

システム動作設計支援技法SSDESIGNの研究

—— 性能仕様プロトタイピングの考察 ——

4M-3

吉田 ゆき 池本 浩幸 楠井 洋一 津田 淳一郎
(株) 東芝 システム・ソフトウェア技術研究所

1. はじめに

近年、システムが大規模化すると共に、システムに対する要求も多様化・複雑化する傾向にある。しかし、要求仕様があいまいのまま開発が進み、結果として下流工程からの戻り作業が発生する場合がある。これは、要求定義の段階だけではユーザ要求を正確に把握することが困難になっているためである。

我々は、ソフトウェアの生産の工業化を行なうIMAP(Integrated software Management and Production support system)の一環として、システム設計段階で要求仕様を明確にし、システム関与者間の仕様認識を統一することを目的とした、システム動作設計支援技法SSDESIGN(System Sequence Design)の開発を行なっている。[1][2] これまで、システム仕様を図式言語であるSSDダイアグラムとして記述するエディタ、並びに、図式言語を解釈実行しイメージ化した仕様を、マウスで操作してその応答を検証する対話機能を持つビジュアルプロトタイピングの開発を行なってきた。[3]

しかし、ユーザがあらかじめシステムの実現イメージを体験し、システム仕様を理解・評価するには、機能面の検証だけでは不十分であり、性能を含めた総合的なプロトタイピングが必要となる。そこで、性能面のプロトタイピングのひとつとして、特にユーザにとって重要な処理部分について時間の概念を導入するための問題点の考察、および実現方式について本稿で述べる。

2. システム設計仕様の検証

我々は、SSDESIGNの開発において、機能面に主眼を置いたプロトタイピングの開発を行なってきた。

図式言語であるSSDダイアグラムの横軸はセクションと呼ばれるシステムの機能分担、縦軸は時間の推移を表わしている。SSDエディタにて、システムの機能を表わすボックス(機能ボックス)間を処理の順番を表わす線(アロー)をつないでいき、フローチャートのシステム動作仕様を記述することにより、次工程への正確な仕様伝達を行なう。ビジュアルプロトタイピングは、図1に示すように、ダイアグラムの処理の流れをトレースしながら、システムのイメージを表示するシミュレーションを行なうことにより、システムの動作仕様を確認し、設計仕様を最終的なユーザニーズを満たすものに近づけることができる。

一般に、機能面の要求仕様がほぼ反映された後、それをどのように実現していくかが問題となる。このとき、機能が時間的にどれくらいユーザの思惑どおりに動くか、すなわち性能面の仕様が重要となる。性能は機能を実現するときの制約として関わっているため、システムの仕様決め段階で、機能・性能の両方のユーザ要求を同時に満足できない場合がある。そこで、性能面を考慮したプロトタイピングによって、要求された機能と性能のトレードオフ関係を把握し、最適な実現手段を選択する必要がある。

3. 時間の概念に着目した性能検証

時間の概念を考慮したプロトタイピングを行なうにあたり、性能検証項目についての考察を述べる。

ユーザが性能面の仕様において重視する時間項目は、以下のものが挙げられる(これを目的レベルと呼ぶ)。

項目①: ジョブ遂行時間(トータル時間)

項目②: 操作系の応答時間

項目③: 非操作系の処理時間

①は、一連の動作の集りよりなるまとまった業務を見積るものであり、これにより例えば1日当りの処理件数などが見積られる。この場合、全処理項目の動作時間を決める必要がある。

②は、オペレータが対話処理を行なう時の1操作毎の応答時間であり、処理内容によってはある程度実績から推測できる場合や、逆にユーザからの要求値などにより具体的に設定できることも多い。

③は、化学プラントにおける反応処理、事務処理における伝票作成など、特に対話処理を意識しないひとかたまりの処理をいう。(対話処理、操作が含まれていてもよい)。

プロトタイピングとしては、②の対話処理時間がユーザにとって興味深い時間感覚として意識されるのに対し、①および③は、一般に、実際の処理が長時間にわたることもあるので、単にシミュレーションの経過時間とするのは効率上好ましくない。このため、ユーザにとって特に有効と思われる時間以外は一定時間でシミュレーションを行なうことにより、要求確認のプロトタイピングの効率を重視する。

