない。効率的なプロダクト生成のためには、ソフトウェア開発過程での生産物及びノウハウの双方の再利用が重要である。

(c)に関しては、近年ソフトウェアの品質保証を確実にするためにISO-9000への適合性が重要視されるようになってきた。それに伴い、生産物の品質を確認する手段として、視覚的に容易に確認できるドキュメントの重要性が増加している。標準化されたドキュメントの作成、及び設計書において製品の実現を段階的にトレースし確認することができる仕組みが必要とされている。

## 3. ソフトウェア情報の共有化

ソフトウェア開発において設計情報の共有化を進める場合、上で述べた課題を考慮すると、再利用を行うための開発体制/仕組み作りから、開発者の作業支援を行う応用機能までの総合的な環境を構築する必要がある。

### 3.1 情報共有化モデル

ドキュメントを中心とした総合的な効率化システムは、まず情報を電子化するための基盤環境の整備を行い、次に開発情報を共有できる形で蓄積していき、更にその情報を有効に活用するというステップで構築される。この各ステップを実現するための仕組みを3つの階層に分け、図1のようにモデル化した。

#### Layer 1: コミュニケーション層

ドキュメント情報を共有化するための基盤としては、設計情報の分散蓄積/集中管理により必要情報に自由にアクセスできるデータベース管理、拠点間での情報転送、様々なアプリケーションに対応した情報への形式変換等のコミュニケーション環境

の整備が必要である。

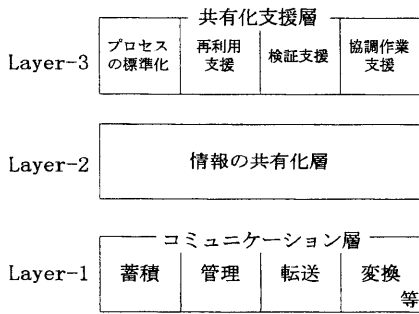


図1 情報共有化モデル

### Layer 2: 情報の共有化層

誰が作成しても高品質なドキュメントの作成やドキュメントの再利用のためには、文書構造の標準化及び記述レベルの統一が重要である。また、管理、転送、形式変換等の電子的処理のためには、互換性を持った文書の構造化を行うことが必要になる。

### Layer 3: 共有化支援層

標準化された設計情報を有効に活用するためには、目的に応じて設計情報の共有化を支援するアプリケーションが必要になる。以下のような支援機能が考えられる。

#### (a) プロセス標準化支援

設計情報の共有化と共に、作業プロセスの標準化も開発作業の効率化につながる。プロセス標準化支援には、プロジェクト毎のドキュメント執筆規約の内容を執筆中のドキュメント構造に合わせて案内する設計書執筆ガイダンス機能等が考えられる。

#### (b) 再利用支援

既存生産物の再利用は開発作業の効率化、信頼性を高めるものである。生産物の再利用支援機能には、上流工程から下流工程へ設計情報の継承を行う設計情報連携機能、仕様書から試験書を半自動的に生成する試験情報作成機能、各種設計情報をキーにして必要なドキュメントあるいはドキュメント内の必要部分等を検索する検索機能等が考えられる。

#### (c) 検証支援

開発の各段階において実現されていくソフト

ウェアの機能の実現過程に関する記述は、その流れを追うことによりソフトウェアの品質を検証するための有効な情報となり、一種の情報の再利用と考えることができる。この品質検証を効率的に行うためには、開発フェーズに従って情報をトレースする品質検証支援機能が必要となる。ここでは、各工程での設計書に実現過程に関する情報を正確に記述することが前提条件となる。また、ドキュメントが標準化されていれば、その構造を検証することによりドキュメント品質の向上を図ることもできる。

#### (d) 協調作業支援

開発グループ内で行う各種協調作業をスムーズに行うことは、開発作業を効率的に行うためには重要である。協調作業支援には、開発過程で得られるレビュー結果、開発ノウハウ等の設計書には記述されない可能性のある情報を設計書と連携を持って保存し、グループ内で共有するコメント付加機能等が考えられる。

