

Pemrograman Qt 10

QTextEdit, QFile, QTextStream, QLabel, Membaca Standard Output Shell, Membaca Berkas, dan Membuat Berkas TXT

Ade Malsasa Akbar

2 Desember 2013

Ringkasan

Bismillahirrahmanirrahim. Tulisan ini menjelaskan bagaimana membuat sebuah aplikasi kecil *repo changer* yang bisa mengganti konfigurasi repositori Ubuntu kita dan bisa membaca isi `sources.list` kita secara mudah dengan Qt. Aplikasi ini masih bersifat percobaan dan sama sekali bukan versi final. Seperti biasa, pada akhir tulisan disertakan tautan unduh kode sumbernya. Semoga tulisan ini bermanfaat.

Daftar Isi

1	Persiapan	3
1.1	Spesifikasi Sistem	3
1.2	Daftar Kelas	3
1.3	Daftar Method	3
2	Teori	3
2.1	Arah Tulisan Ini	3
2.2	Apa Itu <i>Standard Output</i> ?	4
2.3	Antara <i>Standard Output</i> dan Program Ini	5
2.4	Mengenai <i>sources.list</i> Kita	5
3	Pemrograman	6
3.1	Kode	6
3.1.1	mainwindow.h	6
3.1.2	mainwindow.cpp	8
3.1.3	Qt Creator dan Kode	12
3.1.4	Hasil	13
4	Pembahasan	14
4.1	mainwindow.h	14
4.2	mainwindow.cpp	14
4.2.1	Fungsi Ganti Repo & Notifikasi QLabel	14
4.2.2	Fungsi Baca Berkas	16
5	Unduh Kode Sumber	16
6	Kesimpulan	17
7	Rujukan	17
8	Tentang Dokumen Ini	18
9	Tentang Penulis	18

1 Persiapan

1.1 Spesifikasi Sistem

- Ubuntu 12.04
- Qt Creator 2.4.1
- Qt 4.8.0 (32 bit)

1.2 Daftar Kelas

- QTextEdit
- QFile
- QProcess
- QStringList
- QString
- QLabel

1.3 Daftar Method

- start()
- waitForFinished()
- setText() (milik QLabel)
- setReadOnly() (milik QTextEdit)
- readAllStandardOutput() (milik QProcess)

Cuma 5? Ya, cuma 5.

2 Teori

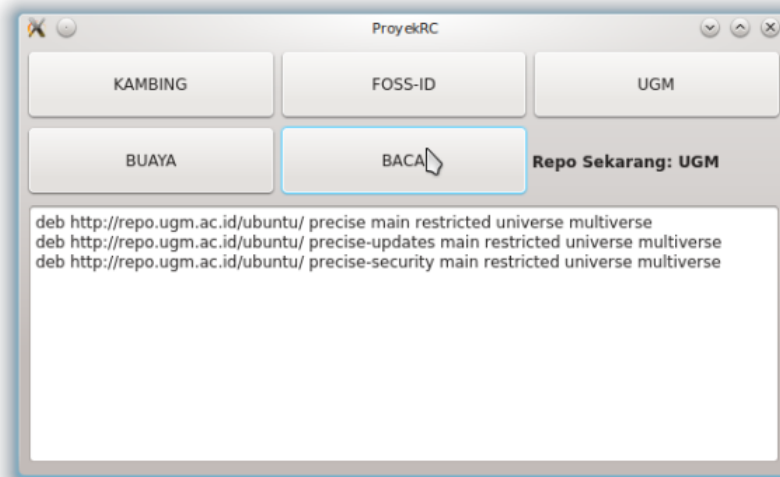
2.1 Arah Tulisan Ini

Tulisan ini akan menjelaskan bagaimana membuat program yang bisa melakukan hal-hal berikut

- Jika satu tombol diklik, maka isi sources.list berubah.
- Klik tombol Kambing maka isi sources.list berisi konfigurasi repo¹.

¹Repository, server yang berisi aplikasi lengkap Ubuntu.

