

## PROTOCOLE MQTT – Réalisation d'un client avec Qt

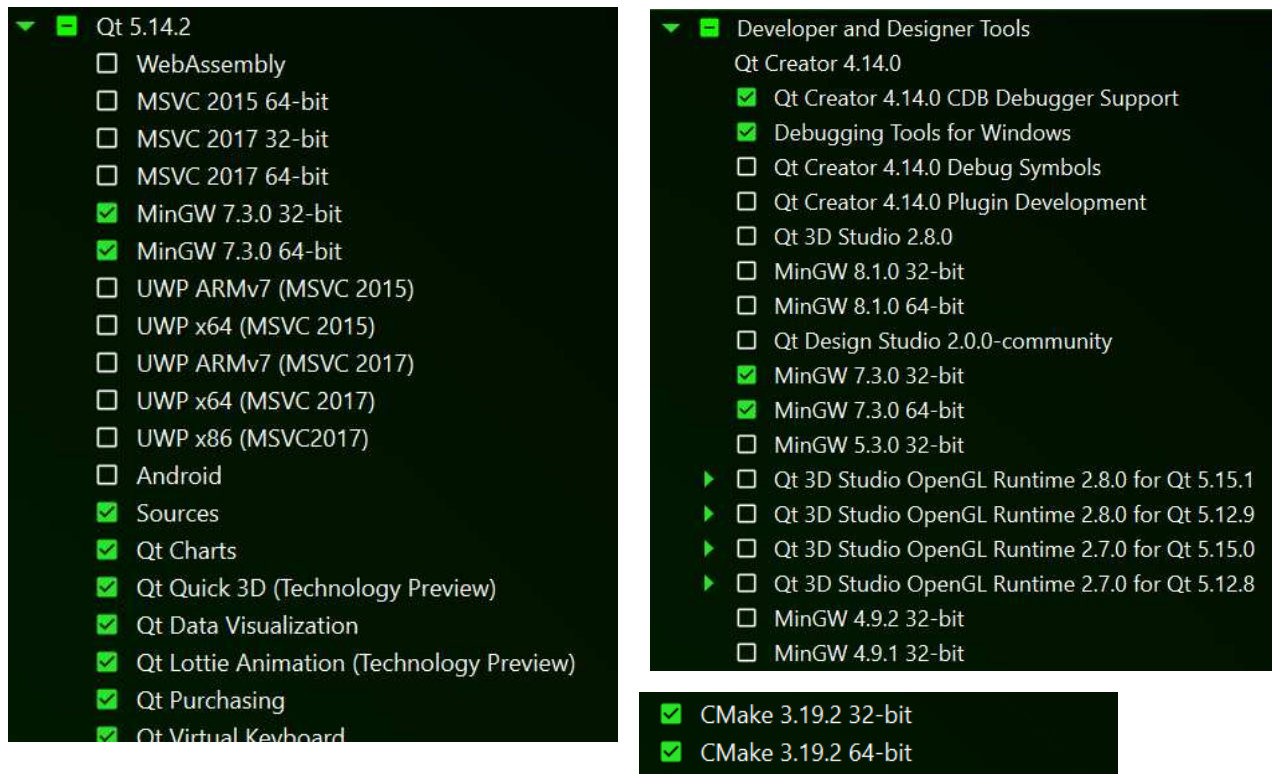
Ce TP s'effectue sous Windows mais peut être facilement adapté sous Linux.

**Prérequis** : programmation objet en C++, bibliothèque Qt, et EDI Qt créateur.

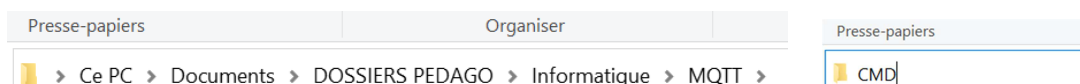
La bibliothèque MQTT n'est pas incluse dans la version gratuite de Qt. Il est cependant possible de l'y ajouter.

Un interpréteur PERL est nécessaire pour installer QtMQTT, <https://www.perl.org/get.html>.