## 4. ソフトウェア CALS によるアプローチ

本節では、上で述べた情報共有化環境の実現のためのソフトウェア CALS によるアプローチについて述べる。

### 4.1 CALS とは

CALS とは、あらゆる企業/企業内部部門の間で製品の設計/開発/製造/流通/サービスがデジタルデータを用いて、共通のデータベースのもとで、リアルタイムに行うことができるネットワークング、あるいはその推進運動のことである。CALS は時代の要求と共に“Computer-Aided Acquisition and Logistic Support”, “Continuous Acquisition and Life-cycle Support”とその意味を変えてきたが、現在では“Commerce At Light Speed”を意味する言葉となっている。

CALS では情報流通を効率的に行うためにその情報を標準化することを推奨している。そのため、CALS は多くの国際標準を採用しており、ソフトウェア CALS で使用する文書交換のための記述規格が後述する SGML である。

## 4.2 SGMLの概要

SGML(Standard Generalized Markup Language)とは、文書記述と電子文書交換のための規約であり、ISO(International Organization for Standardization: 国際標準化機構)により国際標準規約として制定されている。SGMLでは文書の構造と体裁を分離し、文書記述の際には文書の論理構造だけを意識すればよいようになっている。文書の論理構造はタグと呼ばれるマークを文書中に付加することにより認識可能になる。SGMLは米国を中心にした CALS プロジェクトにおいても、線画規格の CGM, 2次元 CAD 規格の IGES 等と共に文書の記述標準として採用されていることから、標準化を目的とする文書記述の言語としての実績もある。また、コンカレント・エンジニアリングのキー・テクノロジーとしても注目を集めている。

SGML で記述された文書は3つの部分から構成されている。これらは、使用するコード体系等の別システムで利用する時に必要な情報を定義する SGML 宣言、文書の論理構造を定義する DTD(Document Type Definition: 文書型定義)、及び実際の文書内容を記述する文書インスタンスである。このうち、SGML 宣言と DTD は SGML の知識のあるドキュメント管理者がドキュメント種別毎に作成するもので、設計担当者が記述するのは文書インスタンスのみである。

## 4.3 SGMLを応用した情報共有

SGML を用いて開発ドキュメントを記述することで、前記の情報共有化モデルの各層の課題が実現できる。図2にSGMLの特徴と共有化モデルの実現との関係を示す。

### (1)コミュニケーション層の実現

Layer 1においては、SGMLの特徴を活かし、文書構造毎に管理する SGML データベースを用いてドキュメントを蓄積する。

SGML はもともと文書の電子化配布を目的とした記述言語であり、文書の論理的構造のみに注目し、体裁情報とは分離している。そのため、異なるプラットフォーム間においても体裁情報に影響されることなく論理内容だけを交換することが可能である。

また、文書の電子的処理(変換、加工など)にも優れているので、データベースで管理される情報の実体はただ1つで、様々な視点から複数の用途での利用が可能になる。

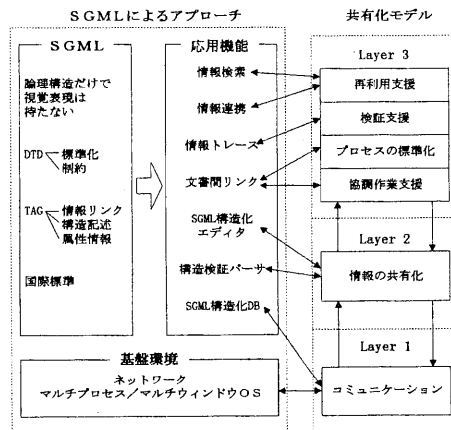


図2 SGMLの特徴と共有化モデル

### (2)情報の標準化層の実現

Layer 2においては、SGMLで記述された文書は DTD に従う制約から、文書種別毎に DTD に従った型で構造化され、ドキュメント体系の統一、文書記述の標準化が徹底される。作成された文書が正しい構造で記述されているかは、パーサと呼ばれるチェックプログラムによって検証可能である。SGML 構造化エディタを利用して執筆すれば DTD によって定義されている構造から外れた文書は作成されない。