- Kambing, klik tombol UGM maka berisi konfigurasi repo UGM, dan seterusnya.
- Klik tombol Baca maka isi sources.list akan ditayangkan di QTextEdit.
- Klik tombol Kambing, maka indikator (QLabel) menayangkan bahwa repo saat ini adalah Kambing dan seterusnya.



Gambar 1: Program yang Ingin Dibuat

Secara teknis, kita akan belajar bagaimana menggunakan QTextEdit, membaca standard output, memasukkan standard output ke dalam QTextEdit, membuat berkas teks dengan Qt, dan menulis string di C++.

2.2 Apa Itu *Standard Output*?

Standard output adalah teks yang dikeluarkan oleh program dari dalam sistem setelah melakukan suatu proses. Biasanya kita melihat standard output di dalam Terminal. Contohnya ketika kita gunakan perintah `ls -l` maka keluarannya seperti ini:

```
total 3960
-rw-r--r-- 1 master master 2673568 Mar 13 2011 kbbi.dict.dz
-rw-r--r-- 1 master master 1348181 Mar 13 2011 kbbi.idx
-rw-r--r-- 1 master master 297 Mar 13 2011 kbbi.ifo
-rw-r--r-- 1 master master 24330 Mar 13 2011 kbbi.syn
```

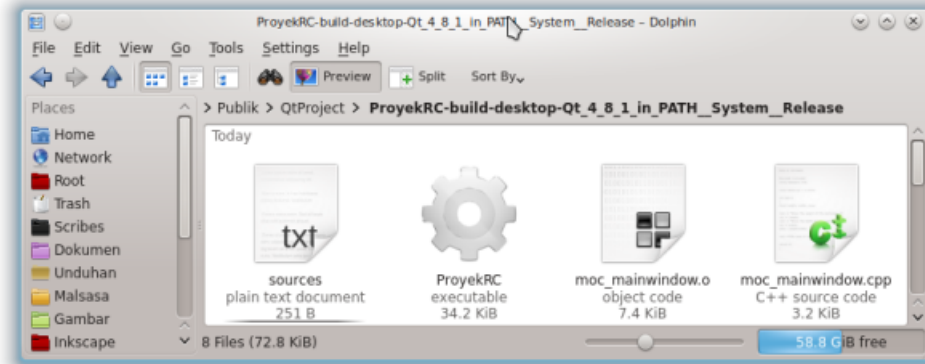
Keluaran di atas disebut standard output. Pendek kata, keluaran dari segala perintah Linux adalah standard output. Yang paling penting untuk diingat adalah standard output itu dikeluarkan secara broadcast (seperti siaran televisi) sehingga tidak hanya Terminal kita saja sebetulnya yang bisa menayangkan keluaran tersebut. Jika kita membuat aplikasi yang bisa membaca standard output, maka keluaran dari perintah Linux apa saja bisa ditayangkan di dalam aplikasi kita. Artinya, tidak perlu membuka Terminal lagi untuk menayangkan keluaran perintah yang kita panggil. Mirip Synaptic yang bisa menayangkan keluaran dpkg secara realtime ketika menginstal program.

2.3 Antara *Standard Output* dan Program Ini

Apa hubungan standard output dengan program kali ini? Hubungannya adalah program kita ini harus bisa membaca isi dari berkas teks (sources.list). Metode pembacaan berkas ada banyak sekali. Kita pilih satu metode saja, yakni menggunakan cat. Kita mengenal perintah cat (concatenate) untuk mengeluarkan isi teks ke layar Terminal. Kalau kita lakukan perintah cat sources.list, maka teks isinya akan ditayangkan di layar Terminal. Teks inilah standard output. Jika kita bisa menangkapnya, maka teks dari sources.list bisa ditayangkan di dalam aplikasi kita. Pembacaan standard output adalah rahasianya.

