

Uebung3

Surendiran Sithamparam
Denis Pecekanovic
Regina Traber

3.4.2.3 Ein Sender, mehrere Empfänger (Point to Point Messaging)

Beobachten sie, wieviele Meldungen jeder Empfänger erhalten hat (Vergleichen sie die Meldungsnummern).

Das Starten der zwei Receiver ist ein manueller Prozess. Zuerst wird Receiver 1 gestartet und dann Receiver 2. Würden sie "gleichzeitig" gestartet, würden sie gleich viele Meldungen (+1) erhalten, einer die geraden und der Andere die Ungeraden. Bei 30 Meldungen wären es somit 15. Da der Receiver verzögert gestartet wurde, erhält er weniger Meldungen. Jede Meldung des Senders wird von nur einem Receiver empfangen/verarbeitet.

Receiver 1: gestartet Receiver 2: noch nicht gestartet	Receiver1 erhält alle Meldungen
Receiver 1: gestartet Receiver 2: gestartet	Die Receiver erhalten abwechselnd die Meldungen. Eine Meldung kann nicht von beiden Receivern erhalten werden.

3.4.2.4 Synchronisierter Meldungs Austausch (Client/Server Point to Point Messaging)

Beobachten sie, was beide Programme ausgeben und ziehen sie ihre Schlüsse.

Der SyncMsgQueueSender wartet auf eine Antwort von MsgQueueReplyer oder auf ein Timeout von 10 Sekunden, bevor die nächste Nachricht versendet wird. Hat der SyncMsgQueueSender eine Antwort erhalten, gibt er dies aus auf der Konsole aus.

SyncMsgQueueSender	MsgQueueReplyer
Sent a message "Message[1]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received a message of type AnswMsg[1] Sent a message "Message[2]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received a message of type AnswMsg[2] Sent a message "Message[3]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received a message of type AnswMsg[3] Sent a message "Message[4]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received a message of type AnswMsg[4] Sent a message "Message[5]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received a message of type AnswMsg[5] Sent a message "Message[6]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received a message of type AnswMsg[6] Sent a message "Message[7]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1	Received text message: Message[1] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [1] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[2] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [2] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[3] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [3] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[4] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [4] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[5] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [5] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[6] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [6] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[7] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [7] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[8] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [8] to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Received text message: Message[9] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1

Received a message of type AnswMsg[7] Sent a message "Message[8]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821- 1684570252547-1:1:1 Received a message of type AnswMsg[8] Sent a message "Message[9]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821- 1684570252547-1:1:1 Received a message of type AnswMsg[9] Sent a message "Message[10]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59821- 1684570252547-1:1:1 Received a message of type AnswMsg[10]	Has replied [9] to ID:DESKTOP-58KOUCD-59821- 1684570252547-1:1:1 Received text message: Message[10] will reply to ID:DESKTOP-58KOUCD-59821-1684570252547-1:1:1 Has replied [10] to ID:DESKTOP-58KOUCD-59821- 1684570252547-1:1:1
---	---

3.4.2.5 Synchronisierter Empfänger mit mehreren Sendern (Point to Point Messaging)

Beobachten sie darauf, welcher Sender welche Meldung (Temporäre Queue Nr) sendet, und ob der Empfänger auch dem richtigen Sender die Antwort zurück sendet.

Der Empfänger und jeder Sender nummerieren die Nachrichten für sich durch. Der Empfänger extrahiert aus jeder empfangenen Nachricht ein ReplyQueue-Objekt an welches er dann die Antwort versendet. Da jeder Sender eine eigene temporäre Queue instanziiert, über welche die Beantwortung der Nachricht abgewickelt wird, werden alle Antworten korrekt zugestellt. Da der Ablauf synchron abläuft, werden zuerst die Meldungen von Sender 1 beantwortet und dann die von Sender 2, der später gestartet wurde.

