

CENTUM CS バッチ JOB のエンジニアリング手法

(テンプレートを使用したエンジニアリング)

Engineering Method of Batch Process using for CENTUM CS.
(As for Engineering using The Application Template Package.)

佐野 直樹^{*1} 木畑 仁^{*1} 浜口 延正^{*1}
SANO Naoki KIBATA Hitoshi HAMAGUCHI Nobumasa

CENTUM CS で動作する CS Batch パッケージを使用した、重合プラント向けのアプリケーション用テンプレートについて紹介する。本テンプレートは、当部署にて長年に渡り蓄積されたバッチプロセスに対するノウハウを元に、一般的な重合プロセスにおける制御機能を実現したソフトウェア、及びJOBドキュメント(機能仕様書等)を提供するものである。本テンプレートを使用することにより、ユーザでのエンジニアリングを容易に行うことが可能である。

We introduce the application template package for polymerization plant running on CENTUM CS batch system. This application template package provides software for typical polymerization plant including control specifications, and documentations such as DCS function specifications, etc., which have been designed and configured based on the Engineering know-how of batch process. DCS users can easily start engineering for a batch process by using these templates.

1. はじめに

一般的にバッチプロセスに対するエンジニアリングは、DCSを使用して自動化する場合、プロセス及びプラントの性質上、いわゆる連続制御系のプロセスと比べエンジニアリング工数が多大となりがちである。

その主な理由としては、

- ・ 製造する製品が多品種変量であり、数多くのシーケンス(SEQ)を作成し管理する必要がある。
- ・ 現場でのオペレータ介入(原料の手投入などの手作業)、が通常の運転として行なわれる為、SEQとしての考慮が必要となる。
- ・ 中断/再開処理等の異常処理がユーザ及びプラントにより異なりSEQとしての考慮が必要となる。
- ・ 同一のユーザにおいても、工場が変わると運転方法、及び運転形態が異なる場合が多い。そのため標準化が行いにくく、その都度運転方法を検討し、プラントに合わせた仕様決めを実施しなければならない。

等があげられる。

当社では、DCSを発売以来長年にわたり、数多くのバッチプロセスのエンジニアリングを行ってきた。しかし、これまでも上記のエンジニアリング工数が多大と

なりがちな要因が常につきまとい、短時間で効率良いエンジニアリングを行うためにはどうしたら良いか、我々ベンダーのエンジニア、及びユーザの担当エンジニアの頭を悩ませてきた。

そこで、今回バッチプロセスのエンジニアリング効率を改善する為に開発したテンプレートについて紹介する。

2. バッチアプリケーションの標準化

現在、バッチシステムの標準化のために用意されている国際規格に、ISA-SP88(1)がある。

当社でもこのISA-SP88.01に準拠した、バッチプロセス用パッケージであるCS Batchパッケージ(2)を1994年より発売している。今回紹介する重合プラント向けのアプリケーションテンプレートもこのISA-SP88.01に準拠したCENTUM-CS用のCS Batchパッケージを使用して作成している。