また、システムの設計段階で処理時間を設定する際、以下の3つの時間が考えられる(これを制約レベルと呼ぶ)。

①ユーザ要求レベル

②実績レベル

③想定レベル

性能検証項目において、これら制約レベルの異なる時間が混在している。このように、設定時間にレベルの差が存在するため、時間の概念を考慮したプロトタイピングが意味を持たなくなることが考えられる。従って、プロトタイピングを行なう際に時間データのレベルの違いを明示する必要がある。

また、要求仕様を明確にするプロトタイピングという観点からいえば、ユーザ要求レベルに重点をおくべきであるので、通常①②③の順に制約レベルの優先度を定める。しかし、設計の観点からみると、②③と①との差を明示する必要がある。

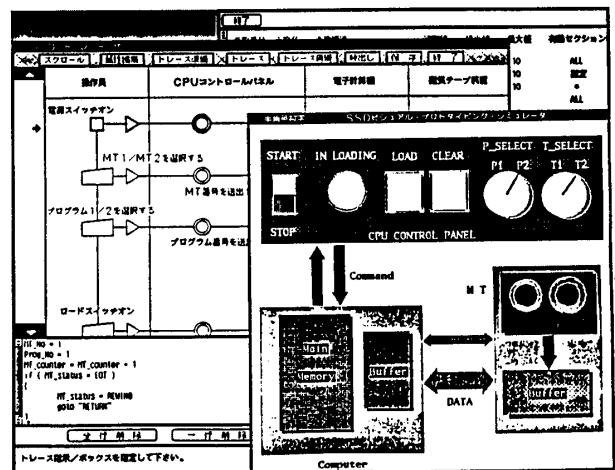


図1 SSDビジュアルプロトタイピング

4. S S DESIGN における性能検証

以上の考察を踏まえ、今回は第1ステップとして、SSDダイアグラムの機能ボックスに処理時間の属性を付加し、ユーザ要求に対するマンマシンインタフェースの仕様確認を中心とした性能検証を行なった。このとき、ダイアグラムに関して以下の制約をつけることにした。

- ダイアグラムの横軸方向のタイミングは、制約ではなく要求である場合が多い。従って、要求が変わらないかぎり、そこで記述したダイアグラムと設定した時間との違いを明示する。
- SSDダイアグラムの記述規則として、図2に示すように、ダイアグラムの途中で開始または終了するボックスは、最初に処理を開始するボックス（開始点ボックス）と最後に処理を終了するボックス（終了点ボックス）につながる経路中のボックスと開始時刻が同期していなければならない。

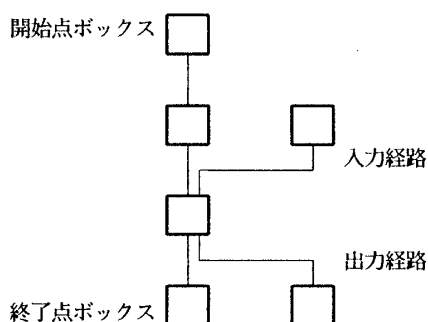


図2 開始点ボックスと終了点ボックス

次に、SSD性能プロトタイピングの機能について述べる。

4.1 設定時間属性機能

ダイアグラムの各機能ボックスに処理時間を設定する属性機能を付加した。設定時間は、ある機能ボックスが処理を実行し次の機能ボックスに到達するまでの所要時間である。記述するボックスは全てではなく、ダイアグラム上の任意の範囲を指定することができる。また、指定していない場合は、前後の接続から判断した値をデフォルトにする。

設定時間は、3つの制約レベル（要求、実績、想定）を時間テーブルを埋める方式で値を設定する。

さらに、この値をリアルタイムで行なうか（Rモード）、仮想時間で行なうか（Vモード）を選択することにより、リアルタイムの範囲を限定できる。対話入力を表わす機能ボックス（□）は、目的レベル②の操作系を意味する特殊なボックスで、操作が終わってから次の操作が可能となるまでの時間を記述する。このとき、デフォルトはRモードとなっている。