2.4 Mengenai *sources.list* Kita

Kita tidak akan menyentuh berkas sources.list yang asli pada /etc/apt/ karena ini adalah berkas sistem. Berkas ini berisi baris-baris alamat repo kita. Kalau terdapat kesalahan satu huruf saja di dalamnya, apt pasti menayangkan pesan error ketika kita menginstal aplikasi. Jika sudah error, maka kita harus membetulkannya secara manual. Selain itu, jika kita buka berkas sources.list buatan sendiri di /home, otomatis Synaptic Package Manager akan terbuka karena adanya asosiasi sistem. Walaupun sebenarnya tidak masalah, tetapi kita memilih jalan paling praktis yakni membuat berkas bernama sources. Ya, sources begitu saja tanpa akhiran .list. Berkas ini diibaratkan sebagai sources.list betulan sehingga apa yang bisa dilakukan dengan ini, pasti hasilnya sama jika dilakukan pada sources.list yang asli. Berkas ini akan dibuat di dalam direktori build yang diciptakan otomatis oleh Qt Creator. Dalam Ubuntu saya, berkas ini akan dibuat di direktori /home/master/Publik/QtProject/ProyekRC-build-desktop-Qt_4_8_1_in_PATH_System_Release/.



Gambar 2: Path Kompilasi Program

3 Pemrograman

3.1 Kode

3.1.1 mainwindow.h

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5  #include <QtGui>
6
7  namespace Ui {
8  class MainWindow;
9  }
10
11 class MainWindow : public QMainWindow
12 {
13     Q_OBJECT
14
15 public:
16     explicit MainWindow(QWidget *parent = 0);
17     ~MainWindow();
18     QGridLayout      *v1;
19     QVBoxLayout      *s1;
20     QVBoxLayout      *dl;
21     QWidget          *widget;
22     QPushButton      *pb_kambing;
23     QPushButton      *pb_fossid;
24     QPushButton      *pb_ugm;
25     QPushButton      *pb_buaya;

```

```
26     QPushButton      *pb_baca;
27     QTextEdit        *te_baca;
28     QLabel           *lb_repo;
29
30 public slots:
31     void ganti_ke_kambing();
32     void ganti_ke_fossid();
33     void ganti_ke_ugm();
34     void ganti_ke_buaya();
35     void baca_sources_list();
36
37 private:
38     Ui::MainWindow *ui;
39 };
40
41 #endif // MAINWINDOW_H
```

3.1.2 mainwindow.cpp

```
1 //ProyekRC = Proyek Repo Changer
2 //proyek mini ini dibuat sebagai bak pasir bagi saya untuk mempelajari I/
   0, string,
3 //dan pembuatan berkas teks dengan Qt
4 //16 Oktober 2013
5 #include "mainwindow.h"
6 #include "ui_mainwindow.h"
7
8 MainWindow::MainWindow(QWidget *parent) :
9     QMainWindow(parent),
10    ui(new Ui::MainWindow)
11 {
12     v1         =         new         QGridLayout();
13     s1         =         new         QVBoxLayout();
14     dl         =         new         QVBoxLayout();
15     widget     =         new         QWidget();
16     pb_kambing =         new         QPushButton("KAMBING");
17     pb_fossid  =         new         QPushButton("FOSS-ID");
18     pb_ugm     =         new         QPushButton("UGM");
19     pb_buaya   =         new         QPushButton("BUAYA");
20     pb_baca    =         new         QPushButton("BACA");
21     te_baca    =         new         QTextEdit;
22     lb_repo    =         new         QLabel;
23
24     v1->addWidget(pb_kambing, 1, 1);
25     v1->addWidget(pb_fossid, 1, 2);
26     v1->addWidget(pb_ugm, 1, 3);
27     v1->addWidget(pb_buaya, 2, 1);
28     v1->addWidget(pb_baca, 2, 2);
29     v1->addWidget(lb_repo, 2, 3);
30     v1->setHorizontalSpacing(3);
31     v1->setVerticalSpacing(3);
32     dl->addWidget(te_baca);
33
34     pb_kambing->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::
   Expanding);
35     pb_fossid->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::
   Expanding);
36     pb_ugm->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::Expanding)
   ;
37     pb_buaya->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::
   Expanding);
38     pb_baca->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::Expanding
   );
39     te_baca->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::Expanding
   );
40
```



