

ENTWURF

ENTWURF ZUR AUFGABE 3 AUS DER VORLESUNGSREIHE
“VERTEILTE SYSTEME”

TEAM 1: MICHAEL STRUTZKE, IGOR ARKHIPOV

Aufgabenaufteilung

Alle Aufgaben wurden gemeinsam entworfen und bearbeitet.

Quellenangaben

Aufgabenstellung, Vorstellung der Aufgabe in der Vorlesung.

Bearbeitungszeitraum

Datum	Dauer in Stunden
20.11.16	5
22.11.16	3
24.11.16	5
Gesamt	13

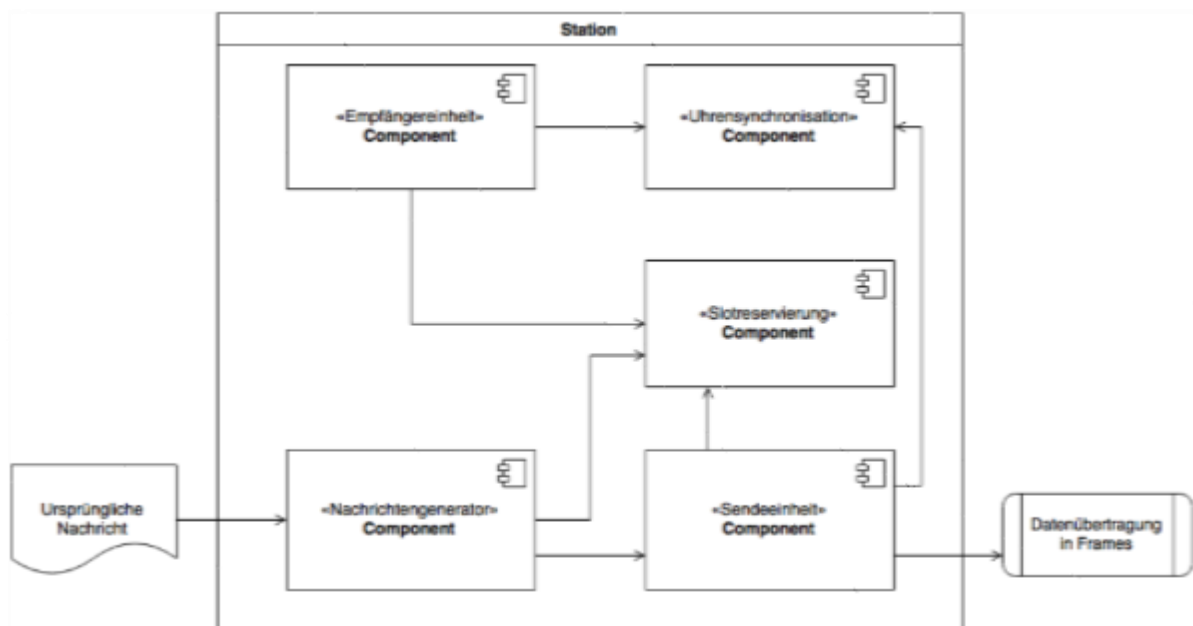
Aktueller Stand

Der Entwurf ist fertig. Die Implementation läuft gerade.

Änderungen im Entwurf

Noch keine.

Details zum Entwurf:



Es soll ein Simulator für das Zeitmultiplexverfahren (STDMA) System implementiert werden. Insgesamt werden bis zu 25 Stationen unterstützt, die über Multicast ihre 34 Byte-lang Nachrichtenpakete austauschen und diese in festen Zeitslots einordnen. Zu sendende Daten kommen von der vorgegebenen Datenquelle (als „ursprüngliche“ Nachricht markiert). Die empfangenen Daten werden in der Konsole und einer Logdatei rausgegeben.

Die Koordination der Stationen findet dezentralisiert statt. Daher enthalten alle Nachrichten eine Zeitslotnummer, die der Station einen einzigen Zeitslot zuordnet, um Kollisionen auf dem gemeinsamen Übertragungskanal zu vermeiden. Gleichzeitig werden die Ressourcen des Kanals besser ausgenutzt.

Damit alle Stationen den Kanal gleich nebenläufig verwenden können, werden alle von denen über eine Sendezeit als Zeitstempel synchronisiert.

Gemäß Vorgabe werden alle Kanalmitglieder zwischen 2 Uhrensynchronisationsklassen A und B unterschieden. Die Stationen der Klasse A haben die Uhren, die als hinreichend genau gelten, und die der Typ B müssen entsprechend mit der Klasse A synchronisiert werden.

Komponentenübersicht

Jede Station ist eine abgeschlossene Standalone-Anwendung und bietet keine externen Schnittstellen für andere Prozesse an. Das Programm, das eine solche Station repräsentiert, ist in fünf Prozesse aufgeteilt, die ihre eigenen Verantwortlichkeiten haben. Es handelt sich hierbei um die folgenden logischen Blöcke: Empfänger, Uhrensynchronisation, Slotreservierung, Nachrichtengenerator und eine Sendekomponente.