La dernière version de MQTT pour Qt pose parfois des problèmes de compilation. **Il est préférable d'utiliser la version 5.14.2.**



Dans Qt\5.14.2\Src\ ouvrir une fenêtre de commande (entrer CMD dans le chemin du dossier de la fenêtre Windows) :

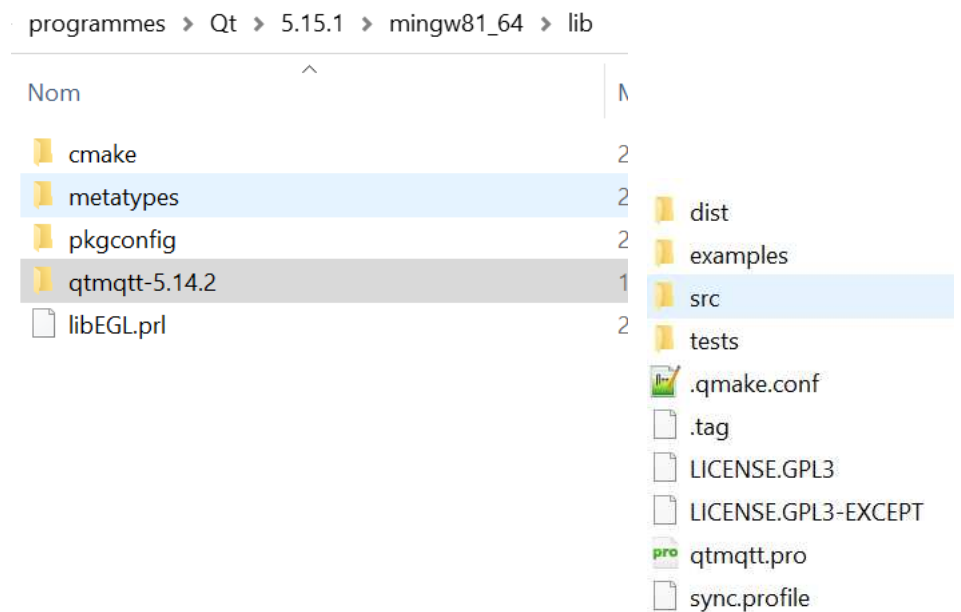
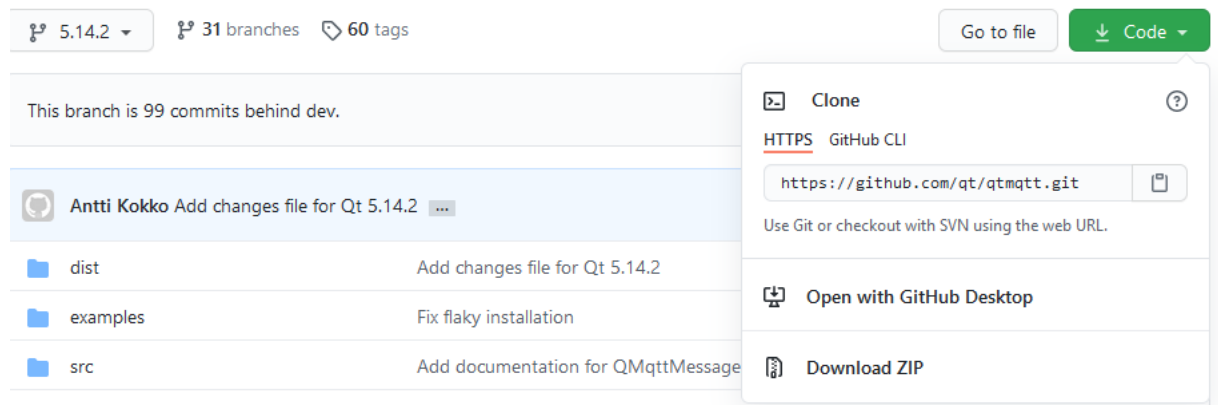