```

41     pb_kambing->setMinimumHeight(55);        //pb_kambing->setMaximumWidth
        (133);
42     pb_fossid->setMinimumHeight(55);        //pb_fossid->setMaximumWidth
        (133);
43     pb_ugm->setMinimumHeight(55);          //pb_ugm->setMaximumWidth(133);
44     pb_buaya->setMinimumHeight(55);        //pb_buaya->setMaximumWidth
        (133);
45     pb_baca->setMinimumHeight(55);         //pb_baca->setMaximumWidth(133)
        ;
46     te_baca->setMinimumSize(575,199);
47
48     sl->addLayout(v1);
49     sl->addLayout(d1);
50
51     widget->setLayout(sl);
52
53     this->setCentralWidget(widget);
54
55     //saatnya mengasosiasikan tombol dengan fungsi
56     connect(pb_kambing, SIGNAL(clicked()), this, SLOT(ganti_ke_kambing()
        ));
57     connect(pb_fossid, SIGNAL(clicked()), this, SLOT(ganti_ke_fossid()));
58     connect(pb_ugm, SIGNAL(clicked()), this, SLOT(ganti_ke_ugm()));
59     connect(pb_buaya, SIGNAL(clicked()), this, SLOT(ganti_ke_buaya()));
60     connect(pb_baca, SIGNAL(clicked()), this, SLOT(baca_sources_list()));
61
62 }
63
64 MainWindow::~MainWindow()
65 {
66     delete ui;
67 }
68
69 void MainWindow::ganti_ke_kambing()
70 {
71     QStringList argumen;
72     argumen << "-c" <<
73         "echo 'deb http://kambing.ui.ac.id/ubuntu precise main
        restricted multiverse universe\n"
74         "deb http://kambing.ui.ac.id/ubuntu/ precise-update main
        restricted multiverse universe\n"
75         "deb http://kambing.ui.ac.id/ubuntu/ precise-security
        main restricted universe multiverse\n"
76         "deb http://kambing.ui.ac.id/ubuntu/ precise-backports
        main restricted universe multiverse' > sources";
77     /* C++ itu otomatis melakukan concatenate (penggabungan) untuk string
        yang dipisahkan barisnya seperti di atas
78     - perhatikan " dan ' di sana, asalnya baris-baris string itu satu
79     - Tanda ' ditanggguhkan sampai baris terakhir supaya seluruh baris
        dianggap sebaris oleh QStringList

```

```

80     - jangan lupa \n (newline) di setiap akhir baris karena sources.
      list itu berkas berbahaya
81     - sementara berkas keluaran tidak dinamakan _sources.list_ karena
      asosiasi penyunting teks dalam sistem
82     - sumber: http://stackoverflow.com/questions/4276026/how-to-write-a-qstring-on-several-lines
83     16 Oktober 2013 */
84
85     QProcess perintah;
86     perintah.start("/bin/bash", argumen);
87     perintah.waitForFinished(-1);
88
89     //baru terpikir pada 17 Oktober
90     this->lb_repo->setText("<b>Repo Sekarang: Kambing</b>");
91 }
92
93 void MainWindow::ganti_ke_fossid()
94 {
95     QStringList argumen;
96     argumen << "-c" <<
97         "echo 'deb http://dl2.foss-id.web.id/ubuntu/ precise main
      restricted universe multiverse\n"
98         "deb http://dl2.foss-id.web.id/ubuntu/ precise-updates
      main restricted universe multiverse\n"
99         "deb http://dl2.foss-id.web.id/ubuntu/ precise-security
      main restricted universe multiverse\n"
100        "deb http://dl2.foss-id.web.id/ubuntu/ precise-
      backports main restricted universe multiverse' > sources";
101
102     QProcess perintah;
103     perintah.start("/bin/bash", argumen);
104     perintah.waitForFinished(-1);
105
106     this->lb_repo->setText("<b>Repo Sekarang: FOSS-ID</b>");
107 }
108
109 void MainWindow::ganti_ke_ugm()
110 {
111     QStringList argumen;
112     argumen << "-c" <<
113         "echo 'deb http://repo.ugm.ac.id/ubuntu/ precise main
      restricted universe multiverse\n"
114         "deb http://repo.ugm.ac.id/ubuntu/ precise-updates main
      restricted universe multiverse\n"
115         "deb http://repo.ugm.ac.id/ubuntu/ precise-security
      main restricted universe multiverse' > sources";
116
117     QProcess perintah;
118     perintah.start("/bin/bash", argumen);
119     perintah.waitForFinished(-1);

