

B-041

## システム開発における作業状況監視方法に関する一提案 The Observation Method of Working Situation in System Development

池田 貴之†  
Takayuki Ikeda

大原 茂之‡  
Shigeyuki Ohara

### 1. はじめに

近年のシステム開発の形態は、一から開発するのではなく既存システム部品の再利用を用いたものが主流となっている。それに伴い、開発者も絶対的な開発量より既存システム部品を再利用することに重点を置いた開発作業を求められる。

ここで、現在のシステム開発において作業状況を把握するために必要なデータは、以下の項目である。

- 新規開発量
- 開発進度
- 再利用率

第一に、新規開発量とは何かについて述べる。現在の日本の主要産業ともいえる組込みシステムの開発現場では、新規開発ソースコードの量は年々減少傾向にある。新規開発ソースコード量は日本および欧州、米国のいずれにおいても 1,000 行から 10,000 行未満という回答の企業が約半数を占める<sup>1)</sup>。つまり、より多くソースコードを書けば良いというわけではないといった現状であるため、今までの新規開発量に対する開発管理では対応しきれていない。そのため、他の要素との連携によってシステム開発の作業はモニタリングされるべきである。

第二に、開発進度とは何かについて述べる。システム開発の現状として開発期間の短期化がある。現在、さまざまな機器の組込みソフトウェア開発においては、6ヶ月未満が全体の約40%、6ヶ月以上1年以上が全体の約50%を占めている<sup>1)</sup>。以上のような大変短い開発サイクルで組込みソフトウェアの開発が行われている。その期間内で開発項目すべてを開発納期までに間に合わせるためには、どの項目が完成していて未完成な項目はどれなのかをすぐに把握できなければならない。

第三に、再利用率とは何かについて述べる。組込みソフトウェア開発企業の半数以上が全ソースコードのうち20%から60%を既存システム部品の再利用が占めているという報告が経済産業省の調査により出ている<sup>1)</sup>。開発期間短縮のために再利用が行われるケースが多いと考えられているが、利点はそれだけではない。既存システム部品は一度デバッグされ検証された上で再利用されるので、バグの混入を極力防ぐことが出来ることになる。それにより、開発しているシステムの信頼性の向上と開発工程での手戻りの減少を見込めるという大きな利点が発生する。そのため、全開発量に対する再利用率が重要な項目と注目されている。しかし、再利用を考慮した開発管理体制が整備されていないのが現状である。

以上の三項目の必要性より、開発量と開発期間のみによる従来のシステム開発管理では、今の開発形態に対応しき

れなくなっている。そのため、Project Manager や管理者が開発者一人一人の作業状況をすべて把握することは困難である。そこで、上述の三項目が自動抽出できれば、現在のシステム開発形態に合った開発作業状況の把握が容易になると考えられる。

そこで本研究では、システム開発において開発者の作業状況の把握を容易にするための作業状況監視方法を提案する。

### 2. 作業状況モニタリング環境

作業状況モニタリング環境とは、開発者に負担をかけることなく作業状況情報の抽出を行う環境である。作業状況モニタリング環境の前提として、開発者、Project Manager、管理者はインターネットブラウザで管理サーバに接続できるものとし、管理サーバ内の作業状況監視システムを仲介してすべての作業を行うものとする。

開発者は、管理サーバにログインし開発用記憶装置にある作業ファイルを獲得し開発を行う。Project Manager は、

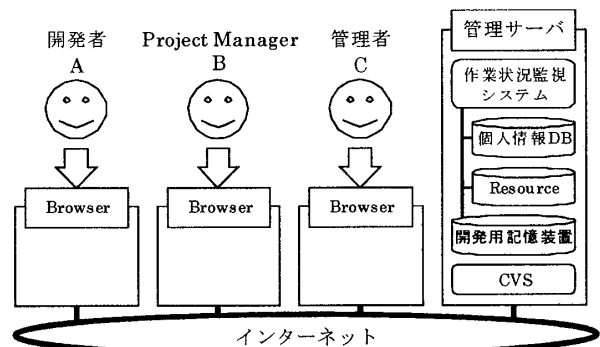


Figure.1 作業状況モニタリング環境の概略図

開発者と同様の開発と開発者の管理を行う。管理者とは、直接システム開発には携わらず各開発者とシステム開発部門の全体を管理する役割をもつ。

Figure.1 に作業状況モニタリング環境の概略図を示す。個人情報データベースは、開発者の作業状況データテーブルがある。作業状況データテーブルには、開発者の開発量や開発進度、再利用率などが格納されている。Resourceには、既存システム部品や仕様書などが収められている。開発用記憶装置には、開発者それぞれに対し作業ディレクトリが与えられ作業ファイルの保存先となる。

