

How to enable QML Import Traces

Introduction

When developing with Qt Quick, it is possible to enable the QML's import traces to improve debug traces and get more information on what components are loaded in and their respective order.

Using Environment Variable

QML_IMPORT_TRACE environment variable can be set to enable QML's import traces

QML_IMPORT_TRACE with Qt

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

    // set the value of the environment variable QML_IMPORT_TRACE as 1
    QByteArray data = "1";
    qputenv("QML_IMPORT_TRACE", data);

    QmlApplicationViewer viewer;
    ...
    return app.exec();
}
```

Output

Now you would see traces for you QML snippet, below its shows QtQuick1.0 is loaded and then respective components are resolved in order of Rectangle, Text and MouseArea respectively

```
QDeclarativeImportDatabase::addImportPath: "C:\QtSDK\Simulator\Qt\mingw\imports"
QDeclarativeImportDatabase::addImportPath: "C:/Documents and Settings/Owner/helloworld1-
build-simulator/debug"
QDeclarativeImports(file:///.../helloworld1-build-
simulator/qml/helloworld1/main.qml)::addImport: "." -1.-1 File as ""
QDeclarativeImports(file:///.../helloworld1-build-
simulator/qml/helloworld1/main.qml)::addImport: "QtQuick" 1.0 Library as ""
QDeclarativeImports(file:///.../helloworld1-build-
simulator/qml/helloworld1/main.qml)::resolveType: "Rectangle" => "QDeclarativeRectangle"
QDeclarativeImports(file:///.../helloworld1-build-
simulator/qml/helloworld1/main.qml)::resolveType: "Text" => "QDeclarativeText"
QDeclarativeImports(file:///.../helloworld1-build-
simulator/qml/helloworld1/main.qml)::resolveType: "MouseArea" => "QDeclarativeMouseArea"
```