SyncMsgQueueSender 1	SyncMsgQueueSender 2	MsgQueueReplyer
Sender has the following Tmpqueue: ID:DESKTOP- 58KOUCD-59914- 1684570847506-1:1:1 Sent a message "Message[1]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Received a message of type AnswMsg[1] Sent a message "Message[2]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Received a message of type AnswMsg[2] Sent a message "Message[3]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Received a message of type AnswMsg[3] Sent a message "Message[4]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1	Sender has the following Tmpqueue: ID:DESKTOP- 58KOUCD-59919- 1684570849352-1:1:1 Sent a message "Message[1]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919- 1684570849352-1:1:1 Received a message of type AnswMsg[8] Sent a message "Message[2]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919- 1684570849352-1:1:1 Received a message of type AnswMsg[10] Sent a message "Message[3]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919- 1684570849352-1:1:1 Received a message of type AnswMsg[12] Sent a message "Message[4]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919- 1684570849352-1:1:1	Received text message: Message[1] will reply to ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Has replied [1] to ID:DESKTOP- 58KOUCD-59914-1684570847506- 1:1:1 Received text message: Message[2] will reply to ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Has replied [2] to ID:DESKTOP- 58KOUCD-59914-1684570847506- 1:1:1 Received text message: Message[3] will reply to ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Has replied [3] to ID:DESKTOP- 58KOUCD-59914-1684570847506- 1:1:1 Received text message: Message[4] will reply to ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Has replied [4] to ID:DESKTOP- 58KOUCD-59914-1684570847506- 1:1:1 Received text message: Message[5] will reply to ID:DESKTOP-58KOUCD-59914- 1684570847506-1:1:1 Has replied [5] to ID:DESKTOP- 58KOUCD-59914-1684570847506- 1:1:1

<p>Received a message of type AnswMsg[4] Sent a message "Message[5]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Received a message of type AnswMsg[5] Sent a message "Message[6]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Received a message of type AnswMsg[6] Sent a message "Message[7]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914-168Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Received a message of type AnswMsg[13]</p>	<p>Received a message of type AnswMsg[14] Sent a message "Message[5]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Received a message of type AnswMsg[15] Sent a message "Message[6]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Received a message of type AnswMsg[16] Sent a message "Message[7]" to queue "Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919-168Q.CDS.regina.traber.Request" and await answer on ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Received a message of type AnswMsg[20]</p>	<p>Received text message: Message[6] will reply to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Has replied [6] to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Received text message: Message[7] will reply to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Has replied [7] to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Received text message: Message[1] will reply to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Has replied [8] to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Received text message: Message[8] will reply to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Has replied [9] to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Received text message: Message[2] will reply to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Has replied [10] to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Received text message: Message[9] will reply to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Has replied [11] to ID:DESKTOP-58KOUCD-59914-1684570847506-1:1:1 Received text message: Message[3] will reply to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Has replied [12] to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 SKTOP-58KOUCD-59919-1684570849352-1:1:1 Received text message: Message[10] will reply to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1 Has replied [20] to ID:DESKTOP-58KOUCD-59919-1684570849352-1:1:1</p>
---	--	--

3.4.3.1 Einfacher Publisher mit Subscriber (1:1 Publish/Subscribe)

Beobachten sie, ab welcher Meldungsnummer der Consumer die Meldungen erhält.

Der Subscriber erhält nur diejenigen Meldungen, die von Publisher nach dem Zeitpunkt versendet werden, ab dem sich der Subscriber am Topic des Publishers registriert hat.

In diesem Publisher/Subscriber Modell gibt es keinen Meldungs-Queue, welche die Messages für den Subscriber zwischenspeichert, und keine Sender/Empfänger Synchronisation. Meldungen, welche vom Subscriber verpasst wurden, sind für ihn verloren.

3.4.3.2 Einfacher Publisher mit mehreren Subscribern (1:n Publish/Subscribe)

Beobachten sie, ab welcher Meldungsnummer die Consumer die Meldungen erhalten

Die Consumer enthalten die Meldungen, nachdem sie gestartet wurden und subscribed haben. Alle Consumer enthalten dann alle Meldungen. z.B erhalten alle die Message[9], Message[10]...

Consumer 1	Consumer 2	Consumer 3
Received text message: Message[1]	Received text message: Message[9]	Received text message: Message[16]
Received text message: Message[2]	Received text message: Message[10]	Received text message: Message[17]
Received text message: Message[3]	Received text message: Message[11]	Received text message: Message[18]
Received text message: Message[4]	Received text message: Message[12]	Received text message: Message[19]
Received text message: Message[5]	Received text message: Message[13]	Received text message: Message[20]
Received text message: Message[6]	Received text message: Message[14]	...
Received text message: Message[7]	Received text message: Message[15]	
Received text message: Message[8]	Received text message: Message[16]	
Received text message: Message[9]	Received text message: Message[17]	
Received text message: Message[10]	Received text message: Message[18]	
Received text message: Message[11]	Received text message: Message[19]	
Received text message: Message[12]	Received text message: Message[20]	
Received text message: Message[13]	..	
Received text message: Message[14]		
Received text message: Message[15]		
Received text message: Message[16]		
Received text message: Message[17]		
Received text message: Message[18]		
Received text message: Message[19]		
Received text message: Message[20]		
..		

