QString和翻译

柴强

管理字符串

- C中的字符串

```
char *text = "Hello world!";
```

- QString是Qt中处理字符相关功能的类
  - 处理Unicode和字符编解码器
  - 性能优化
- 建立QString

```
QString str( “Hello” );
str = “def”
```

建立QString

- QString重载了operator +

```
QString res = "Hello  “ + name +
     “, the value is " + QString::number(42);
```

- QStringBuilder

```
QString res = "Hello  “ % name %
     “, the value is " % QString::number(42);
```

- QString的arg函数功能

```
QString res = QString("Hello %1, the value is %2")
    .arg(name)
    .arg(42);
```

Qt字符处理

- 数据可以从字符串中获取
- 数字
  - int val = QString::toInt();
  - float val = QString::toFloat();
- 字符串
  - QByteArray bytes = str.toLatin1();
  - char * raw = bytes.data();

Qt的数据结构

- 连续容器：QList,QLinkedList,QQueue，QStack
- 关联容器：QMap,QHash,QSet
- 比std::STL快速轻灵
- 如果熟悉STL可以继续使用

使用数据结构

- Using QList

```
QList<QString> list;
list << "one" << "two" << "three";
QString item1 = list[1]; // "two"
for(int i=0; i<list.count(); i++) {
  const QString &item2 = list.at(i);
}
int index = list.indexOf("two"); // returns 1
```

- Using QMap

```
QMap<QString, int> map;
map["Norway"] = 5; map["Italy"] = 48;
int value = map["France"]; // inserts key if not exists
if(map.contains("Norway")) {
  int value2 = map.value("Norway"); // recommended lookup
}
```

翻译

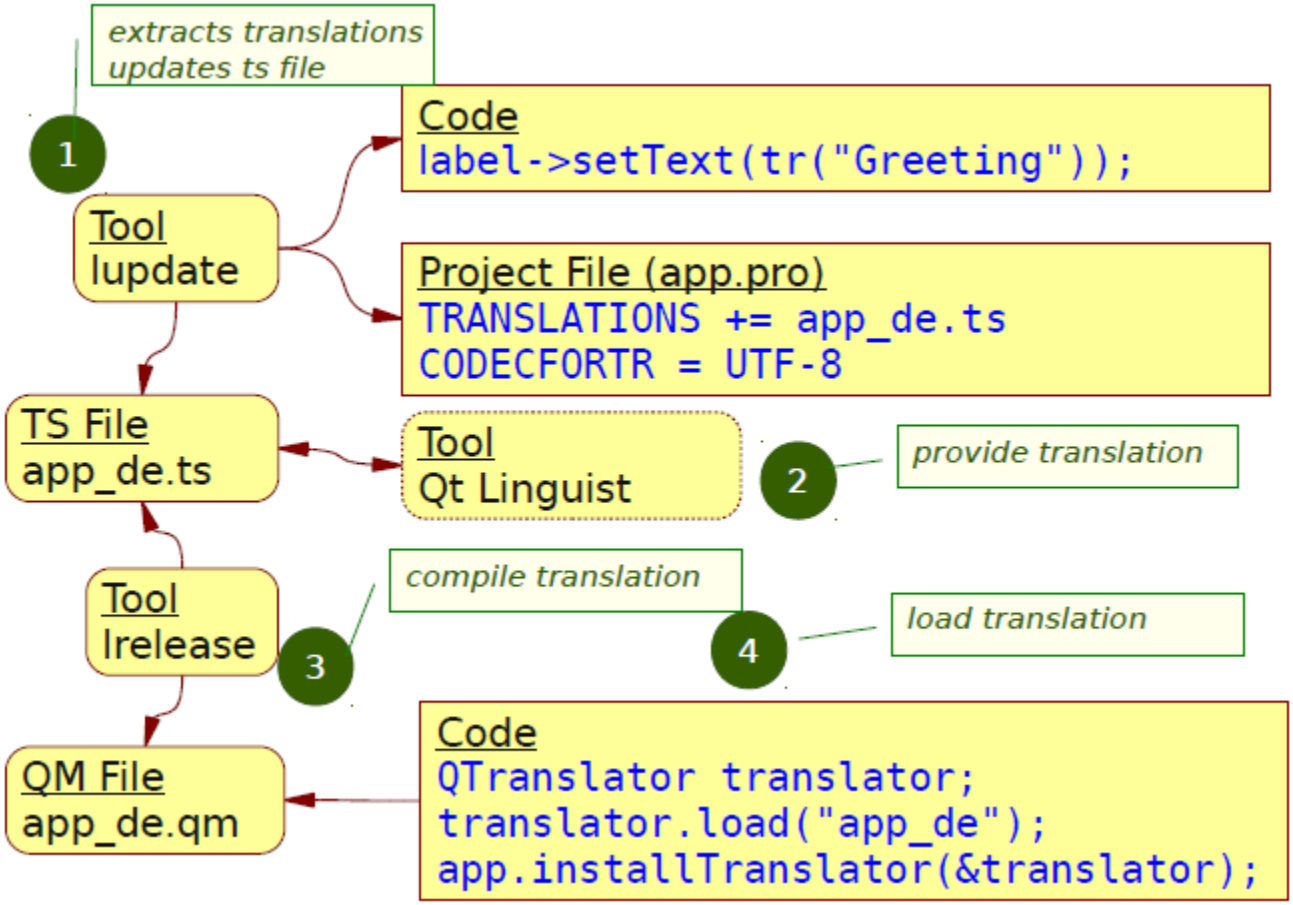
- 在代码中使用tr()函数
- 生成ts文件
  - lupdate HelloQt.pro
- 保证文件格式为UTF-8
  - 使用记事本保存
- 生成qm文件
  - lrelease ts_file -qm qm_file

加载翻译文件

- 建立资源文件
  - 修改pro文件:RESOURCES += helloqt.qrc
- 加载qm文件

```
QTranslator qtTranslator;
qtTranslator.load("helloqt_zh_CN.qm",":/");
app.installTranslator(&qtTranslator);
```

翻译步骤


extracts translations
updates ts file
1
Code
label->setText(tr("Greeting"));
Tool
lupdate
Project File (app.pro)
TRANSLATIONS += app_de.ts
CODECFORTR = UTF-8
TS File
app_de.ts
Tool
Qt Linguist
2
provide translation
compile translation
Tool
lrelease
3
4
load translation
QM File
app_de.qm
Code
QTranslator translator;
translator.load("app_de");
app.installTranslator(&translator);

Q & A