Si vous disposez de GIT :

```
git clone https://code.qt.io/qt/qtmqtt.git --branch 5.14.2
```

sinon récupérer le zip ici <https://github.com/qt/qtmqtt/tree/5.14.2> et créer un dossier qtmqtt dans SCR,

## PROTOCOLE MQTT – Réalisation d'un client avec Qt

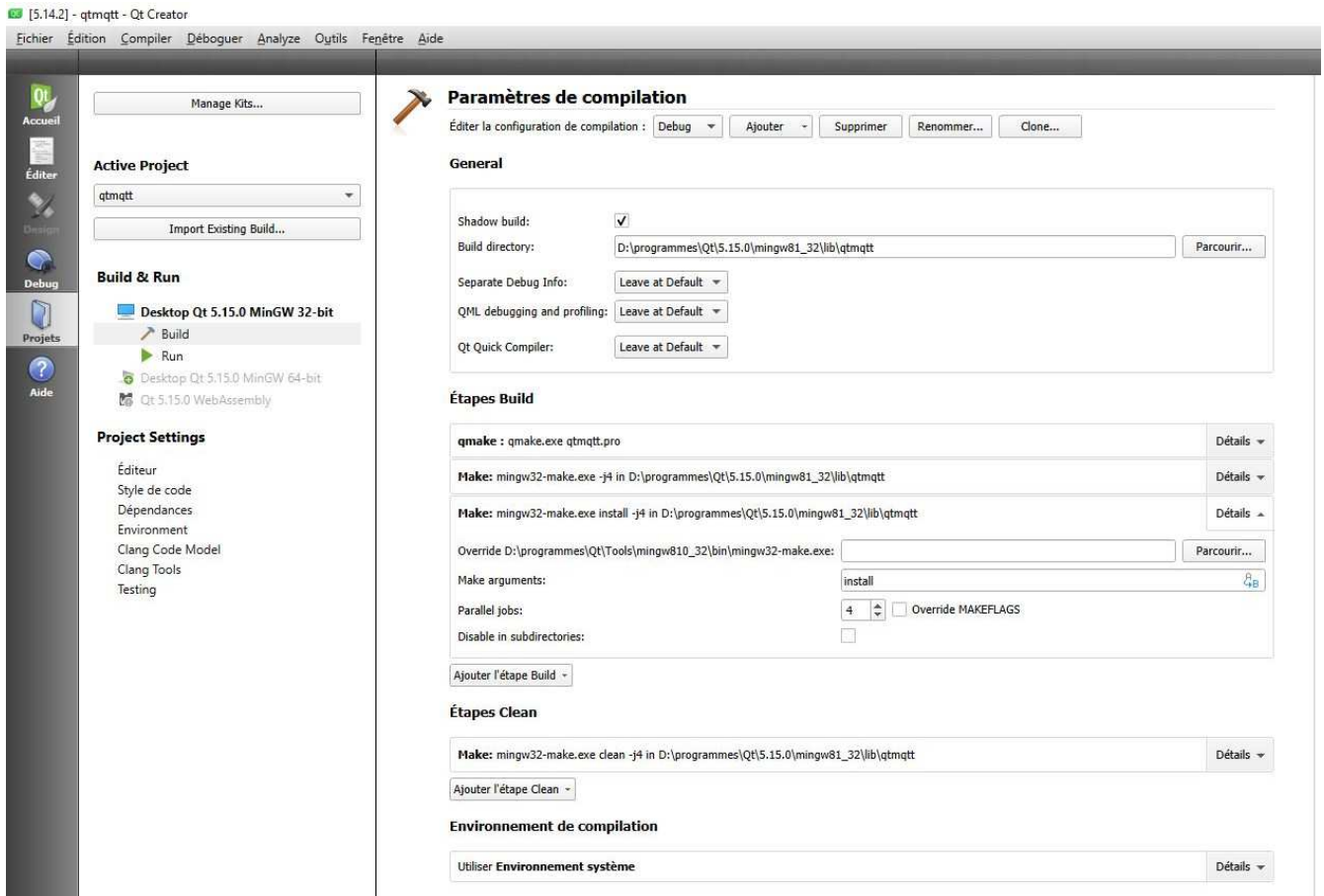


et le décompresser dans le dossier MQTT.