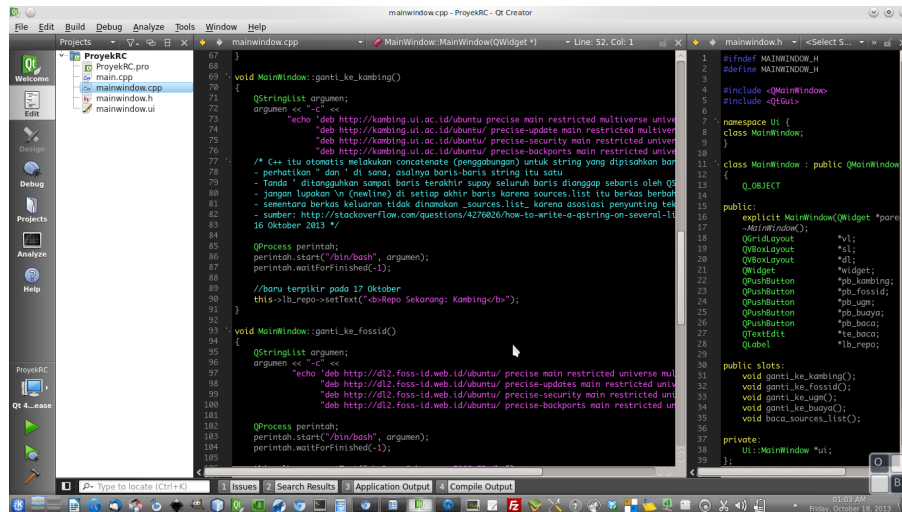
```

```

120
121     this->lb_repo->setText("<b>Repo Sekarang: UGM</b>");
122 }
123
124 void MainWindow::ganti_ke_buaya()
125 {
126     QStringList argumen;
127     argumen << "-c" <<
128         "echo 'deb http://buaya.klas.or.id/ubuntu/ precise main
129         restricted universe multiverse\n"
130         "deb http://buaya.klas.or.id/ubuntu/ precise-updates
131         main restricted universe multiverse\n"
132         "deb http://buaya.klas.or.id/ubuntu/ precise-security
133         main restricted universe multiverse' > sources";
134
135     QProcess perintah;
136     perintah.start("/bin/bash", argumen);
137     perintah.waitForFinished(-1);
138
139     this->lb_repo->setText("<b>Repo Sekarang: Buaya</b>");
140 }
141
142 void MainWindow::baca_sources_list()
143 {
144     //kode ini saya buat dengan menebak saja
145     QString isi_sources_list;
146     QProcess perintah_baca_sources;
147     perintah_baca_sources.start("cat sources");
148     perintah_baca_sources.waitForFinished(-1);
149     isi_sources_list = perintah_baca_sources.readAllStandardOutput();
150
151     this->te_baca->setText(isi_sources_list);
152     this->te_baca->setReadOnly(true);
153 }