### (3)共有化支援層の実現

タグには属性と呼ばれる付加情報を付けることができる。Layer 3においては、タグとその属性を利用した支援機能が実現できる。

例えば、タグの情報を基にドキュメント構造を加工し、異なる種別のドキュメントに内容を引き継ぐことができる。更に、タグや属性は検索のキーとすることができるため、文書全体が特定の一部分だけを必要に応じて検索することが可能となる。これらは設計情報の再利用を支援する。

また、タグとその属性を利用して、複数文書をリンクで関係付けることもできる。作成される文書と

執筆規約文書を文書間リンクでつなぎ、文書執筆中に関連する執筆規約を参照することにより、同一プロジェクトにおける設計書の執筆作業の進め方を統一することができ、記述レベルを統一すると共に、作業プロセスの標準化を進めることができる。同様に、レビュー結果や開発中に得られた情報をコメントとして文書化し、設計書と文書間リンクを張ることにより、同一の情報を複数の開発要員で参照することができ、プロジェクト内の協調作業を支援することができる。

#### 4.4 情報共有化アプリケーション

##### 4.4.1 ガイダンス機能

ガイダンス機能は3.1 情報共有化モデルでの(a)プロセス標準化支援や(d)協調作業支援を行うものである。設計書作成の参考となる執筆規約やサンプル文書をSGML化しデータベースに格納しておき、またDTDで定義されている設計書の構造と執筆規約等の構造の対応を定義した対応テーブルを予め作成しておく。設計書の新規作成時には、対応テーブルを基に属性として執筆規約等のドキュメント名、及び設計書の構造毎に対応する執筆規約等の構造のタグの識別子を設定する。この属性値を基に、文書間での構造のリンクを作成し、設計書作成時には必要に応じて現在位置に関連して知りたい情報を呼び出す。また、開発過程で得られたノウハウや設計書のレビュー結果等もコメント文書として別に作成することで、執筆規約同様に参照することができる。

##### 4.4.2 情報連携機能

情報連携機能は工程間で設計情報の連携を行うものであり、3.1 情報共有化モデルでの(b)再利用支援を行うものである。

工程間情報連携では、複数の設計書に記述される内容を1つの設計書だけに記述する。この記述の前後を外部から参照されることを示すタグで囲み、タグの属性としてそのタグを一意的に認識できる識別子を設定する。他文書では情報を継承したい場所でタグを挿入し、継承元となる設計文書名とタグ識別子を属性として設定することにより全くそのまま情報が継承されることを記憶する。このように、

上位工程文書から下位工程文書を生成する。情報の実体が1つのため、下位工程で情報の更新が行われても自動的に上位工程にフィードバックをかけることができる。

試験情報作成機能では、SGMLで記述された仕様書の構造とこれから作成される試験項目書/手順書の構造の対応を定義した変換テーブルを予め作成しておくことにより、この変換テーブルに従って、仕様書の構造を試験項目書/手順書の構造に変換する。仕様書の記述内容に対応した正確な試験書を半自動的に作成する。

##### 4.4.3 検索機能

検索機能は3.1 情報共有化モデルでの(b)再利用支援を行うものである。設計書中のコンポーネント処理、機能概要、コマンド説明、ブロック処理等の記述を検索用のタグで囲み、タグの属性に検索の種別を指定する。タグ属性に検索種別を持つことにより、開発作業で必要となる様々な検索の要望に対応することができる。