4.2 設定時間による性能仕様検証機能

従来は一定時間間隔でシステムイメージを表示する方式をとっていたが、今回は、4.1の機能で記述した設定時間を反映し、以下の方式のプロトタイピングを行なった。

まず、ダイアグラム上に記述された時間軸方向の処理開始時間を合わせ、終了時間をダイアグラムの横軸に並んだボックスの中で一番処理時間が遅いものに合わせる。このとき、設定された時間に従い、最も遅いボックスと整合性をとるための遅延時間を表示するボックスを生成する機能によって、各セクション間における制約レベルの余り時間を明確にする（図3）。また、ダイアグラムの横軸で同期しなければならないとき、まだ実行中の機能ボックスと待機中のものとを色分けして明示することにより、負荷状態を一覧できる。

さらに、シミュレーションの経過時間と共に、任意の処理区間での性能仕様の確認をするために、ある時間のタイミングからの経過時間も表示できるようにする。また、R、Vモードの時間の単位を変えることにより、時間のかかるシミュレーションを短縮して実行できる。

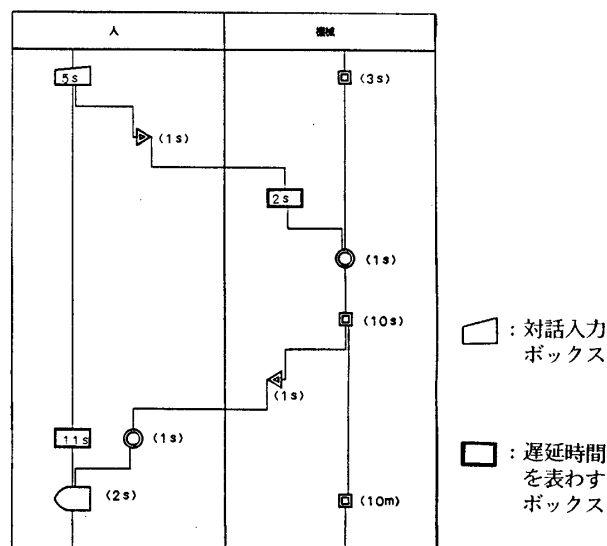


図3 SSDダイアグラムにおける時間の整合性

5. 効果

従来は、機能面での動きや操作に限られた単に画面との対話ができるというプロトタイピングであった。今回、応答時間の検証機能を付加したことにより、より実現イメージに近い操作感覚を体験でき、精度の高い仕様確認が可能となった。また、実行状態と待ち状態が明示されるようになった結果、ムダあるいは負荷の多いセクションのバランスを調整し設計仕様を修正することで、現実の制約を十分考慮したシステム設計を効率よく行なうことができる。

6. 終わりに

このように、今回機能面に加え、性能面のプロトタイピングを実現したことにより、マンマシンのレスポンスでのユーザ要求とのズレを発見でき、仕様の検証に対する支援が強化された。しかしながら、現在の性能面のプロトタイピングは、処理時間に対する要求分析を主目的にしており、設計のための検証・見積りなどについては設定時間の精度の点でまだまだ不十分である。従って、いかに時間要求を設計仕様を持ち込むかという問題に対する考察を深めると共に、今後は、実績データの収集およびデータ量を考慮した性能プロトタイピングにより、きめの細かいシステム仕様の検証を行なうことが課題である。

参考文献

- [1]高柳他 「システム動作設計支援ツール SSDシミュレータについて」 第31回情報処理全国大会
- [2]楠井他 「システム動作設計支援技法 S S DESIGN の開発 (1)エディタの評価」 第35回情報処理全国大会
- [3]池本他 「システム動作設計支援技法 S S DESIGN の開発 (2)ビジュアルプロトタイピングシミュレータ」 第35回情報処理全国大会