

Gewählte Technologie: MQTT

Schwerpunkt: Ressourcenbedarf

Titel: Vergleich des Ressourcenbedarfs von QMQTT und PAHO MQTT

David Tarnow, Patrick Olinger

Use-Case-Szenario:

Es wird pro MQTT-Client-Implementation eine Applikation erstellt, die jeweils 2 Publisher und 2 Subscriber beinhaltet. Diese tauschen über den MQTT-Broker Mosquitto Daten aus.

Die Größe der auszutauschenden Daten ist dabei variabel und wird in mehreren Stufen erhöht.

Dabei erfolgen Testdurchläufe mit einer Datengröße von 0, 1, 2, sowie 10^1 bis 10^7 Bytes.

Der Austausch der Daten erfolgt in fest eingestellten Intervallen von 10 bis 1000 Millisekunden.

Pro Testdurchlauf wird dabei sowohl der Hauptspeicher- als auch CPU-Bedarf der einzelnen Komponenten gemessen.

Schließlich wird ebenfalls der Ressourcenbedarf für längere Laufzeiten gemessen.

Hardware- und System-Setup:

Als MQTT-Broker wird Eclipse Mosquitto eingesetzt.

Für die MQTT-Clients ist jeweils eine Implementierung in QMQTT und PAHO MQTT vorgesehen.

Als Betriebssystem für die MQTT-Clients sowie den MQTT-Broker dient dabei Windows 10.

Die Konnektivität zwischen den MQTT-Clients und dem MQTT-Broker erfolgt über eine LAN-

Verbindung. Parallel zu den Clients läuft ein Programm, das in regelmäßigen Abständen die

Prozessor- und Speicherauslastung der einzelnen Clients ausliest und diese in einer Textdatei

abspeichert. Dabei wird ein Format gewählt, das sich in Microsoft Excel übertragen lässt. Dadurch

wird eine Visualisierung mittels eines Graphen ermöglicht. Das Intervall der Messungen ist dabei

variabel, sodass eine detaillierte Auswertung für kurze Laufzeiten und ein größeres Intervall für

lange Laufzeiten ermöglicht wird.

Zu erstellende Demo-Software:

Die Applikationen bestehen jeweils aus einem MQTT-Broker und vier MQTT-Clients.

Zwei der MQTT-Clients dienen als Publisher, die in voreingestellten Intervallen unterschiedliche Größen an Daten unter einem Topic veröffentlichen.

Die verbleibenden zwei MQTT-Clients dienen als Subscriber und lesen/empfangen die veröffentlichten Daten.

Zusätzlich wird ein Programm erstellt, das den CPU- und Speicherbedarf der Clients auslesen und abspeichern kann.

Ermittlung von Eigenschaften:

Welche Auswirkung haben verschiedene Größen an Daten auf den Ressourcenbedarf der beiden Implementierungen?

Welche Auswirkung hat die zeitliche Abfolge an Veröffentlichungen auf die Hauptspeicher- und CPU-Auslastung der beiden Implementierungen?

Wie verhält sich der Ressourcenbedarf bei längeren Laufzeiten der Client-Applikationen.

Evaluation:

1. Es werden Datensätze unterschiedlicher Größe gesendet/empfangen. In welchem Ausmaß beeinflusst die Datengröße die Performance und den Speicher der MQTT-Clients? Welche Unterschiede sind zwischen den beiden Implementationen feststellbar?
2. Es werden Datensätze in unterschiedlichen Intervallen veröffentlicht. Welche Auswirkung hat die zeitliche Abfolge der Daten auf den Ressourcenbedarf?
3. Es werden über einen langen Zeitraum Daten veröffentlicht. Welche Auswirkungen haben lange Laufzeiten auf die Performance bzw. den Speicherbedarf?

Präsentationsziele:

- Darstellung der Aufgabenstellung (1 Folie)
- Aufbau und Zusammenspiel der einzelnen Komponenten (1 Folie)
- Zu betrachtende Aspekte der jeweiligen Implementation (1 Folie)
- Abgrenzung von nicht zu betrachteten Aspekten (Unterschiedliche Anzahl Publisher/Subscriber, unterschiedliche Bandbreiten) (1 Folie)
- Darstellung der Messergebnisse (1 Folie)
- Evaluation der Technologien (1 Folie)
- Bewertung der beiden Implementationen (1 Folie)
- Live-Online-Demo