##### 4.4.4 情報トレース

情報トレース機能は、3.1 情報共有化モデルでの(c)検証支援を行うものである。

開発の各工程に作成される設計書において機能の実現に関する記述にタグを付加し、このタグと属性の情報を基に逐次検索することで、開発の流れに沿って実現されていく機能をトレースする。また、特定の機能に関する記述の工程間のクロスリファレンス情報を出力する。

##### 4.4.5 グローバル分散開発作業支援

グローバル分散開発作業支援は、3.1 情報共有化モデルでの(d)協調作業支援を行うものである。

ソフトウェアの分散並行開発では、常に最新の設計情報を参照する必要がある。しかし、海外拠点との間での分散並行開発では、日本語設計書と英語設計書(あるいは他の日本語以外の言語で記述された設計書)が存在することになるが、日本語設計書が変更されてもすぐに翻訳されないことが多く、英語設計書と最新版の日本語設計書の間で情報の差異が生じることがある。これでは正しい設計情報を共有できず、これを元に開発されたソフトウェアでは

高品質を保証することはできない。

そこで、従来2つの実体に分かれていた日本語設計書と英語設計書を1つの実体に統合(設計ドキュメントの日英複合文書化)し、日本語/英語どちらかの記述が変更された場合には他方の記述も自動的に更新、あるいは少なくとも変更があったことをユーザーに通知する情報(例:挿入/変更などの文字列)を自動記述する。これにより、日本語/英語の設計書で版数に差がでた場合でも、少なくとも古い情報を最新版と間違えて使用することがなくなり、開発するソフトウェアの品質が向上する。

## 5. システム構成

本節では、現在検討を進めている SGML を応用した通信ソフト向け情報共有化システムについて述べる。

### 5.1 機能構成

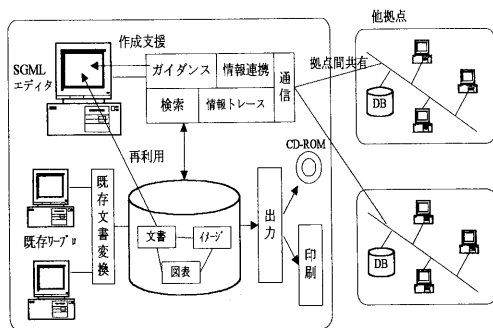


図3 機能構成

図3に本システムの機能構成を示す。これは、各自が自由に利用できるクライアントWS、複数拠点からのアクセスが可能なサーバWS、それらをつなぐネットワーク環境を基本に、SGMLでの記述を容易に行うことのできるSGMLエディタ、作成された電子文書を構造的に格納可能なSGMLデータベース、及び情報の再利用と検証を支援する各種アプリケーションからなる。

### 5.2 SGML データベース(SGML-DB)

SGML-DBは、SGML化された設計書の構造をノードとして認識し、ノード間の関係及び各ノードとテキスト内容の関係により構造化された設計書

を管理、格納する。ドキュメント全体ではなく、章/節/項などのドキュメント構造毎に版数、セキュリティ等を管理できる。

#### 5.2.1 目的、特徴

SGML-DBは、前に述べたように、SGML化された文書その構造を反映した形でデータベースとして管理することに特徴がある。管理の対象として、ソフトウェア開発で作成される設計文書に加えて、顧客納入文書、トレーニング用のマルチメディア・マテリアルなども格納できるよう開発が進められている。

情報共有のためには、グローバルに分散する各拠点で最新情報が確実に参照できることが重要である。SGML-DBでは、情報の定期的な同期メカニズムと必要に応じた即時通達が可能になっている。

また、WWW連携によりSGML-DB中の文書の参照が可能であり、電子メールその他との連携を含め、イントラネット技術を取り入れている。今後は、文書更新/レビュー/承認といった文書作成に関わる作業の支援、既存のプロジェクト管理システム、問題管理システム、試験管理システムなどとの連携が予定されている。

#### 5.2.2 機能

SGML-DBの機能は大きく分けると以下のものがある。この中には、SGML化された文書その構造を反映した形で格納するための、文書構造を解析するパーサ機能を含んでいる。

