Using Maemo 5 thumbnails

Overview

This article shows how to use Maemo 5 thumbnails

Basic Idea

Maemo 5 has Tracker search engine running, which is responsible for a thumbnail creation. Thumbnails are stored in $HOME/.thumbnails/cropped on Maemo 5 platform and $HOME/.thumbnails/normal See Thumbnail Managing Standard for details.

That's all about theory, now we are going to create QDirModel to get list of the files, QListView to show the files and custom class derived from QAbstractItemDelegate to draw thumbnails in the item view.

main.cpp:

```
#include <QApplication>
#include <QDirModel>
#include <QListView>
#include <QDebug>

#include "filedelegate.h"

int main (int argc, char **argv)
{
    QApplication app(argc, argv);

    QDirModel model;
    FileDelegate delegate;
    QListView view;
    view.setModel(&model);
    view.setViewMode(QListView::IconMode);
    view.setResizeMode(QListView::Adjust);
#ifdef Q_WS_MAEMO_5
    view.setRootIndex(model.index(QDir::homePath() + "/.thumbnails/cropped"));
#else
    view.setRootIndex(model.index(QDir::homePath() + "/.thumbnails/normal"));
#endif
    view.setItemDelegate(&delegate);
    view.show();

    return app.exec();
}
```

Class Declaration

filedelegate.h

```
#ifndef FILEDELEGATE_H
#define FILEDELEGATE_H

#include <QItemDelegate>

class FileDelegate : public QAbstractItemDelegate
{
    Q_OBJECT

public:
    FileDelegate(QObject *parent = 0);
    void paint(QPainter *painter, const QStyleOptionViewItem &option,
               const QModelIndex &index) const;
    QSize sizeHint(const QStyleOptionViewItem &option,
                   const QModelIndex &index ) const;
};

#endif
```

Class Implementation

filedelegate.cpp:

```
#include <QDirModel>
#include <QPainter>
#include <QDebug>

#include "filedelegate.h"

FileDelegate::FileDelegate(QObject *parent)
    : QAbstractItemDelegate(parent)
{
}

void FileDelegate::paint(QPainter *painter, const QStyleOptionViewItem &option,
                         const QModelIndex &index) const
{
    const QDirModel *model = qobject_cast<const QDirModel *>(index.model());
```


(C) Copyright Nokia 2013. All rights reserved.

```
    QPixmap pixmap(model->filePath(index));
    painter->drawPixmap(option.rect, pixmap);
}

QSize FileDelegate::sizeHint(const QStyleOptionViewItem &option,
                             const QModelIndex &index) const
{
    Q_UNUSED(option);

    const QDirModel *model = qobject_cast<const QDirModel *>(index.model());
    QImage image(model->filePath(index));
    return image.size();
}
```


(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.