3.4.3.3 Mehrere Publisher mit einem Subscriber (n:1 Publish/Subscribe)

Wie 3.4.3.1 aber Publisher mehrmals gestartet und beobachten sie den Subscriber.

Der Consumer bekommt ab dem Zeitpunkt, nach dem er gestartet und sich subscribed hat, alle Meldungen aller Publisher. Die Reihenfolge innerhalb der Publisher (Message [1], ..., Message [15]) stimmt (durch die Queue), die Anordnung Producer 1, Producer 2 muss aber nicht iterierend sein.

3.4.4.1 Schlussfolgerung und Zusammenfassung

Erstellen sie eine kurze Zusammenfassung aller oben markierten Beobachtungen in einem PDF-Dokument und schreiben sie auf, was sie daraus schliessen.

Beobachtung Zusammenfassung:

Siehe Aufgaben vorher

Point to Point (Producer/Consumer)

- Durch die Queue wird jeweils sichergestellt, dass Nachrichten in derselben Reihenfolge empfangen werden, wie sie gesendet worden sind.
- Gibt es mehrere Empfänger und keine Synchronisation, ist nicht vorhersehbar, welcher Empfänger welche Nachricht empfängt
- Durch die Queue können die Nachrichten auch zeitlich versetzt empfangen werden
- Durch eine Synchronisation kann eine bessere Kontrolle über den Empfang der Meldungen erzielt werden (wer empfängt was/wann)

Publish/Subscribe

- Der Subscriber empfängt alle Nachrichten aller Publisher, sofern er bei den Publisher eingetragen ist
- Der Subscriber empfängt die Nachrichten, sobald er sich registriert hat. Nachrichten, welche durch den Publisher vorgängig gesendet wurden, sind verloren bzw. der Subscriber empfängt diese nicht.
- Ein Subscriber kann auch von mehreren Publisher Nachrichten empfangen. Das Eintreffen der Nachricht bzw. deren Abfolge ist nicht deterministisch.

Allg. Zusammenfassung:

Die Point to Point-Kommunikation ist eine direkte Verbindung zwischen einem Sender und einem Empfänger, während das Publish/Subscribe-Modell eine lose Kopplung ermöglicht, bei der Publisher Nachrichten an Themen oder Kategorien senden und Subscriber die Nachrichten empfangen, für die sie sich interessieren.

Schlussfolgerungen:

Die Point to Point-Kommunikation und das Publish/Subscribe-Modell sind zwei verschiedene Ansätze für die Datenübertragung in einem Netzwerk, die je nach Anforderung gewählt werden.

Schreiben sie je drei praktische Beispiele aus Ihrem Berufsumfeld auf, bei denen sie die eine oder andere. (PTP oder P/S) Topologie einsetzen könnten.

Berufsumfeld: Business Software Applikationen

Point-to-Point Messaging

- E-Mail-Kommunikation: Senden von E-Mailbenachrichtigungen bei bestimmten Aktionen der Anwendung.
- Direkte API-Kommunikation: Benutzen von REST APIs um Frontend-Client Daten von einem Backend-Server abzurufen, um sie auf einer Webseite anzuzeigen.
- WebSockets-Kommunikation: Chat-Anwendung mit WebSockets, um Sofortnachrichten zwischen Benutzern auszutauschen
- Verwendung von Webservices
- Benutzung von Message Queues

Publish/Subscribe

- Publisher-Subscriber Kommunikation: Benachrichtigungssystem wo sich interessierte Personen für Benachrichtigungen subscriben können
- Logging und Protokollierung: Das Logging-System agiert als Publisher und veröffentlicht Log-Einträge, während andere Komponenten oder Systeme sich als Subscriber anmelden können, um diese Log-Einträge zu empfangen und entsprechende Aktionen auszuführen.
- Echtzeit-Analysen: Ein Analyse-System fungiert als Publisher und sendet Datenströme über Benutzerinteraktionen, die diese Daten verwenden, um Echtzeit-Einblicke und Dashboards zu generieren.
- IoT-Datenverarbeitung: Sensordaten können an interessierte Subscriber gesendet werden, die sie verarbeiten und analysieren.