

小笠原 秀人
OGASAWARA Hideto

會澤 実
AIZAWA Minoru

山田 淳
YAMADA Atsushi

ソフトウェアが大規模になったことと、分散処理技術の進歩がクロスオーバーしたことにより、ソフトウェアの分散開発が増えてきている。分散開発環境下でのソフトウェア品質管理を実現するために、当社はイントラネットと構成管理ツールを組み合わせた開発環境を構築した。このような開発環境の利用によって、分散した開発拠点間でのソフトウェアの品質評価、情報共有などが効率よく実現できる。

With the increasing size of computer programs and the growth in demand for distributed computer systems and related technologies, the need for distributed software development has inevitably arisen.

In order to manage and control the quality of programs being developed in different locations, we have built development environments combining the use of intranets and configuration management tools. Using such an environment, it is possible to efficiently evaluate software quality and share information among distributed development sites.

1 まえがき

開発対象システムの大規模化や複雑化に伴い、複数組織の共同によるソフトウェア開発が非常に多くなってきている。当然、複数の拠点にソフトウェア技術者が分散した分散開発の形態がとられることが多くなる。

ソフトウェア技術者の専門性と知識を物理的な距離を超えて集約する必要性と、ネットワークなどの分散開発の環境を支えるインフラの成熟により、ソフトウェアの分散開発は今後ますます増えると推測される。このため、ソフトウェア品質管理も分散開発環境下で実現することが要求されている。

当社では、エージェントメールシステム⁽¹⁾やリアルタイム電子会議支援システム⁽²⁾など、ソフトウェア分散開発環境の研究開発を行ってきた。そして、それらは実際のプロジェクトで利用されてきた。近年、イントラネットと Web ブラウザ (Internet Explore^(注1)や Netscape Navigator^(注2)) の急速な進展により、分散開発環境下で開発を行っている利用者から、共通インタフェースによる操作の実現、ソフトウェア品質管理のためのアプリケーションの充実などの要求があがってきた。

このような要求に対応するため、イントラネットと構成管理ツールを組み合わせた分散開発環境におけるソフトウェア品質管理環境を構築した。

2 イントラネットを利用したソフトウェア開発環境

当社では、分散開発環境下でも進捗(ちよく)管理や問題

点管理などの品質管理⁽³⁾を行うための環境を提供してきた。しかし、これらの環境では部門ごとに利用しているソフトウェア品質管理ツールとのリンクが難しいという問題があった。また、成果物の品質を自動的に確認したい、柔軟性の高い品質管理環境を構築したいという要求も強まってきた。

このような問題を解決し、分散開発環境下でのソフトウェア品質管理の要求にこたえるために、イントラネットを利用し、Web ブラウザをインタフェースとしたソフトウェア開発環境を提案する。

ソフトウェア開発環境をデファクトスタンダードである Web ブラウザをベースに提供することによって、各開発拠点独自のコンピュータ環境に依存しない開発環境を提供できる。

イントラネットを利用した分散開発環境のソフトウェア構成を図 1 に示す。各構成要素について次に述べる。

- (1) Mail-Daemon 利用者から出されたメールを受信し、alias^(注3) ファイルをチェックする。alias 定義が記述されていれば、それに従ってデータ登録プログラムを起動する。
- (2) WWW-Daemon 利用者の Web ブラウザから出された要求に応じて、CGI (Common Gateway Interface) を介してデータ登録プログラムやデータ抽出プログラ