3. テンプレート開発の背景

実際にバッチプロセスのアプリケーションを設計する際には、以下に示すエンジニアリングのポイントがあげられる。

*1 システム事業部エンジニアアプリケーション技術部

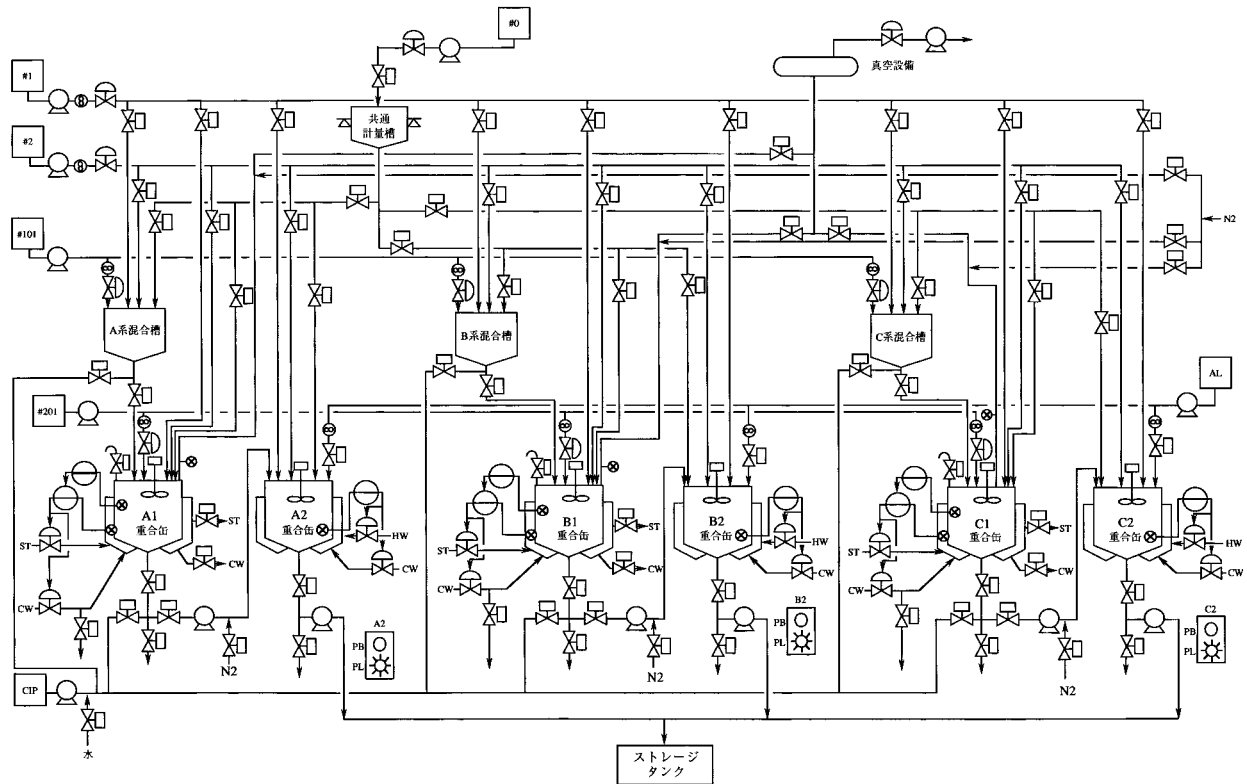


図1 重合テンプレート用に作成したプロセス

- ・ 自動化の範囲。どのような機能を実現し、DCSにて制御を行うか？
- ・ オペレーションはどの単位で行うか？また、どのように行うのか？
- ・ 状態 (STATUS), モード (MODE), 工程 (PHASE) をどのように定義し切り分けを行うか？
- ・ SEQの中断/再開等の異常処理をどこまで自動化し、どのように実現するか？
- ・ 処方 をどのように定義するか (処方の管理方法, データの持ち方, 工程の切り分け方, 等)？
- ・ 将来的な改造, 及び増設に対してどのような考慮を行うか？

バッチプロセスのエンジニアリングが成功するか否かは、上記項目を総合的に検討し、整合性をとり、機能を満足できるかどうかにかかっている。しかしながら、これらを全く白紙の状態から短時間で全て決定することは容易なことではない。また、実際にアプリケーションとしてどのように実現するのも考慮すると、短時間でかつ効率良く作業を行うことはとても難しい。テンプレートの開発目的は、ユーザにてこれら数多くの検討項目を効率良く決定し、なおかつ短時間でアプリケーションを完成させることにある。このアプリケーションのテンプレートを使用することにより、

1. 機能仕様決めにかかる時間の削減。
 2. 機能仕様の標準化。
 3. アプリケーション設計 / 製作 / 検査時間の削減。
 4. アプリケーション品質の向上。
- をはかることが可能である。

4. 重合テンプレートの詳細

一口にアプリケーションのテンプレートと言っても、対象となるプロセス、及び業種により機能が異なり多種多様なものが考えられる。今回本稿で紹介するのは、重合プロセス向けに開発したテンプレートである。

4.1 テンプレート用プロセスの概要

重合テンプレート用に作成したプロセスを図1に示す。

4.2 プロセス概要

- ・ 本プラントはA系列, B系列, C系列の3つの系列から構成される重合プラントである。
- ・ 各系列とも1つの混合槽と2つの重合缶が直列に接続された構成の、多段重合を行う。
- ・ これら3つの系列の機能は全て同一である。
- ・ 最終重合缶 (A2, B2, C2) からは、ストレージタンクへ製品を移送する。