Der Empfängerprozess dient dem Empfang jeglicher Nachrichten und überprüft, ob diese eine Kollision aufweisen (mittels der Sendezeit). Dafür sammelt er die Nachrichten für jeden Slot und bewertet deren Anzahl. Ist tatsächlich mehr als eine Nachricht am Ende eines Slots eingegangen, handelt es sich somit um eine Kollision. In diesem Fall müssen diese überschneidenden Nachrichten verworfen werden. Ansonsten wird die Nachricht an die Slotreservierung (bzw. an die Uhrensynchronisation) weitergeleitet. Dort wird ein Eintrag für die reservierten Slots vorgenommen, bzw. eigene Programmzeit synchronisiert.

Die Uhrensynchronisation bekommt die Nachrichten von dem Empfänger und passt einen eigenen Zeitzustand am Ende des Frames an. Zunächst werden die eingehenden Nachrichten bezüglich ihrer Stationsklasse unterschieden. Die Sendezeiten der Stationen vom Typ A bilden am Ende des Frames mit Hilfe des arithmetisches Mittels eine neue verrechnete Zeit. Die Uhrensynchronisation liefert auch die aktuelle Zeit zurück, falls die anderen Komponenten danach fragen.

Die Slotreservierung kümmert sich um die Handhabung und Buchung des Slots innerhalb des nächsten Zeitframes. Ihre wichtigste Aufgabe ist die Slotvergabe, die nach der Ermittlung von noch freien Slots passiert. Bei jeder empfangenen Nachricht wird die Liste der freien Slots aktualisiert. Immer wenn ein Frame vorbei ist (mit Hilfe von Timer gemessen), wird die Slotreservierung darüber informiert und von der Uhrensynchronisation nach der aktuellen Zeit gefragt. Darüber hinaus wird ein Timer bei jedem Slot hinterlegt. Nach jedem seinen Ablauf wird der Empfänger entsprechend benachrichtigt und der Timer wieder gesetzt.

Im Falle der erfolgreichen Slotreservierung wird dieser Slot und die aktuelle Zeit der Sendekomponente übergeben. Wurde kein Slot reserviert, wird ein Slot zufälligerweise aus dem letzten Frame gewählt und gespeichert. Im nächsten Frame wird dann darauf geachtet, dass ein zufälliger Slot wirklich frei ist.

Die Sendekomponente nimmt die generierte Nachricht und die zusätzliche Information nach dem Timerereignis (ein Frame ist vorbei) von dem Nachrichtengenerator. Zuerst muss hier noch eine Prüfung des aktuellen Sendezeitpunktes geschehen. Wenn die Zeit passt, wird das Packet mit der Sendezeit versehen und per Multicast verschickt. Sonst wird kein Versand in dem aktuellen Frame für diese Station passieren. Die Nachricht muss folgenderweise vorbereitet werden:

- Byte 0: Stationsklasse (,A' oder ,B') ist bei der Initialisierung gesetzt und verändert sich nicht im Laufe der Zeit;
- Byte 1-24: Nutzdaten aus dem Puffer, die die Datenquelle liefert.
- Byte 25: reservierte Slotnummer für den nächsten Frame.
- Byte 26-33: Zeitpunkt, zu dem die Nachricht gesendet wurde.

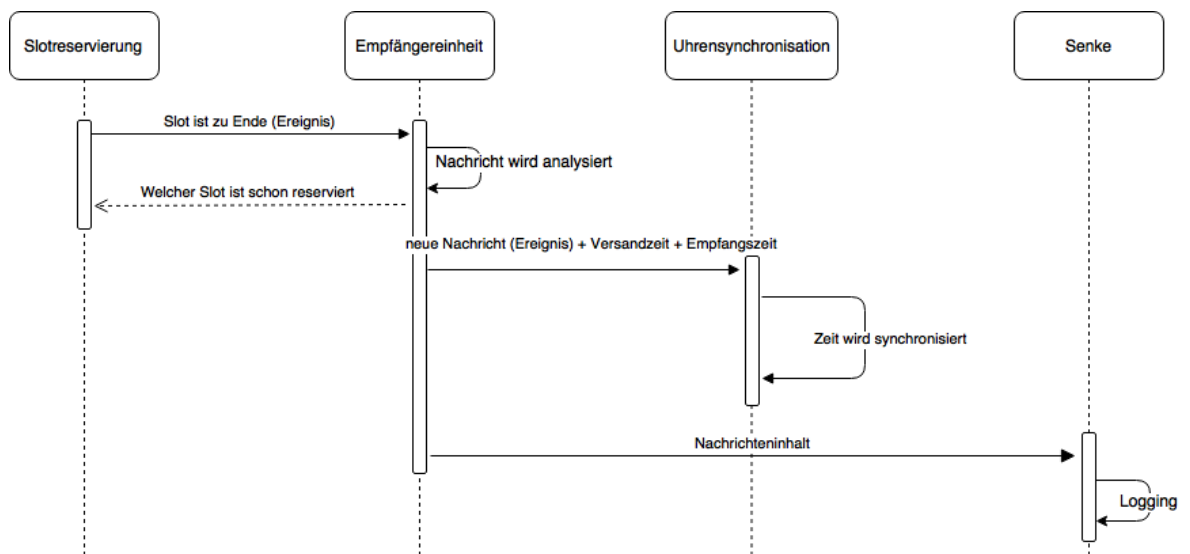
Um die eingehenden Nachrichten weiter analysieren zu können, gibt es noch eine zusätzliche Komponente Senke, die die Nachrichten an Empfänger in einer Log-Datei speichert.