(注 1) Internet Explore は、Microsoft 社の商標。

(注 2) Netscape Navigator は、Netscape Communications 社の商標。

(注 3) alias : 受け取ったメールを処理する方法 (例えば、データベースに登録する、メンバーに転送するなど) が定義されている。

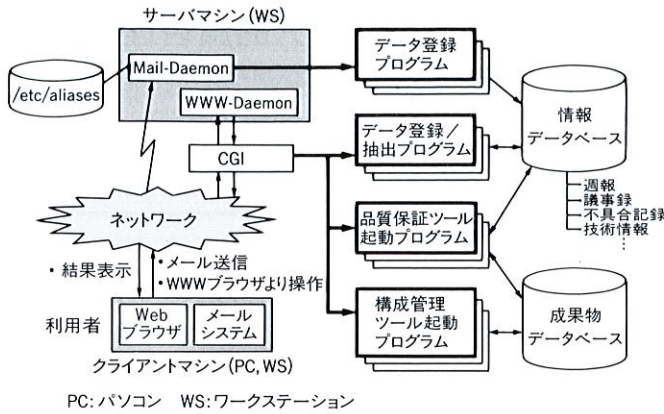


図1. インtranetを利用したソフトウェア開発環境 利用者はWebブラウザを利用して処理を行う。分散された拠点間で、構成管理された情報にアクセスできる。

Software development environment based on intranet

ムを起動する。

- (3) データ登録プログラム 利用者からメールで提出された週報、議事録、技術情報などの非定型情報や、Webブラウザから入力された不具合記録などを情報データベースに登録する。
- (4) データ抽出プログラム 利用者から要求された情報を情報データベースから抽出する。また、結果をHTML^(注4)化して、利用者のブラウザに表示する。
- (5) 品質保証ツール起動プログラム 利用者が指定する成果物に応じて、品質保証ツールを起動し、結果を情報データベースに登録する。
- (6) 構成管理ツール起動プログラム 利用者から要求された構成管理の要求(成果物の払出し/登録/バージョン一覧表示など)に応じて、構成管理ツールを起動し、成果物データベースから必要な情報を抽出する。構成管理ツールは、各開発部門で利用しているツールを利用する。

3 分散開発環境におけるソフトウェア品質管理

分散開発環境におけるソフトウェア品質管理環境を構築するために考慮した項目は以下のとおりである。これらの項目に対応させて、品質管理のためのアプリケーションを開発した。

3.1 構成管理機能

分散環境におけるソフトウェア品質管理では、成果物や開発過程で生成されるデータを確実に蓄積し、参照できるようにすることが前提となる。つまり、構成管理はこの開発環境でもっとも基本的な機能である。

(注4) HTMLは、Hyper Text Markup Languageの略語で、Webブラウザに表示させるための言語。

ソフトウェア開発では、二つ以上の類似した製品(例えば、上位機種/普及機種、顧客A向け/顧客B向けなど)を同時に開発することが頻繁に発生する。このような並行開発のために、成果物(ドキュメントやソースプログラムなど)の登録、払出しは当然として、その他ラベル管理や枝分かれ(派生)管理なども要求される。構成管理機能として実現している項目を表1に示す。図2に、登録、払出しのためのインタフェース画面を示す。

このような構成管理機能の実現によって、ドキュメントやソースプログラムのバージョンの取り違えなどを防止できる。

表1. 構成管理機能の内容

Functions of configuration management tools

項番	機能	内容
1	登録	成果物をコメントを付けて登録
2	払出し	登録されている成果物の払出し
3	ラベル付け	あるマイルストーン(テスト開始時点、リリース時点など)で、成果物すべてにラベルを付ける。このラベル単位での払出しも可能
4	枝分かれ	並行開発ができるように、ある時点のバージョンと同じ内容のソースコードを複製
5	比較	バージョンの違う同一ファイルの内容比較や、ラベル間で違いのあるファイル(修正された、削除されたファイルなど)を抽出
6	マージ	バージョンの違う同一ファイルを連結し、新しいバージョンを生成