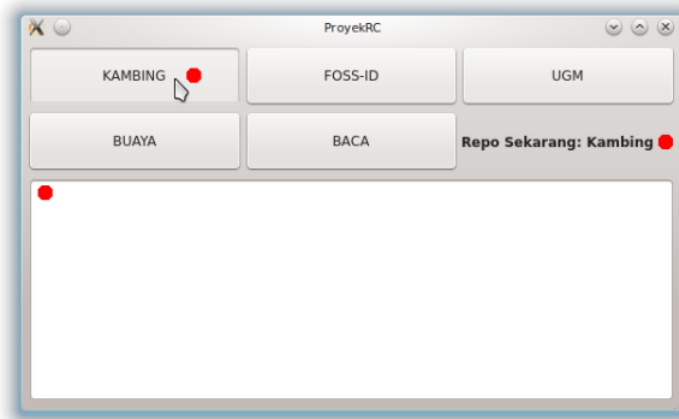
```

3.1.3 Qt Creator dan Kode



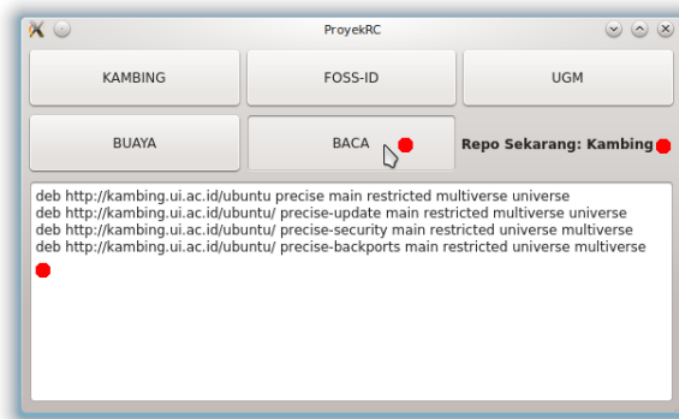
Gambar 3: Qt Creator Menunjukkan Kode Program

3.1.4 Hasil



Gambar 4: Hasil 1

Jika tombol KAMBING diklik, maka QLabel akan memberikan informasi bahwa repo sekarang adalah Kambing. Perhatikan, QTextEdit masih kosong.



Gambar 5: Hasil 2

Barulah setelah tombol BACA diklik, maka QTextEdit membaca isi dari berkas sources kita.

4 Pembahasan

Pembahasan untuk kode sumber kali ini hanya berpusat pada `mainwindow.cpp` saja.

4.1 `mainwindow.h`

Isi header kali ini masih selaras dengan header program sebelumnya. Silakan merujuk [ke sana](#) jika Anda belum paham cara deklarasi dalam header ini.

4.2 `mainwindow.cpp`

Ada 2 bagian saja yang penting dalam berkas `cpp` kita kali ini. Bagian kelas utamanya (pembentukan jendela) tidak perlu diperhatikan. Namun jika Anda belum paham soal bagaimana membentuk jendela dan membuat koneksi SIGNAL & SLOT dengan Qt, silakan merujuk [ke sini](#) dan [ke sini](#).

4.2.1 Fungsi Ganti Repo & Notifikasi QLabel