Ouvrir avec un explorateur le dossier qtmqtt et lancer MQTT.pro.

## PROTOCOLE MQTT – Réalisation d'un client avec Qt

Ajouter une étape build avec l'argument INSTALL :

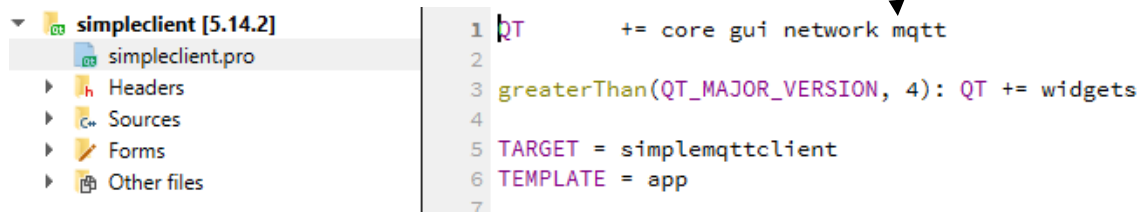


Lancer la compilation (cliquer sur le marteau).

S'il n'y a pas d'erreur de compilation, la bibliothèque MQTT est installée.

Pour l'utiliser il faudra s'assurer que network et mqtt sont liés à Qt dans le fichier de configuration.pro du projet.

Exemple :



qtmqtt a été installé dans le dossier SRC de Qt, **il n'est pas possible** de compiler les projets vers un exécutable dans ce dossier.

## Premier essai de QtMqtt

Recopier le dossier simpleclient vers votre dossier de travail :

programmes > Qt > 5.14.2 > Src > qtmqtt > exemples > mqtt

Nom	Modifié le
build-simpleclient-Desktop_Qt_5_14_2_MinGW_...	02/02/20;
consolepubsub	02/02/20;
doc	02/02/20;
quickpublication	02/02/20;
quicksubscription	02/02/20;
simpleclient	03/02/20;
subscriptions	02/02/20;
websocketsubscription	02/02/20;
mqtt.pro	02/02/20;

exemples > mqtt > simpleclient

Nom
main.cpp
mainwindow.cpp
mainwindow.h
mainwindow.ui
simpleclient.pro
simpleclient.pro.user

Dans le dossier simpleclient de votre espace de travail lancer simpleclient.pro

Compiler et exécuter le projet en mode release ou debug (flèche verte)

Entrer **test.mosquitto.org** comme hôte mqtt :

Cliquer "connect" ;

State 0 : Deconnecté ;

State 1 : Connexion ;

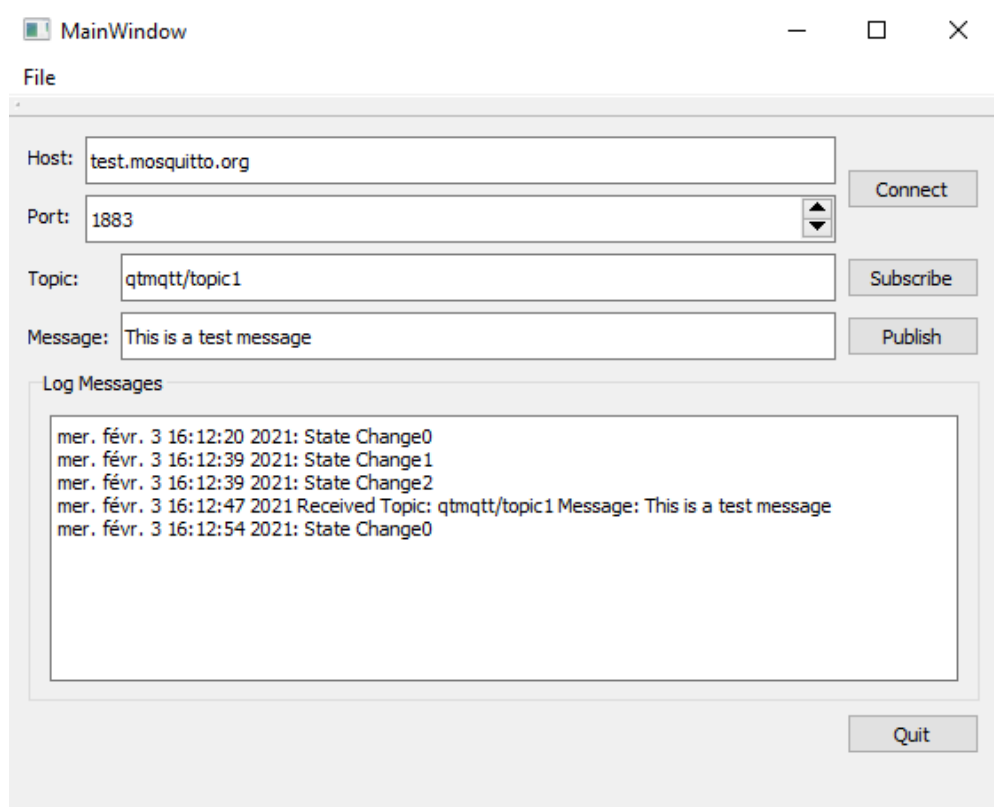
State 2: Connecté.