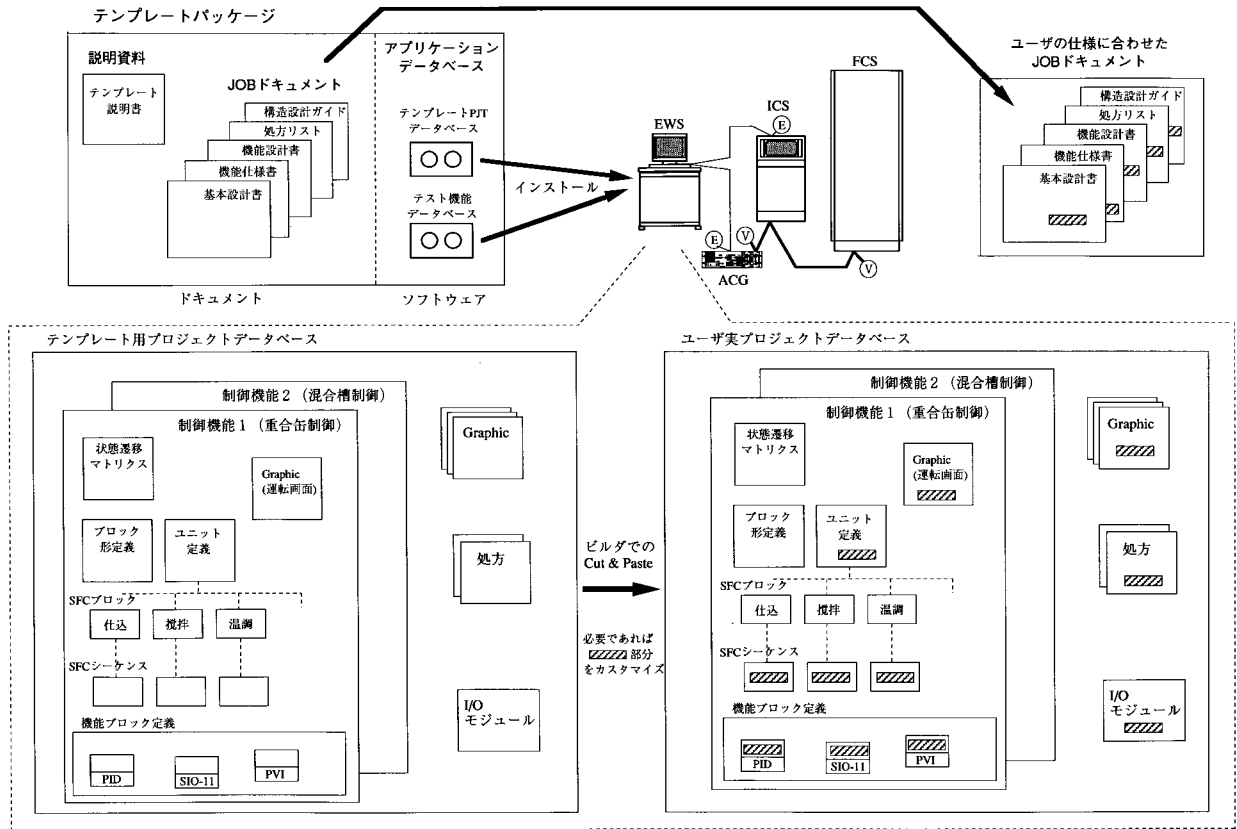


図2 テンプレートを利用したエンジニアリングイメージ

- A1, B1, C1重合缶では、内温制御、外温制御、内-外温制御を行う。
- A2, B2, C2重合缶では内温制御のみ行う。
- A1, B1, C1重合缶では真空設備による真空引き、及び窒素加圧を行う。
- A1, B1, C1重合缶では、各系列専用の混合槽より仕込を行う。
- A2, B2, C2重合缶では現場での手作業の指示確認の為、ランプと押し釦が存在する。
- 全重合缶は缶上よりCIPを行う。
- 共通計量槽は#0原料を各重合缶及び系列混合槽へのロードセルによる減算計量を行う。

