

クライアント/サーバ型分散システム設計における
評価指針

7N-7

西永誠司 川崎隆二 山本 修一郎

NTT ソフトウェア研究所

1 はじめに

近年、オープンシステム環境下において、情報処理システムのクライアント/サーバ(C/S)型分散システムへの移行、メインフレームからのダウンサイジング、業務や情勢の変化に柔軟に対応できる迅速なシステム開発の実現といった要請が大きくなっている。

GUI構築支援ツールやミドルウェアといったパッケージ/ツール類の技術の進歩は、C/S型分散システム構築の容易化に寄与しているが、C/S型分散システム設計におけるこれらの個々の技術、および、個々の技術を組合せて利用する技術は、試行錯誤的な取り組みが現状であり、以下のような問題が顕著であった。

- 業務の拡張性の確保が困難
 - 業務の拡張に伴い、多種多様なパッケージ/ツール類の知識が必要となる
 - 固定的なSQL分散形態のみであると、業務に変更があった場合の拡張が困難である
- 設計の手戻りによる生産性の低下
 - オープン化による選択肢の多様化に伴い、選択基準が不明確となる
 - システム設計方針を決定する上での明確な判断基準が無い
- 性能面での可用性の確保が困難
 - インタフェースの多様化/複雑化により、性能面での予測が困難である
 - 性能条件による業務/運用等への影響が大きい

本稿では、上記の問題を解決する一方策として、C/S型分散システムの設計手順をモデル化し、設計段階で事前に評価が必要な項目と評価指針を明確化することによって、C/S型分散システム設計作業の効率化を図る方法について提案する。

2 C/S型分散システムの設計手順

C/S型分散システムの開発にあたっては、業務プロセスに対する要求の変化のスピードに可能な限り対応することが要求されるようになってきている。最近では、従来のウォーターフォール型の開発プロセスから、短期間でシステムを稼働させるとともに、その評価に基づいてシステムの部分的な改廃と追加を繰り返しながらシステムを改良していくスパイラル型の開発プロセスが前提となりつつある。

このようなC/S型分散システムの設計手順を、図1のようにモデル化する。

評価は、要求分析/設計の段階で実施し、方式検討に反映する事前評価と、試験工程で実施し、結果を前工程へフィードバックする事後評価とに分類する。事後評価によるフィードバックは、スパイラル型開発プロセスの特徴ではあるが、要求分析の段階まで手戻りが必要になると、作業効率の面での影響が大きい。しかし、このような、要求分析の段階まで手戻りが必要な事項の要因には、要求分析/設計の段階で、適切な事前評価を行なうことで回避可能な項目もある。そこで、事前評価の項目を整理し、方針をガイドライン化することによって、事後評価によるフィードバックの手戻りを抑制し、設計の効率化が期待できる。

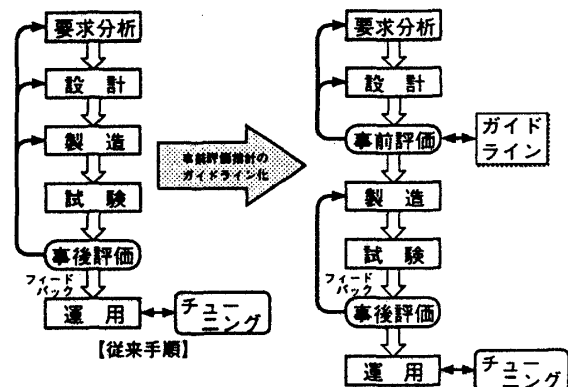


図1: C/S型分散システムの設計手順

Guidelines for designing and developing client-server distributed systems

Seiji Nishinaga, Ryuji Kawasaki and Shuichiro Yamamoto
NTT Software Laboratories

3 C/S型分散システム設計時の評価指針

C/S型分散システム設計における事前評価項目を、表1に示す4種類に分類する。これらの評価項目は、相互に影響を及ぼし合う。(図2参照)これらの事前評価項目に対する評価指針の例を表2に示す。