Cliquer "Subscribe" pour un abonnement au topic topic1.

Cliquer "Publish" le message est envoyé à mosquitto qui le retourne puisque l'on est abonné au topic.

Cliquer "Disconnect" pour déconnecter de mosquitto, le state repasse à 0.

## PROTOCOLE MQTT – Réalisation d'un client avec Qt



Le fichier mainwindow.cpp présente un exemple de méthodes de la bibliothèque qtmqtt pour créer un client mqtt.

```
m_client = new QmqttClient(this);      m_client est une instance d'un objet de type QmqttClient
m_client->setHostname(ui->lineEditHost->text());  on entre ici le nom du broker
m_client->setPort(ui->spinBoxPort->value());      on entre ici le numéro du port
```

Pour donner l'adresse du broker sans passer par lineEdit :

```
m_client->setHostname("test.mosquitto.org");
```

Pour donner le numéro de port sans passer par spinBox :

```
m_client->setPort(1883);
```

La méthode `QmqttClient::messageReceived` retourne un `QString` nommé message contenant le payload mqtt.

Les méthodes `stateChanged` et `disconnected` renvoient vers les méthodes d'ouverture et de fermeture de connexion.

Le bouton `buttonConnect` appelle essentiellement les méthodes :

```
m_client->connectToHost();
et
m_client->disconnectFromHost();
```

qui établissent ou suppriment la liaison mqtt.

## PROTOCOLE MQTT – Réalisation d'un client avec Qt

Le bouton **buttonSubscribe** réalise la souscription à un topic :

```
auto subscription = m_client->subscribe(ui->lineEditTopic->text());
```

Le bouton **buttonPublish** publie le payload de **lineEditMessage** :

```
m_client->publish(ui->lineEditTopic->text(), ui->lineEditMessage->text().toUtf8())
```

ou

```
m_client->publish(nom_du_topic, payload).
```

La documentation de la bibliothèque qtmqtt est ici : <https://doc.qt.io/QtMQTT/index.html> .

## Exercice

À partir du node réalisé sur Raspberry Pi et de l'image ci-dessous :

- Réaliser un programme avec Qt affichant la température sur un QtLCD et disposant d'un bouton permettant d'allumer ou d'éteindre la LED.
- Pour cela recopier à nouveau l'exemple "simpleclient" dans un nouveau dossier de votre dossier de travail.
- Dans Qt creator, cliquer-droit sur la fenêtre d'édition puis "mettre en page -> casser la mise en page" ce qui vous permettra d'ajouter des widgets.

Ajouter `m_client->setClientId(ui->lineEditID->text())` ;

pour donner une identification sur mosquitto qui sera différente de celle sur le Rpi ou MQTT.fx.