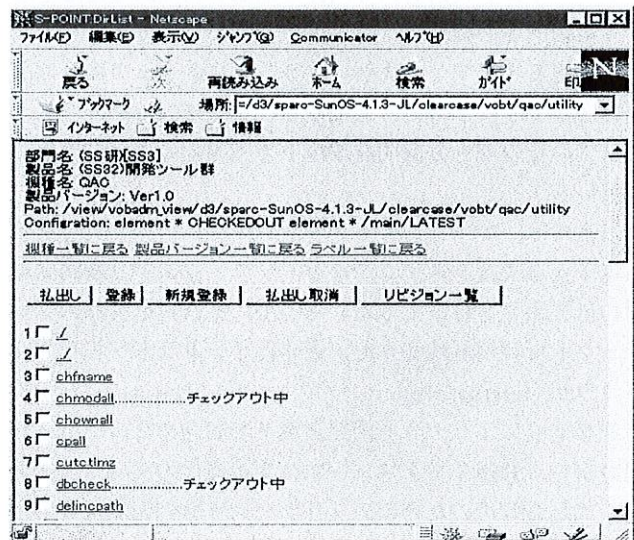


図2. 登録、払出し画面 登録、払出しをするファイルをチェックし、処理を実行する。

User interface for check-in and check-out

3.2 進捗管理機能

分散拠点間では、進捗を把握する会議を頻繁に開催できない。一般に、進捗管理は、ドキュメントのページ数、ソ

ースプログラムの変化量、テスト消化件数などの定量的な値を用いて評価されることが多い。

定量的な進捗管理のために、ソースプログラムのステップ数の推移などの情報を提供する。成果物を定量化するために品質保証ツールの一つであるメトリクス^(注5)計測ツール⁽⁴⁾を利用する。メトリクスの計測例を図3に示す。このような計測結果を時系列ごとにまとめて、利用者に見やすい形式に加工し、進捗情報として提供する。また、週報や議事録などの非定型の情報も参照できる仕組みになっているので(3.5章参照)、定量的な側面からだけでは判断できない作業の質や内容が把握できる。

```

Date : 98.4.10
Target Directory : /project/work

<メトリクス計測結果>
File Name  Module Name  ステップ数  条件文数  最大ネスト  ...
esq-com.c  main_loop      214        12        3
           autonum_sub   49         3         7
           dpram_data_set 120        23        1
           :           :           :           :

<静的解析結果>
レベル  警告番号  出現頻度  警告メッセージ
A       3341     18       浮動少数点の式が等しい事を(“=”又は“!”で比較すると予期しない結果が発生するので避けてください。
B       3202     3        名札(label)は、この関数の中で使用されていません。
A       3201     2        この文には到達できません。
:       :         :
    
```

図3. プログラムの静的解析結果 メトリクス計測結果と、プログラミング上問題のありそうな内容をチェックした例。

Result of program static analysis

3.3 テスト情報/不具合情報管理機能

テストの予定項目数や実施件数の入力、不具合情報の入力用インタフェースを提供する。また、入力された情報から、重要度別/機能別の不具合検出一覧表などを作成する不具合分析機能をもつ。さらに、当社で開発したソフトウェア信頼度推定ツール⁽⁵⁾を利用するためのデータを生成し、信頼度推定を容易に実現できる仕組みを提供している。不具合情報の登録画面を図4に示す。

このような機能を利用したテスト情報/不具合情報の共

(注5) メトリクスは、プログラムを定量化するときの尺度(ものさし)で、代表的な例としてステップ数がある。

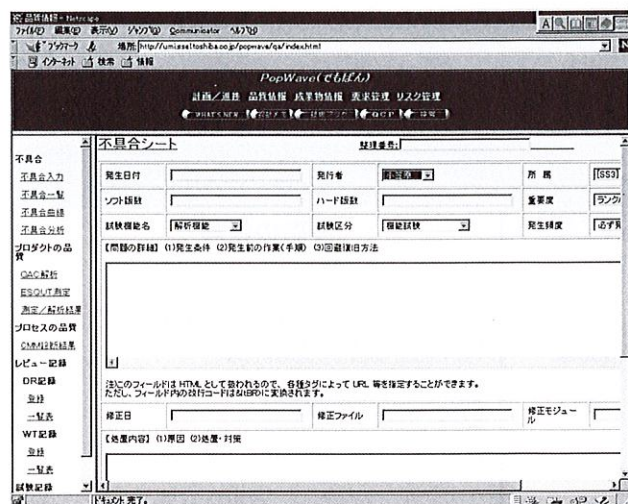


図4. 不具合情報登録画面 不具合現象、重要度ランク、発生頻度など登録する内容をカスタマイズできる。

User interface for defect tracking tool

有により、同じような誤りを複数の開発拠点間で起こすことを防止できる。

3.4 リモート品質評価機能

分散した拠点間では、管理する側がレビューにより品質のチェックを行うことは難しい。レビューの代用として、静的解析/メトリクス計測ツールによる品質評価は、品質の作り込みに有効である^{(6),(7)}。

この機能では、構成管理されているソースプログラムを対象に、プログラムの静的解析を行う。静的解析ツールは言語(例えばC, C++, VB^(注6)など)ごとに用意されている。図2の画面で、解析対象とするファイルをチェックし、静的解析ツールを起動すると、図3に示すような形式で解析結果を出力する。利用者はこの解析結果を参考にして、プログラミング上問題のある部分や、構造的に複雑になりすぎている部分を修正する。

3.5 共有情報登録/参照機能

分散した開発拠点間の開発スケジュールや仕様変更の内容の共有が、目的意識の一致、コミュニケーション不足の解消などのために必要である。また、技術情報や伝達事項を正確に伝えノウハウとして蓄積することも、開発者間での情報の横展開や技術の伝承に必要である。

この機能では、打合せ記録、技術情報、仕様変更記録などの情報を共有する仕組みを提供する。ある特定の送り先にメールを送ると、自動的にデータベースに登録される仕組みを実装した。メールで発信された設計情報を、Web上で参照している例を図5に示す。このように共有すべき情報を容易にアクセスできるため、伝達漏れなどの防止につ

