Archived:Remote Compiler troubleshooting

Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The Qt Remote Compiler service has been discontinued.

Getting Started

Network problems (proxy)

These problems exhibit themselves by not being able to authenticate with the service, not seeing any targets in "Remote Compiler" build configuration, or not being able to initiate a build.

Typically, these are caused by being behind a proxy, and running Linux. If you are using Linux and behind a proxy, you need to set HTTP_PROXY environment variable before launching Qt Creator: The section below shows how to do this where 192.168.220.6 is IP address of proxy server and 8080 is proxy port. Both are separated via colon.

```
export http_proxy="http://192.168.220.6:8080/"
export https_proxy="http://192.168.220.6:8080/"
./QtSDK/QtCreator/bin/qtcreator
```

(Please adjust above settings according to your own proxy setup)

Daily usage

Build failure (missing files)

If you see a build failing because of some files (like translations) are missing, then ensure that they are specified in OTHER_FILES section in the .pro file. Remote Compiler sends out all the files it thinks are part of the project through parsing the .pro files, but the process is not foolproof and manual additions are sometimes needed. As an example, a project may want to ensure that translations get sent:

```
# helloworld.pro

OTHER_FILES += helloworld_en.ts helloworld_zh_CN.ts helloworld_en.qm helloworld_zh_CN.qm
```


(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.