Phasenübersicht

Das System geht die folgenden Phasen durch: die Initialisierungsphase, die Empfangsphase und die Versandphase.

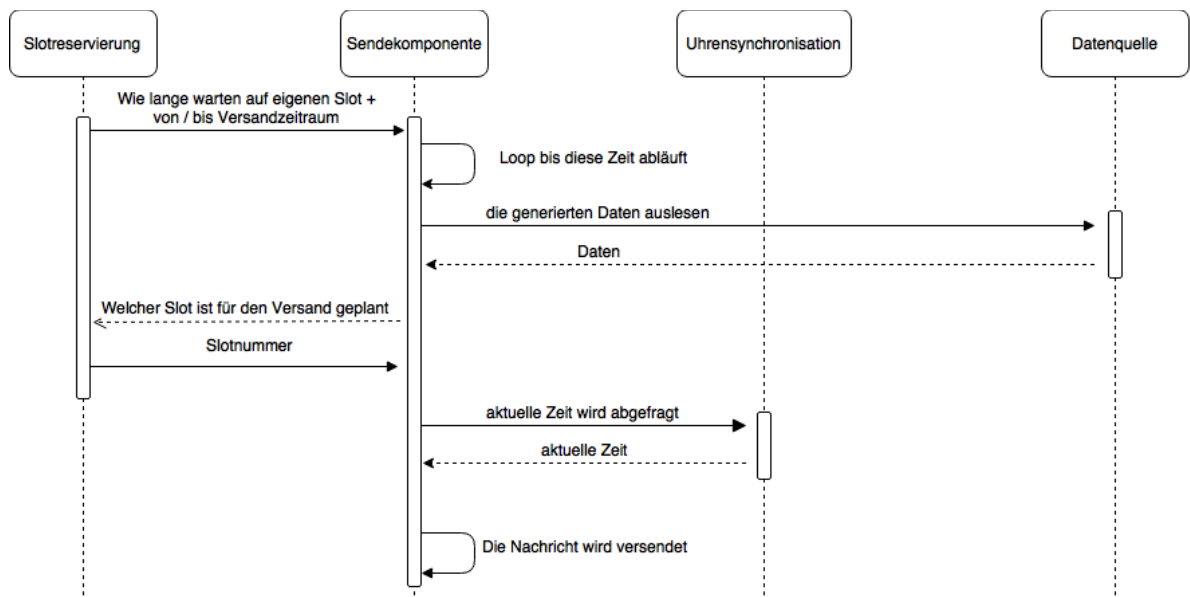
Die Initialisierungsphase findet dann statt, wenn eine Station gestartet wird, und bereitet die notwendigen Threads für jeweiligen Komponenten.

Die Empfangsphase wird in Form von Sequenzdiagramm dargestellt:



Die Slotreservierung Komponente informiert die Empfängereinheit, dass ein Slot schon zu Ende ist. Die Empfängereinheit zählt die Nachrichten, die innerhalb dieses Slots angekommen waren. War es die einzige Nachricht, wird die analysiert und der für diese Nachricht gebuchte Slot wird an Slotreservierung gesendet. Das ermöglicht die Vergabe des freien Slots für den nächsten Frame. Weiter wird die Nachricht an Uhrensynchronisation (mit aktueller Zeit und Empfangsstempel) und Senke übergeben.

Die Versandphase wird in Form von zweitem Sequenzdiagramm dargestellt:



Zuerst schickt die Slotreservierung Komponente das Zeitintervall, währenddessen die Sendekomponente noch warten muss. Dazu gehört auch die Zeitgrenze, innerhalb deren der Versand geschehen muss. Wenn die Sendekomponente danach aufwacht, baut sie eine eigene Nachricht für den Versand auf. Die Komponente sendet Requests an Datenquelle und auch Slotreservierung, damit die Rohdaten und Slotnummer von denen für den nächsten Frame gesammelt werden. Als nächster Schritt wird die Zeit von der Uhrensynchronisation abgefragt, weil wir genau wissen müssen, ob der geplante Slot gleich gestartet wird. Der Zeitstempel wird dann der Nachricht beigelegt und die komplette Nachricht wird als multicast gesendet. Wenn der Slot aus irgendwelchen Gründen vorbei ist, wird die konstruierte Nachricht nicht gesendet.