(注6) VB (Visual Basic) は、Microsoft 社の商標。

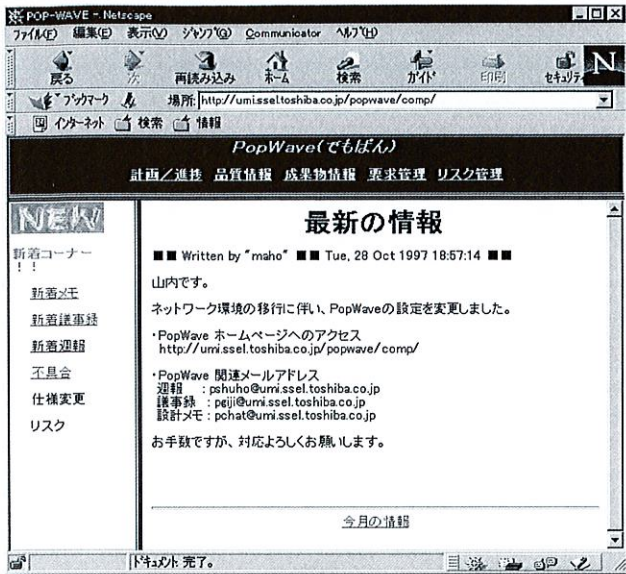


図5. 情報共有の画面 Web上で非定型の情報を参照している例。このような情報の伝達は、分散化した開発環境では不可欠である。

User interface for communicating design information

ながる。また、これらの情報は、データベース化されているため、仕様に対する解釈の相違などに起因する問題も減少すると考えられる。

4 分散開発環境の評価と品質保証エージェントの構想

これまで述べてきた、イントラネットを利用したソフトウェア開発環境を実プロジェクトへ適用中である。構成管理機能など基本的な機能は有効に活用できている。しかし、品質チェック機能などは、利用者がアクションを起こす必要があるため、必ずしも全員が利用しているわけではない。さらに、共有すべき情報なども、自分からアクセスしないと入手できないので、全員が同じ情報を共有していない場合がある。今後、分散開発のために構築した仕組みを有効に活用するためのプロセスの確立と、品質保証ツールなどが自律的に動く仕掛けが必要と考えている。

例えば、PlangentTM(⁸)などのエージェント機能を利用することによって、静的解析によるソースプログラムの品質チェックや、成果物の定量化による進捗管理などの自動化が期待できる。このときに必要なことは、成果物がデータベース化されてきちんと管理されているということである。なぜならば、もし開発した成果物(例えばプログラム)がバックアップのために重複してコピーされていたり、テスト

のために余分なファイルが作られていると、不要なプログラムも解析/計測することになり、ソフトウェア品質管理ツールの処理結果の精度や信頼度が低下する。このような混乱をなくすために、構成管理は必須(す)である。

5 あとがき

イントラネットと Web ブラウザを利用したソフトウェア品質管理環境を構築し、その実現性と有効性の確認ができた。今後、エージェント機能を利用したソフトウェア品質管理の自動化の推進と並行して、要求仕様の管理や、リスク管理などの機能も検討し、プロジェクト全体を管理するための開発環境として整備する予定である。

文献

- (1) 松尾 朗, 他. ソフトウェア分散開発における非同期型コミュニケーション支援. 東芝レビュー, 48, 8, 1993, p.614-616.
- (2) 太田哲夫, 他. ソフトウェア分散開発向けリアルタイム電子会議支援システム. 東芝レビュー, 48, 8, 1993, p.617-619.
- (3) 小林信裕, 他. CASE ツールによるソフトウェア品質管理. 東芝レビュー, 48, 8, 1993, p.620-622.
- (4) 小笠原秀人, 他. 品質評価・管理のための手法とツールの実践的適用. 東芝レビュー, 51, 2, 1996, p.42-45.
- (5) ㈱東芝 研究開発センター S&S 研究所. ソフトウェア信頼度推定ツール. http://www.toshiba.co.jp/tech/software/sqatp/index_j.htm
- (6) 會澤 実, 他. プログラム解析・計測ツールを利用したソフトウェア品質向上支援活動. 日科技連 第17回ソフトウェア生産における品質管理論文集, 1997, p.81-88.
- (7) Ogasawara, H., et al. Experiences of Software Quality Management Using Metrics through the Life-Cycle. Proceedings of the 18th ICSE, March 1996, p.179-188.
- (8) ㈱東芝 研究開発センター S&S 研究所. 考えながら動き回るエージェント. http://www2.toshiba.co.jp/plangent/index_j.htm



小笠原 秀人 OGASAWARA Hideto

研究開発センター S&S 研究所主務。
ソフトウェア品質評価・テスト技術の研究・開発に従事。
Systems & Software Research Labs.



會澤 実 AIZAWA Minoru

研究開発センター S&S 研究所。
ソフトウェア品質評価・構成管理技術の研究・開発に従事。
Systems & Software Research Labs.



山田 淳 YAMADA Atsushi

技術企画部 技術業務プロセス革新センター主務。
ソフトウェア品質評価・プロセス改善技術の研究・開発に従事。
Technology Planning & Coordination Div.