

EPICS、QT をベースにした制御ソフト開発

DEVELOPMENT OF THE CONTROL SYSTEM BASED ON EPICS AND QT

塚田義則^{#A)}, 照沼信浩^{B)}

Yoshinori Tsukada ^{#A)}, Nobuhiro Terunuma ^{B)}

^{A)} Kanto Information Service (KIS)

^{B)} High Energy Accelerator Research Organization (KEK)

Abstract

In the recent ATF control system, EPICS is widely applied to build the control software especially for ATF2 international program. The Qt based GUI which easily communicate with EPICS has advantages on the easy programming and quick preparation for the running requests by ATF R&D programs. It is realized by tools, Qt creator, epicsqt and also C++. The method applied in the ATF control system will be reported.

1. はじめに

先端加速器試験装置(ATF)を立ち上げた 90 年代中頃から、制御システムには Vsystem^[1]を使用してきた。近年の国際化に伴って PC ベースの EPICS^[2]が十分に使えるようになってきたことから ATF でも EPICS での開発が進み、Vsystem と共存するようになった。最近では EPICS のみで開発を進めている。

EPICS を使用しているいくつかのところでは、GUI を EPICS と連携させるツールとして Java を使用している例が多く見受けられるが、ATF では以前から、ソフト開発において C、C++ 言語でのプログラミング開発を行っており、数多くの過去の資産がある。それらの資産を引き継いで、いかに効率良く開発が進められるかを検討した結果、GUI を開発するツールとして Qt^[3]を使用するのが最適であると判断し、ソフト開発を進めている。

本稿では、現在 ATF の実験で使用されている EPICS と Qt をベースに開発を行った制御ソフトを例にあげ報告する。

2. 開発について

2.1 開発環境

ATF で使用されている制御 PC には Scientific Linux 6.5 の OS がインストールされている。下記にあげたアプリケーションソフトウェアをそれぞれの制御 PC にインストールして ATF の制御ソフトウェア開発を進めている。

2.2 Qt について

Qt は C++ で開発されたアプリケーション開発用のフレームワークである。マルチプラットフォームで Linux、Mac OS X、Windows に対応している。Qt にはソフト開発を補助する統合開発環境 QtCreator が備わっている。QtCreator には GUI 作成に役立つ強力なツールである QtDesigner があり、ユーザーは簡単に

ボタンやチェックボックス、テキストなどを任意の場所に配置することができ、タブ機能を行使用することで GUI 画面をコンパクトに、思い通りにレイアウト編成ができるため、効率の良い開発が行える。

また、Qt にはシグナル・スロットと呼ばれる機能があり、これはオブジェクト間の通信を提供するもので、ユーザーのアクションをもとに動作させることができる。例えばボタンをクリックされた場合、アプリケーションはあるコードを実行する。QtCreator はこのコードのテンプレートを容易に作成するので、作業の負担を軽減できる。バージョン管理ツールの git などにも対応しており、複数人での開発がスムーズに行える。

他にもいくつか統合開発環境は存在しているが、例えば Eclipse の場合、Qt との連係が QtCreator ほど無い。CSS は現段階では調査不足のため、開発に利用するにも準備作業などで負担が大きく ATF での開発においては QtCreator を選択し作業を進めることとなった。

実験の際、解析プロットの表示用として利用しているのが ROOT^[4]である。Qt から ROOT のグラフ表示をするうえで必要となるのが ROOT のパッケージの QtROOT である。執筆時点での Qt の最新バージョンは 5.3 であるが、QtROOT が未対応のため、ATF においては Qt4.8 バージョンを利用している。

Table 1: Application and Version

アプリケーション名	バージョン
Qt	4.8.6
EPICS	3.14.12.4 (base)
epicsqt	2.8.1
ROOT	5.34.14
QtCreator	3.1.2

[#] ytsukada@post.kek.jp

2.3 EPICS と Qt を連携できるツール

Qt と EPICS を連携させる上で非常に役立つツールとなるのが、epicsqt^[5]である。

epicsqt とは C++を使用して、グラフィカル Qt フレームワークに EPICS CA 機能の拡張 API を組み合わせたものである。

統合開発環境 QtCreator に対応するプラグインがあるので、ユーザーは EPICS の PV データを GUI 作成段階から使用できる。

EPICS を扱った GUI 開発において、多く使われている EDM、MEDM ツールは X11 ベースでやや古く、EPICS のホームページでも Extensions の項目に epicsqt が掲載されているくらい認知度が広がっている。