4.3 制御機能一覧

- 重合缶制御
- 混合槽制御
- 共通計量槽制御
- 原料仕込設備制御
- 真空設備制御
- CIP設備制御

4.4 テンプレートとして提供するドキュメント

重合テンプレートは、実際のJOBと同様に以下の仕様書を提供する。

- DCSシステム基本設計書
- DCSシステム機能仕様書
- DCSシステム機能設計書
- 処方リスト
- 構造設計ガイド

また、テンプレート本体の説明書として以下の物も提供する。

- CS Batch重合テンプレート説明書(エンジニアリング解説編)
- CS Batch重合テンプレート説明書(動作環境説明編)

4.5 テンプレートとして提供するソフトウェア

- テンプレートとして作成したアプリケーション一式
- テスト機能用データベース一式

4.6 動作環境

実機にて動作させる場合：

- EWS × 1台, ICS × 2台(1台で可),
- FCS × 1台, ACG × 1台

FCSシミュレータにて動作させる場合：

EWS×1台

5. エンジニアリング方法

今回提供するテンプレートは、1つのJOBとして完全な形で作成を行っている。以下に示す手順でテンプレートを使用することにより、効率良いエンジニアリングを行うことが可能である。

5.1 CS Batchのエンジニアリングを行う際の勉強資料として使用

実際にエンジニアリングを行う前に、CS Batchとはどのようなものであるかを理解する為にソフトをマシンにインストールし、動作を確認する(CS Batchの動作、ICSでの操作等の学習)。

これからエンジニアリングを行うにあたり、JOBドキュメントをもとにして、仕様のまとめ方、事前に準備する資料が何であるかの参考とする。

5.2 テンプレートより使える部分を検討し、仕様を決定

テンプレートは、当社にて想定した重合プラントに対して実際にエンジニアリングを行ったものであるため、ユーザの実プラントと全く同じ仕様ではない。従ってユーザでの仕様を決める際以下の手順に従い、テンプレートを最大限活用する。手順としては、

- 1)仕様をまとめる際、可能な限り本テンプレートの思想を取り入れ仕様をまとめる。(ユニット分け、オペレーション方法等)
- 2)テンプレートで提供されている機能で、そのまま使える機能はそのまま使用する。
- 3)どうしても、テンプレートには合わない機能、もしくは存在しない機能があれば、テンプレートの中で1番良く似通った機能をもとにカスタマイズする。

5.3 必要なソフトウェアのコピー

実際にEWS上で、テンプレートを実PJTにコピーするには、標準のビルダを使用しコピーを行う。

手順としては、

- 1)EWSにテンプレートをインストールする。
- 2)ユーザがこれからエンジニアリングするPJTを登録する。
- 3)両方のプロジェクトを、ビルダで開き、必要な部分をCut & Pasteしユーザのプロジェクトを構築する。

6. エンジニアリングイメージ

図2にテンプレートを使用した、実際のエンジニアリングイメージを示す。テンプレートパッケージとして提供されたJOBドキュメントを、ユーザでの仕様に合わせ必要部分の機能修正、及び機能追加を行う。仕様が決定したらソフトウェアをEWSにインストールし、あらかじめ定義しておいた、ユーザ実プロジェクトに対し、ビルダを使用しCut & Pasteを行う。カスタマイズが必要な部分(テンプレートとの差分)を作成し、動作確認を行う。

7. まとめ

以上紹介したように、テンプレートを使用したエンジニアリング手法を導入することにより、従来のエンジニアリングと比べ、短時間で効率良くエンジニアリングを行うことが可能である。また、初めてCS Batchシステムをエンジニアリングされるユーザにおいては、システムの理解、及びエンジニアリング手法をスムーズに習得することが可能である。

なお、現在重合テンプレートに続き、他のプロセス(空送系、等)についても、テンプレートの作成を計画中である。

参考文献

- (1)坂本英幸.“パッチシステムの国際標準化「SP88」——その背景と概要”.計装. Vol. 38 No. 10 pp. 61~63(1995)
- (2)元吉伸一.“ISA SP88準拠のパッチ管理パッケージ”.計装. Vol. 38 No. 10 pp. 39~43(1995)
- (3)坂本英幸,加納俊之.“SP88とCS Batch”.横河技報. Vol. 39 No. 1 pp. 3~6(1995)