- ・文書の登録/参照/更新
- ・DTDや図などの外部実体ファイルの管理
- ・版数/構成管理
- ・アクセス制御
- ・分散DB制御

#### 5.2.3 システムアーキテクチャ

全体として、サーバ/クライアント構成である。APIを提供し、ユーザ固有のクライアント作成を可能にしている。また、サーバ側では、上で述べたような機能を持つSGML-DBを構築するため、以下のような三層から成る構造を取っている(図4)。

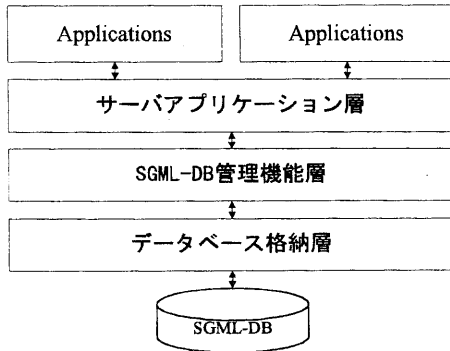


図4 システムアーキテクチャ

(1)データベース格納層

マルチメディアを含む様々な情報を容易に格納するため、オブジェクト指向データベース管理システムをベースとして利用している。

(2)SGML-DB 管理機能層

上で述べた文書の登録/参照/更新といった機能の実現と、そのために必要な文書や作業に関する基本データの管理/制御を行なうものである。

(3)サーバアプリケーション層

システム管理、WWW 連携などの他システム連携機能を、サーバ API を利用して、サーバ側機能として実現したものである。

5.3 SGML エディタ

SGML エディタは、SGML-DB と連携して、選択された構造のみを SGML-DB から取り出す。編集中のタグの挿入は DTD を解析し、構造的に記述可能なタグだけを選択させるので、間違った構造を記述することはない。編集後は更新前後の情報を比較し、変更差分だけをデータベースに追加する。また、パーサによる構造の解析も行う。

5.3.1 SGML 入力試行

SGML エディタの評価を行うために、SGML による設計書入力試行を行なった。その結果、以下の問題点が顕在化した。

- SGML 専用エディタを使用すると、タグと DTD を全く意識しない文書入力是不可能である。

る。可能な限りタグを意識させない入力システムの方が良い。

- 入力途中での文書構造の変更が困難かつ手間がかかる。規約を守りつつ容易に構造を変更できる入力システムが必要である。
- 図形とテキストは混在した形で表示される必要がある。また、編集も同一ウィンドウ内で行ってきた方がよい。
- 必要な部分だけのガイダンスを必要な時に瞬時に参照することができない。
- ドキュメント構造が不安定な時点で複雑な DTD を定義すると、構造が変更される毎に DTD や既存文書に変更が必要になる。

その他に、入力したいタグをどの位置で挿入できるかを表示する機能の不足、入力画面と印刷イメージの不一致、レスポンスの遅さ、操作の複雑さ、等が問題として挙げられた。これらの問題は、SGML 専用エディタが SGML の機能を活用することを優先させ、ユーザ要求やインターフェースの考慮が不十分であるためと思われる。

5.3.2 設計書編集エディタとしての必要条件

上記問題を解決するためには、ユーザが必要としている機能を考慮し、必要とされている機能が複雑な場合でも簡単なユーザインターフェースを提供する設計ドキュメント入力システムを構築し、使い勝手を向上させることが必要である。また、設計書用の DTD はドキュメント構造の変更に柔軟であることが望ましい (表 1)。

表1 設計ドキュメント入力環境の必要条件

項目	必要条件
設計書エディタ (SGML エディタ + 図形エディタ)	入力中での構造変更が容易 タグを意識せずに構造的記述が可能 図形とテキストが混在可能
入力支援ガイダンス	必要なガイダンスだけが瞬時に参照可能
文書型定義(DTD)	変更に対する柔軟性