3. 制御ソフト開発の実用例

3.1 プログラムの概要

今回 EPICS と Qt を使用して例に挙げる ATF2 IP Beam Size Monitor 制御ソフトは、いくつかの用途によって測定する画面がタブにより使い分けできるようになっている。(Figure 1)

GUI から命令を実行させ、EPICS を経由してアクチュエータを制御させたり、自動ステージを制御さ

せたりする。そのデバイスの動作後の測定結果は ADC モジュールからデータを取得する収集プロセスが EPICS 上にあげ、GUI がプロットし解析するソフトである。

3.2 データ表示

デバイスの現在のポジションを表示するアナログ値、ステータスを文字列として表示している箇所は、epicsqt が用意している QELabel とよばれるラベルウィジェットを使用している。

epicsqt を利用する前までは、Qt プログラミングによる非同期スレッドを用いて EPICS レコードにアクセスし、取得してきた値をラベル表示として GUI に書き込む処理を行っていた。

しかし、このラベルウィジェットを使用することによって、QtCreator で編集できるプロパティに必要な情報を設定するだけで、プログラミングコードを作成しなくても、ウィジェット内で EPICS レコードデータをモニターしてくれているので、GUI では簡単にデータ更新が行える。

3.3 Scan パラメータ値入力

Scan で必要なパラメータの設定値は epicsqt で用意

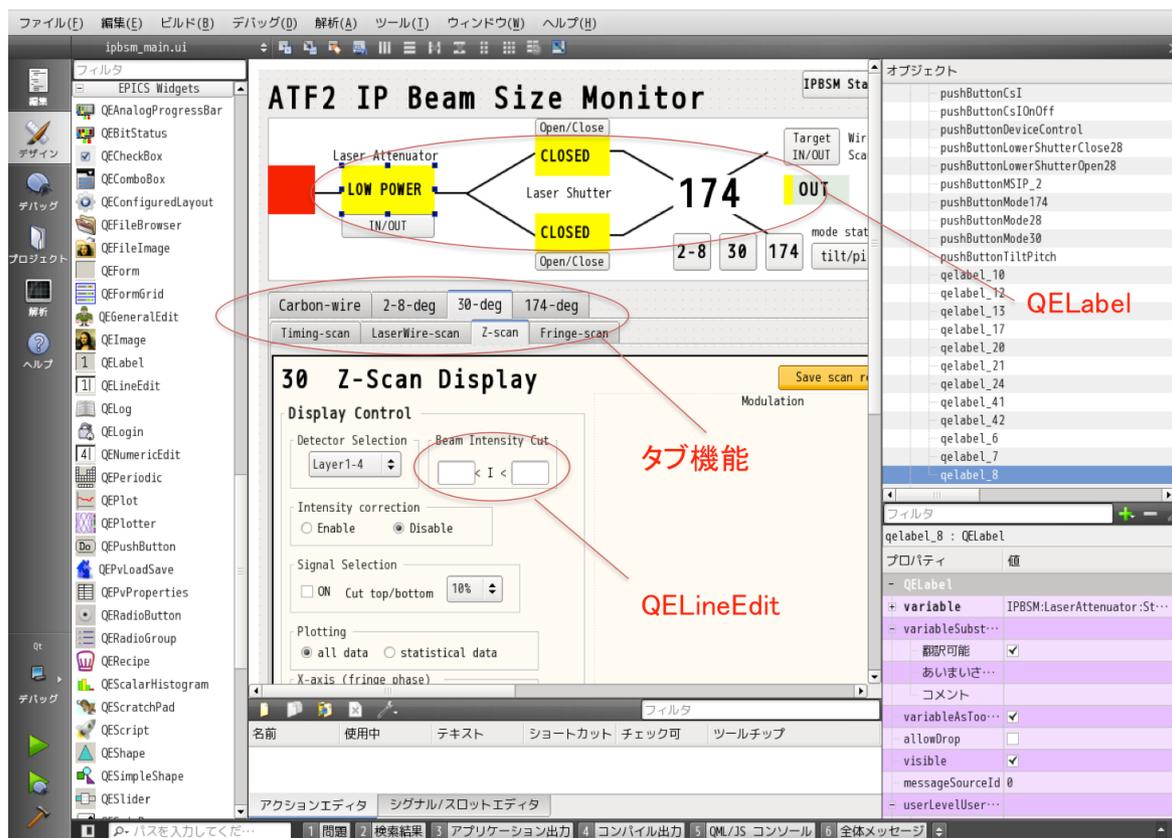


Figure 1: Development screen in QtCreator.