```
1 void MainWindow::ganti_ke_kambing()
2 {
3     QStringList argumen;
4     argumen << "-c" <<
5         "echo 'deb http://kambing.ui.ac.id/ubuntu
6             precise main restricted multiverse universe\n"
7             "deb http://kambing.ui.ac.id/ubuntu/
8             precise-update main restricted multiverse universe\n"
9             "deb http://kambing.ui.ac.id/ubuntu/
10            precise-security main restricted universe multiverse\n"
11            "deb http://kambing.ui.ac.id/ubuntu/
12            precise-backports main restricted
13            universe multiverse' > sources";
14
15     QProcess perintah;
16     perintah.start("/bin/bash", argumen);
17     perintah.waitForFinished(-1);
18
19     this->lb_repo->setText("<b>Repo Sekarang: Kambing</b>");
20 }
```

Kode di atas adalah fungsi buatan sendiri. Nama fungsinya `ganti_ke_kambing()`. Isinya adalah deklarasi lokal dan fungsi-fungsi Qt untuk mengubah isi berkas source dengan teks pengaturan repositori Kambing UI. Cara kerja fungsinya adalah menggunakan `QProcess` sebagai pemanggil `/bin/bash` dan `QStringList` sebagai penampung argumen yakni perintah `bash` yang sebenarnya dipakai. Perintah `bash` yang dipakai adalah `echo` dengan memakai redirector `'>'` untuk membuat berkas `sources`. Jika Anda belum memahami metode pemanggilan perintah

shell ini, silakan merujuk [ke sini](#).

Kode di atas jika dijalankan, maka akan mengosongkan seluruh isi sources lalu menyimpannya dengan teks yang tersebut dalam echo. Ini karena sifat *redirector* '>' yang me-replace isi teks. Tidak berhenti sampai situ. Pengguna memerlukan notifikasi yang menandakan konfigurasi repo telah berubah ke server tertentu. Maka dipanggilah fungsi `setText()` milik objek `lb_repo` (QLabel) dalam jendela (`this`). Kodenya adalah `this->lb_repo->setText(Repo Sekarang: Kambing);`. Cara kerja ini sama untuk fungsi-fungsi `ganti_ke_*`() di bawahnya.

Masalah C++ yang penting untuk diketahui adalah model penyimpanan string di dalam objek `QString`. Untuk memudahkan penulisan, kita harus menaruh satu baris URL repo di bawah baris yang lain. Kalau kita tulis memanjang ke samping, kode kita jadi tidak nyaman dibaca. Maka kita memakai salah satu fitur C++ itu sendiri yakni sanggup menggabungkan (*concatenate*) string yang terpisah baris tetapi setiap baris diapit dengan tanda petik ganda ("") tanpa mengakhirkannya apa-apa setelah " terakhir. Coba perhatikan kode ini:

```
1 argumen << "-c" <<
2 "echo 'deb http://kambing.ui.ac.id/ubuntu precise main restricted
   multiverse universe\n"
3 "deb http://kambing.ui.ac.id/ubuntu/ precise-update main restricted
   multiverse universe\n"
4 "deb http://kambing.ui.ac.id/ubuntu/ precise-security main restricted
   universe multiverse\n"
5 "deb http://kambing.ui.ac.id/ubuntu/ precise-backports main restricted
   universe multiverse' > sources";
```

Mari perhatikan perbedaan pemakaian tanda ' (petik tunggal) dan " (petik ganda) dalam kode ini. Perhatikan pemakaian " sebelum echo dan " sesudah sources. Mereka berdua adalah pengapit seluruh baris. Perhatikan ' sebelum deb pada baris pertama dan ' sesudah multiverse pada baris terakhir. Itu adalah pengapit string bagi perintah echo. Jadi, echo punya string sendiri dan variabel argumen di sini punya string sendiri juga. Ada string di dalam string. Dan otomatis C++ menganggap semua baris sebagai satu string karena "" pada setiap baris hingga akhir tanpa ada tambahan karakter di setiap akhir baris. Ingat, harus ada *escape character* untuk newline (Enter) pada akhir setiap string karena kita harus mengikuti standar apt dalam mengisi sources.list yakni satu URL dalam satu baris, tidak boleh salah. Begitu caranya membuat banyak baris dalam 1 string di dalam C++.

4.2.2 Fungsi Baca Berkas

```
1 void MainWindow::baca_sources_list()
2 {
3     QString isi_sources_list;
4     QProcess perintah_baca_sources;
5     perintah_baca_sources.start("cat sources");
6     perintah_baca_sources.waitForFinished(-1);
7     isi_sources_list = perintah_baca_sources.readAllStandardOutput();
8
9     this->te_baca->setText(isi_sources_list);
10    this->te_baca->setReadOnly(true);
11 }
```

Kode ini berbeda dengan semua fungsi ganti_ke_*() sebelumnya. Kode ini adalah fungsi yang dikoneksikan dengan tombol BACA (pb_baca). Namanya baca_sources_list(). Isinya deklarasi lokal dan fungsi bagi QProcess untuk membaca standard output dari perintah cat sources.

Yang perlu diperhatikan adalah baris:

