

再利用技術を用いたグラフ帳票ジェネレータによるプロトタイピング

3 J-5

北澤 昇、新道 美喜男、品川 麻由美
日本電気㈱ 生産システム開発本部

1. はじめに

工場の生産現場における不良解析、生産管理等の業務を支援するシステムに求められる帳票は様々なものがある。とりわけ、グラフ帳票は開発期間が長く、ユーザ要求に迅速に応えられない、開発工数が多くなる、という傾向が強い。

今回、生産現場の各種実績データをグラフ帳票として出力するグラフ帳票ジェネレータを再利用技術(ビルディングブロック方式)を用いて開発し、これを用いたプロトタイピング手法により、ユーザ向けシステムを構築することで成果を上げたので、ここに報告する。

2. グラフ帳票ジェネレータ

本ジェネレータは、図1に示す3つのサブシステムにより構成される。QDBは工場ホストが収集した各種実績データを格納するデータベースであり、QALは、このデータに対して、統計演算、並べ替えなどのデータ加工を行

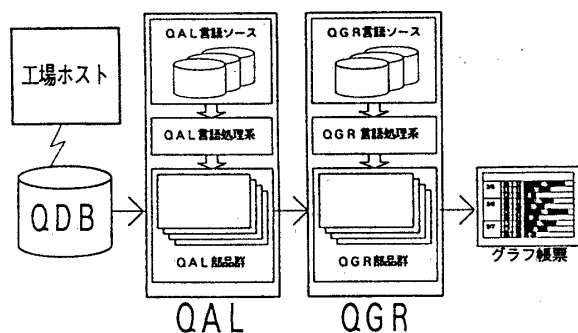


図1 グラフ帳票ジェネレータ

い、QGRは、加工されたデータを図2に示す様なグラフとして出力する。

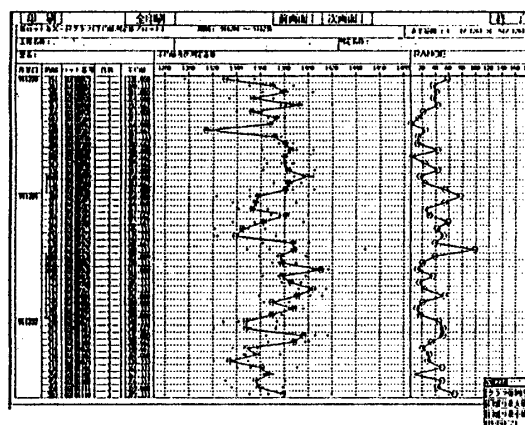


図2 グラフ例

ビルディングブロック方式にてデータ加工処理、グラフ出力処理を構成するため、我々の過去のシステム開発の経験をもとに、それぞれの処理を構成する単位機能を洗いだし、QAL部品として、データベース検索用、データ四則演算用、データソーティング用等の12部品、QGR部品として、タイトル、X軸、Y軸、折線プロット等の14部品を開発するとともに、各部品群の組合せを定義する簡易言語としてQAL言語、QGR言語を設計し、この処理系を開発した。

3. ユーザへのシステム導入

システムの開発・導入を以下の3ステップに分けた。

第1ステップ：プロトタイピング準備作業

- (1)基本システム構成の1-サ先へのレ・ユー
- (2)第1次リリース機能開発時の為のSAを実施した上で第1次リリースに向けた開発を行った。次ステップのプロトタイピングの実施を前提としているので(2)のSAは最低限の工数で行う。

第2ステップ：プロトタイピング実施

初期リリースから、6カ月後の第9次リリースまで、8サイクルのプロトタイピングを実施した。図3にこの期間中に発生した新規ニーズ数を示す。なお、マイナス値は不用なグラフとしてシステムから削除した帳票である。

ユーザニーズの把握は、

(1)リリース毎にイントルーザと共催するシステム運用会議におけるイントルー

(2)ユーザが随時作成するシステム機能・グラフ追加要求書

により行った。

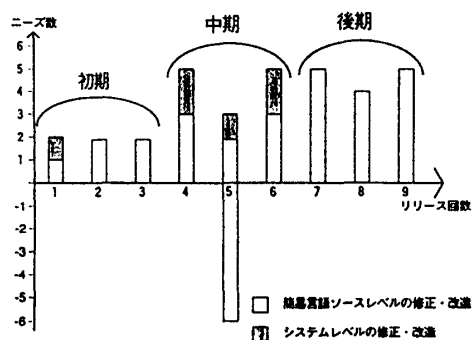


図3 ニーズの推移

ニーズ内容を分析した結果、プロトタイピング期間の初、中、後期で以下のような特徴を挙げることができる。

【初期】仕様の漏れが視覚的にすぐに判断できる帳票フォーマットなどに対する改造要求が主。

【中期】システムレベルの機能追加が発生。グラフ追加要望が増加。

【後期】システムの制約を受けない簡易言語部分の修正要求が主。件数は中期よりさらに増加。

第3ステップ：開発・保守の移管

(1)グラフの開発ニーズが定常的に発生

(2)システムレベルの改造要求が収束

したことから、ユーザ側に簡易言語を作成編集する為のGUIベースの専用エディタ（QALエディタ、図4）を提供することで、以下の様に開発・保

守について分業体制を敷いた。

【ユーザ部門】簡易言語レベルのグラフ帳票

【開発部門】部品追加等のシステムレベル

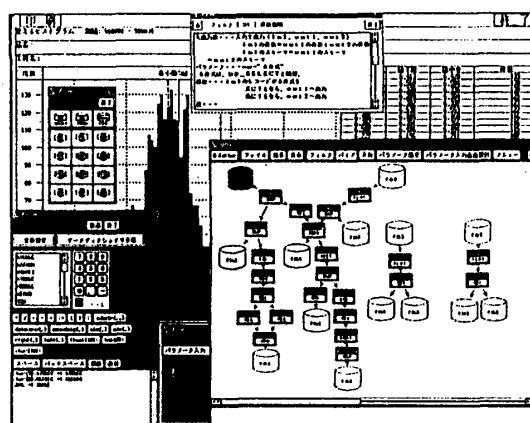


図4 QALエディタ

4. まとめ

(1)過去のシステム開発の経験を生かすことでプロトタイピング実施前の部品の洗い出し、機能設計を適切に行うことができれば、プロトタイピング実施中に発生する新規ニーズの多くは簡易言語レベルの記述変更だけで済み、新部品の追加等のシステムレベルの改造は少量で済む。これによりニーズヒアリングのサイクルを短くすることが可能になり、よりユーザニーズに密着した質の高いシステムが開発できる。

(2)プロトタイピングの過程で、システムレベルの改造要求が収束すれば、ユーザ部門へ開発・保守を移管することでユーザ業務に密着したさらにより良いシステムに成長させることができる。

但し、ユーザがシステムのベース部分に関わる基本的な限界を認識し、これを逸脱するシステムレベルの潜在的改造ニーズを自己規制している可能性もあるので、プロトタイピング中のニーズヒアリングをそのシステムの開発プロセスとしてのみ位置づけるのではなく、次期システムのSA活動として位置づけて取り組む必要がある。