されている専用のテキストボックスやコンボボックスウィジェットを実装している。3.2項同様、これらのウィジェットのプロパティ設定を行うことにより、入力した値がダイレクトに EPICS レコードに反映される。シグナル・スロット機能を使つての Qt プログラミングコードを作成する手間が省ける。

3.4 プロット表示

データ解析用として以前から ROOT を使用してきた。そのためプロット表示には QtROOT を使用している。Qt 標準ウィジェットクラスを ROOT 用のウィジェットクラスに格上げしているため、表示用プログラミングコードは通常の ROOT と変わりなく開発できる。プロット表示に関してはオフライン用の画像解析ソフトにも流用している。

ROOT の描画部分は多機能で、マウスのクリック操作で図形を操作できたりする。その分、描写処理がシンプルではないため、X11 を介して別のホスト端末でリモート実行すると、ネットワークトラフィックの影響を受けやすくなってしまい、プロット表示に関する部分で極端に動作が遅くなる問題が頻発した。

3.5 制御プロセスの構成

アクチュエータやムーバーテーブルなどの各デバイスを EPICS レコードにアクセスすることによって制御できるように用意した。

開発当初は一つのプロセス内で、それらの EPICS レコードにアクセスし、各デバイスの制御を行い 1 イベントごとにプロットしていた。

しかし、3.4項で説明したとおり、リモート端末で Scan プログラムを実行した場合に、ROOT でプロットをさせる描画処理に時間を要して、ワンサイクルでの Scan 時間が遅くなってしまった。ATF ではビームの繰り返しタイミングを 3Hz で行っているのだが、

実際のところこのソフトにおいては同期できるタイミングが 3Hz 以下になってしまい、データ取得が高速でできなくなってしまい、実験の効率が上がらない問題がでてきてしまった。

そこでユーザー側が操作、描画させるクライアント部分と、Scan するためにデバイス制御を自動で行うサーバー部分に切り分けることにした。Figure 2 にそれを示す。

切り分けられたサーバー部分である自動 Scan プロセスは、クライアントである GUI からのパラメータを取得し、Scan する範囲を判断。その範囲内を決められたステップ数によってデバイス制御を行い、データ収集プロセスから測定結果を受け取り、Scan 表示用に用意された EPICS レコードにデータをアップデートする。

クライアント部分である GUI はサーバー側で EPICS レコードにアップデートされたデータをスレッド処理内で 2 秒ごとに取得し、プロットするようにした。この結果、ROOT の描画問題に関係なく Scan 処理が行えるようになり 3Hz 繰り返しでのデータ収集が可能になった。

4. まとめ

Qt を使用することにより、C++で開発が盛んに行われるようになった。クラスのカプセル化ができるようになったためソフトウェア開発の効率が上がった。

QtCreator を統合環境開発に組み込んでいるので、EPICS のデータを簡単に GUI 作成できるようになった。また、プログラミングのデバッグの効率が良くなった。

例にあげた制御ソフト以外にも、画像解析ソフトウェアや、Knob スキャンソフトウェアなど EPICS と Qt をベースにしたソフトウェアの開発を行った。実験で使用するソフトを比較的短時間で開発できるため ATF 実験の効率アップに貢献できている。

今後は Qt バージョン 5 以上に対応できるソフトウェアの開発ができるようにする予定である。そのためには画像プロットに使用するツールを ROOT 自身ではなく、QCustomPlot^[6]などの Qt に特化したプロット描画ツールに更新していく予定である

参考文献

- [1] <http://www.vista-control.com>
- [2] <http://www.aps.anl.gov/epics/>
- [3] <http://qt-project.org/>
- [4] <http://root.cern.ch/drupal/>
- [5] <http://www.epicsqt.org>
- [6] <http://www.qcustomplot.com>

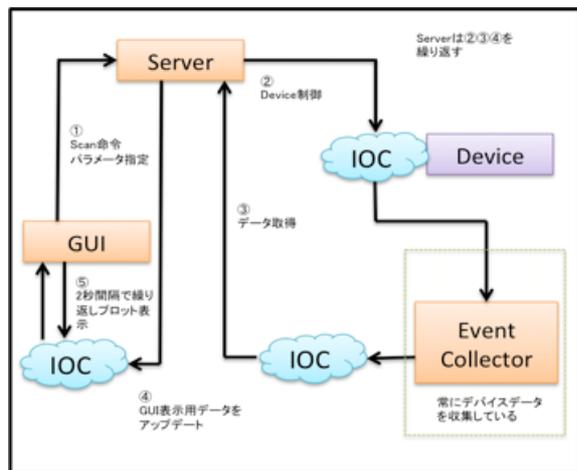


Figure 2: Overview diagram of a scan process.