表1: 事前評価項目

評価対象	評価項目
システムの分散配置	DBの配置 APの配置
プラットフォームの選択	ハードウェア構成の選択 DBMSの選択 ミドルウェアの選択 GUI構築ツール
オペレーション設計	ユーザインタフェース システム運用管理法
ネットワーク設計	ネットワーク構成 ネットワークの運用/管理法 診断技法

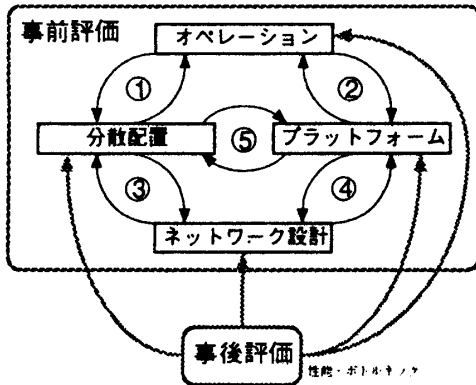


図2: 評価項目の関連

4 適用評価

今回提案した評価指針を実際のC/S型分散システムの開発に適用し、システム全体の検討項目に対する本評価指針の提示範囲のカバー率から、有効性を推定した。その結果、設計の手戻りの防止、性能の事前評価が一部可能になったこと等により、設計工程での稼働を約11%程度、試験工程での稼働削減を含めて、全工程で約9%程度の稼働削減が見込まれている。

5 まとめ

C/S型分散システム開発時の評価指針を明確化し、ガイドライン化することによって開発の効率化が図れることを示した。また、実際のシステムへの適用結果から得られる情報や、設計時のノウハウ的な要素を含めてガイドラインに反映させることにより、さらに効果を上げることが期待できる。

表2: 評価指針

①	<ul style="list-style-type: none"> UIの形式(GUI/CUI)とAP/DBの分散配置方式 → UIの形式に適したAP/DBの配置形態 DB分散形態によるデータ管理機構 → DBの分散に伴い、データの一貫性の保証等の管理機構が複雑化等
②	<ul style="list-style-type: none"> UIの形式とクライアント端末の処理性能 → 高度なGUI等を採用する場合、クライアントマシンに相応の処理能力が要求される UIの形式とミドルウェア/GUI構築ツールの選択 → UI構築にミドルウェアが提供するUI制御機能や汎用GUI構築ツールを利用等
③	<ul style="list-style-type: none"> AP/DBの分散形態とネットワークの信頼性 → 分散化により通信を介する処理が必要となり、ネットワークの信頼性の考慮が必要 AP/DBの分散形態と通信パフォーマンス/コスト → 分散の形態による、通信のパフォーマンス/コストへの影響の検討が必要等
④	<ul style="list-style-type: none"> ミドルウェア/DBMSとネットワークプロトコルの接続性 → ミドルウェア/DBMSのサポートするプロトコルとネットワークとの接続性 ハードウェアのネットワーク構成 → ハードウェアのネットワークとの接続性等
⑤	<ul style="list-style-type: none"> マシン性能によるAP配置への影響 → APを配置するマシンの処理能力と配置形態 ミドルウェアとAP/DBとの接続性 → ミドルウェアで用意されたインタフェースと分散したAP/DBとの接続性等

参考文献

- 1) Morris Sloman and Jeff Kramer (齊藤忠夫監訳): 分散システムと計算機ネットワーク, 丸善, 1988
- 2) NIKKEI OPEN SYSTEMS: C/Sシステム構築法, NIKKEI OPEN SYSTEMS, 1994年4月 (No.13), pp.108-136, 1994
- 3) 長岡満夫: エンドユーザコンピューティング環境, NTT R&D, Vol.42, No.12, pp.1451-1460, 1993
- 4) Yennun Huang and Satish K. Tripathi: Resource Allocation for Primary-Site Fault-Tolerant Systems, IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, Vol.19, No.2, pp.108-119, 1993
- 5) 菅茂: ダウンサイジング, 日経BP社, 1992
- 6) H.Hellerman and T.F.Conroy (大野豊訳): オペレーティング・システムの性能評価, 日本コンピュータ協会, 1980