```
7 isi_sources_list = perintah_baca_sources.readAllStandardOutput();
```

Maksud baris tersebut adalah memasukkan apa yang ditangkap oleh method readAllStandardOutput() di dalam objek perintah_baca_sources, ke dalam variabel isi_sources_list. Ini cara kita menangkap standard output. Setelah itu, kita memasukkan hasil tangkapan tadi ke dalam QTextEdit dengan kode:

```
9 this->te_baca->setText(isi_sources_list);
```

maka jadilah apa yang kita inginkan yaitu menampilkan isi berkas sources ke dalam aplikasi kita. Ya, rahasianya adalah dengan memakai perintah bash lalu membaca standard output-nya. Masih banyak cara lain yang belum saya mengerti. Sementara, kita akan memakai cara ini untuk program-program berikutnya.

5 Unduh Kode Sumber

Program kali ini bernama ProyekRC (Repo Changer). Silakan unduh dan impor di Qt Creator Anda.

- Tautan: <http://otodidak.freesever.me/tarball/ProyekRC.tar.gz>
- Ukuran: 50 KB

6 Kesimpulan

Pembacaan standard output di Qt bisa dilakukan dengan QProcess melalui method `readAllStandardOutput()`. QTextEdit bisa dipakai untuk menerima penulisan teks dari pengguna maupun menayangkan teks dari program. Bahkan bisa menayangkan standard output.

7 Rujukan

<http://stackoverflow.com/questions/4276026/how-to-write-a-qstring-on-several-lines>

8 Tentang Dokumen Ini

Dokumen ini adalah versi PDF dari posting asli <http://malsasa.wordpress.com/2013/10/12/pemrograman-qt-10-qtedit-qfile-qtstream-qlabel-membaca-standard-output-shell-membaca-berkas-dan-membuat-berkas-txt/>. Dokumen ini disusun ulang dengan L^AT_EX dengan antarmuka Gummi di atas Ubuntu 12.04. Dokumen ini adalah bagian dari usaha kecil memperbanyak panduan Linux dalam Bahasa Indonesia. Dokumen ini tidaklah bebas dari kesalahan yang membuat penulis memohon maaf dan mengajak Anda untuk menulis dokumen yang lebih baik.

Penulis berusaha membuat versi PDF ini semirip mungkin dengan versi HTML. Namun penulis menemukan bahwa kode-kode sumber dalam dokumen ini bisa disalin-tempel (copy-paste) dengan sempurna pada Evince PDF Viewer dan tidak sempurna (membutuhkan sedikit penyuntingan tambahan) pada Okular PDF Viewer. Maka penulis menyarankan Anda untuk menggunakan Evince atau menulis sendiri tiap-tiap kode sumber yang ada atau mengunduh langsung versi paket kode sumbernya pada alamat yang tersebut di atas. Penulis memohon maaf atas ketidaknyamanan ini.

Catatan teknis keterbatasan: listings, fontenc, [T1], accsupp, minted (ada bug pada Gummi), Pygments.

Seluruh informasi repositori dalam tulisan ini diambil dari tulisan <http://malsasa.wordpress.com/2013/10/15/daftar-lengkap-repositori-lokal-ubuntu-12-04-plus-cara-mengaturnya>.

9 Tentang Penulis

Penulis adalah warga Forum Ubuntu Indonesia dan aktif di forum-forum Linux lain di Indonesia. Penulis mendukung pendidikan perangkat lunak legal (terutama FOSS) untuk masyarakat. Penulis menyediakan buku-buku panduan Linux untuk pemula maupun ahli untuk diunduh secara gratis². Penulis bisa dihubungi via SMS di nomor 0896 7923 7257.

²<http://malsasa.wordpress.com/pdf>