### 3. 開発者の作業状況の抽出

本研究では、Linux を用いて抽出法を考える。Linux を採用した経緯としては、開発者が余計な操作をせずに作業状況情報を抽出するために内部ソースコードを参照や変更

† 東海大学大学院工学研究科電子工学専攻

‡ 東海大学電子情報学部情報メディア学科教授

を加えることが可能なオープンソース OS である Linux を採用した。

### 3.1 新規開発量の抽出

提案する開発状況監視システムは、ファイル保存を抽出開始イベントとしていることによって Linux カーネルのファイル管理と連携しなければならない。

Linux カーネルにおけるファイルの書き込み要求は VFS (仮想ファイルシステム) に対して `sys_write()` 関数を用いて行われる。通常ファイルの書き込みを行う `generic_file_write` 関数は、`block_prepare_write` 関数と `generic_commit_write` 関数を利用してファイルを書き込む。

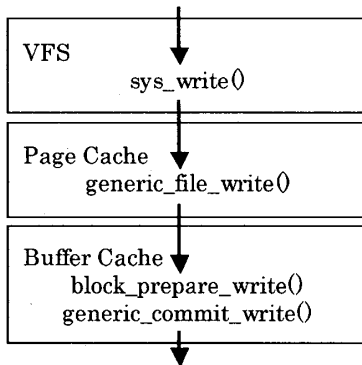


Figure.2 ファイル書き込みの関数参照

Figure.2 にファイル書き込みの関数参照を示す。記憶媒体に対する書き込み処理は実質的に `generic_file_write` 関数が制御している。`generic_file_write` 関数は、呼び出されるとセマフォ `inode->i_sem` を獲得するため、セマフォが開放されない間は一度に複数の書き込み操作が出来なくなる。書き込み終了後にセマフォが開放され、書き込んだバイト数をファイルに返却して終了となる<sup>2)</sup>。

よって、セマフォの開放時を書き込み処理の終了とするため、ファイルの保存処理の一部としてファイル容量の抽出処理を組み込む。抽出の開始イベントは開発用記憶装置内の開発者の作業ディレクトリへのファイル保存とする。ファイル保存が実行されると、作業状況監視システムは対象ファイルの容量計測とファイル差分を抽出し、個人情報 DB 内の開発者のデータテーブルに変更を加える。

### 3.2 開発進捗の抽出

開発計画の際に製作された開発進行予定表を基に、Project Manager が各開発項目に対し完了予定日時を個人情報 DB のデータテーブルに設定しておく。開発進捗の抽出は、開発環境からのログオフを開始イベントとする。

開発者がログオフすると、作業状況監視システムは個人情報 DB 内の開発項目をチェックし完了している項目と完了していない項目を洗い出す。洗い出した結果で完了していない項目に対して、現在の日付と完了予定日時とを比較しあと何日かを算出し、個人情報 DB のデータテーブルの情報を更新する。

### 3.3 既存システム部品の占有率の抽出

開発者は、Resource にある必要とされる既存システム部品を自らのディレクトリへコピーする。そうすることによって作業ディレクトリに既存システム部品を取り込むことが出来る。この既存システム部品の占有率の抽出は、開発環境からのログオフを開始イベントとする。

開発者がログオフすると、作業状況監視システムは開発用記憶装置内の開発者の担当ディレクトリ下にあるファイルに対して、既存システム部品と同じ名前のファイルを検索する。該当したファイルの容量を抽出して合計した再利用容量を開発者の作業ディレクトリ全体の容量で割ることで算出できる。算出結果を個人情報 DB 内の開発者のデータテーブルに変更を加える。

### 3.4 作業データテーブル

Table.1 作業データテーブルの一例

開発者名	東海太郎
所属プロジェクト	プロジェクト A
担当項目数	10 項目
実現比率 (完成項目数/担当項目数)	70%(7/10)
全開発容量	70kB
再利用率 (再利用合計容量/全開発容量)	60%(42/70)

Table.1 に作業データテーブルの一例を示す。作業状況監視システムが抽出してきたデータが格納されている。実現比率項目内にはどの項目が完了していて、未完了の項目については完了予定期間までの日付を提示する。

このデータテーブルより、新規開発量、開発進捗、再利用比率による開発者の作業状況を把握することが出来ると考えられる。

## 4. おわりに

本報告では、開発者の作業状況の抽出方法を提案した。今後の予定として、作業状況監視システムの実装と評価を行う。

## 参考文献

- 1). 経済産業省商務情報政策局  
情報政策ユニット情報処理振興課：  
“組込みソフトウェア産業実態調査報告書”  
公表日：2004年6月21日
- 2). Daniel P. Bovet, Marco Cesati 著、高橋浩和 訳：  
“Understanding the Linux Kernel, 2nd Edition”  
発行所：オライリー・ジャパン  
発行日：2003年6月24日