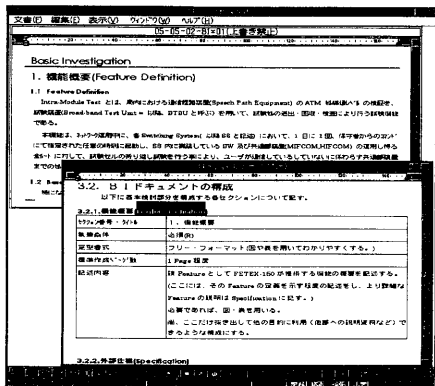


図 5 SGML エディタ画面例

### 5.3.3 SGML 統合エディタ

前記要求事項を解決するために我々は SGML 統合エディタを開発した。このエディタは普段使われたワープロの GUI で入力できるように、外部との API が公開されている市販エディタを使用して、市販ワープロから SGML 解析エンジンプログラム (DLL) を呼び出し、DTD に従った構造の編集を可能にした。このエディタではワープロで使用するスタイルと SGML のタグを対応付け、編集においてはスタイルのみを意識することで、WYSIWYG 環境で設計書作成ができるようにした(図 5)。

本エディタは現在設計書作成ツールとして社内適用中である。

## 6. これまでの成果と今後の課題

これまでワープロで作成してきた既存設計書、約 15,000 ページを SGML に変換した。既存文書は執筆規約が存在したにも関わらず、記述方法が執筆者によって異なっていたため、自動的に SGML 変換することは難しく、多くの手直しが必要であった。しかし、SGML に変換してしまえば、既存の資産を次期開発に再利用できるため、開発効率が向上する。SGML 化された設計情報は HTML、TEX、PostScript など様々な形式への変換が容易である。

現状では、SGML-DB については要件を満たすものが存在せず、ファイルサーバで代用している。ファイルサーバではフレキシブルに編集単位の指定が行えず、またファイルとしての管理しかできないため設計書としての構成管理が不十分である。

今後は、現在開発を進めている SGML-DB と SGML アプリケーションとの連携による文書管理統合システムを構築する予定である。このシステムでは、DB 上では 1 文書 1 実体として扱えるようにすると同時に版数管理、アクセス管理、構成管理等、様々な文書管理を行う。アプリケーション側では目次などを使ってその都度指定された文書中のある範囲を動的に扱えるようにすることで、開発者が使い易い文書編集/管理環境を構築する予定である。

## 7. おわりに

本稿では、大規模ソフトウェアを分散並行開発する場合の設計情報共有化のための問題点を整理し、コミュニケーション層、情報共有化層、共有化支援層の 3 層から成る情報共有化モデルについて述べた。また、このモデルに基づいたソフトウェア CALS によるアプローチとして SGML を応用した情報共有の実現方式について考察し、情報共有化に必要なアプリケーション機能として、情報連携機能、ガイドランス機能、検索機能、情報トレース機能、およびグローバル分散開発作業支援機能を挙げた。具体的なシステム実現例として、SGML エディタ、SGML データベースから構成されるソフトウェア CALS システムについて述べた。今後は、本稿において考察した設計情報共有化支援のための各種応用機能の実現と環境の整備を進め、文書だけでなく文書作成作業フロー等も含めた統合的な開発管理について構築していきたい。

### 参考文献

- [1] 藤井,他,"ソフトウェア設計情報の共有化方式の一考察",電子情報通信学会交換システム研究会,1993
- [2] 藤井,他,"ソフトウェア設計ドキュメントの入力の標準化に関する一考察",情報処理学会第 49 回全国大会,1994
- [3] 藤井,他,"SGML による日英設計書変更管理の実現方式",情報処理学会第 51 回全国大会,1995
- [4] 吉岡 誠 編著,"SGML のススメ",オーム社,1993
- [5] 吉岡 誠 編著,"SGML を使いこなす",オーム社,1996