Methodik für den Einsatz asynchroner Kommunikation beim Entwurf von verteilten, objektorientierten Systemen unter Echtzeitbedingungen

Dissertationsvorschlag Marc Schanne – Kurzfassung (25.8.2005)

1. Motivation und Problembeschreibung

Bei sicherheits- oder geschäftskritischen Systemen wächst die Notwendigkeit zur Verteilung der Verarbeitung auf mehrere Knoten. Durch Ausnutzung der Rechenleistung moderner Sensoren und Aktoren wird es möglich Daten näher an der Quelle bzw. der Senke zu verarbeiten und so unterschiedliche Vorteile, wie sie in der Anforderungsanalyse [Sch05a] beschrieben sind, beim Systemdesign zu nutzen. Die Entwicklung komplexerer, besser fehlerresistenter und skalierbarer Applikationen wird möglich. Durch objektorientierte Techniken möchte man diese Vorteile mit besserer Wiederverwendbarkeit von Komponenten, objektorientiertem Design und einfacherer Wartbarkeit verbinden.

Die Garantie von Echtzeitbedingungen bei der notwendigen Kommunikation erfordert den Einsatz von echtzeitfähigen Netzwerken. Der hier vorgeschlagene Ansatz einer Nachrichtenkommunikation ist mit Hinblick auf existierende Netzwerke und Feldbussysteme im Umfeld eingebetteter Systeme entwickelt. Das vorgesehene Softwareentwurfsmuster [Sch05b] ermöglicht eine asynchronen, direkte Publiziere/Abonniere-Nachrichtenkommunikation auf Basis von 100% reinem Java mit der Echtzeiterweiterung RTSJ. Durch minimalen Protokoll-Overhead sollen die Leistungscharakeristika des zugrundeliegenden physikalischen Netzwerkes voll genutzt werden.

Neben einer Rahmenarchitektur, für die Unterstützung der Nachrichtenkommunikation auf einem für Viele-zu-viele-Kommunikation orientierten Netzwerkprotokoll, definiert die vorgestellte Methodik Anforderungen an das Laufzeitsystem mit mehreren echtzeitfähigen Kontrollfäden und einer Ablaufkoordinierung für feste Prioritäten. Durch eine Systembeschreibung mit Programmierkonstrukten der 5. Generation, mit Angabe von Systemvoraussetzungen und Anwendungszielen, definiert die Dissertation außerdem eine Entwicklungsmethode, mit der die Entwicklung zuverlässiger, verteilter und sicherheits- oder geschäftskritischer Systeme vereinfacht werden soll.

2. Lösungsskizze: Entwurfsmuster und Softwareentwicklungsprozess

Die in [Sch05b] vorgeschlagene Rahmenarchitektur mit Empfänger-Kollektiv, einem Kontrollfaden zur Steuerung von Verarbeitungskontrollfäden und Warteschlangen ermöglicht asynchrone Kommunikation mit Nachrichten auf Basis von synchronen Ein-/Ausgabeschnittstellen auf einem Ein-Prozessorsystem in harten Zeitgrenzen. Die notwendige Struktur mit den wesentlichen Komponenten sowie deren Interaktion wird hier nur grob dargestellt. Durch Anforderungen der verwendeten Methodik an die Laufzeitumgebung wird die Integration des Nachrichtendienstes in die Ablaufkoordinierung bestimmt und eine vereinfachte Softwareentwicklungsmethode basierend auf einer Beschreibung mit XML definiert.

2.1. Struktur und Interaktion

Die in [Sch05b] vorgestellte zentrale Komponente des Nachrichtendienstes ist ein logisches Objekt für den Nachrichtenkanal. Der Nachrichtenkanal definiert Attribute für Nachrichten und die verarbeitenden Einheiten auf Sender-, Empfänger- oder Vermittlerseite (vgl. Abbildung 1). Diese Informationen müssen bei allen beteiligten Kommunikationspartnern gleich zur Verfügung stehen. Für den Versand und Empfang von Nachrichten, die Nachrichtenkanälen zugeordnet sind, über das physikalische Kommunikationsnetzwerk verwendet jeder Knoten einen einfachen Byte-Array orienterten Zugangssockel. Hier wird die Abbildung von Nachrichten und Nachrichtenkanälen auf einfach zu übertragende Bytes und eine Zuordnung zu den Attributen der Nachrichtenkanäle garantiert.

Bei der statischen Version des Nachrichtendienstes für harte Echtzeitbedingungen kann dies mittels einer Konfiguratonsdatei mit allen notwendigen Beschreibungen erfolgen [Sch05c]. Diese Beschreibung kann mit Systemvoraussetzungen und Anwendungszielen verbunden werden und eine architekturneutrale Beschreibung des verteilten Systems

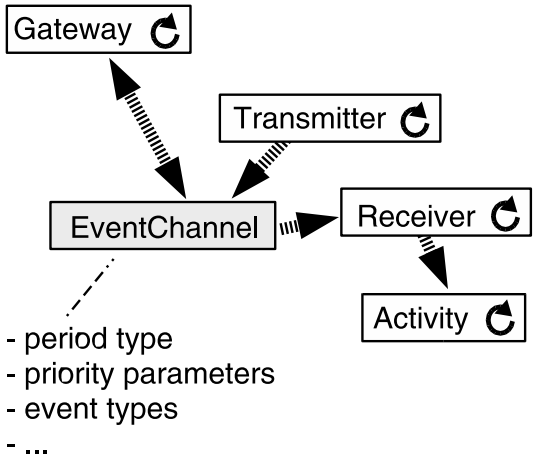

Abbildung 1. Nachrichtenkanal und zugeordnete verarbeitende Einheiten

bieten. Die flexible Version bei weichen oder gemischten Echtzeitanforderungen definiert einen gesonderten allen Partnern bekannten Nachrichtenkanal, über den notwendige Verhandlungen und Benachrichtigungen stattfinden können [SH04].

Jeder Knoten verfügt pro Zugangssockel über einen oder mehrere Kontrollfäden mit höchster Priorität, die periodisch die asynchron eingetroffenen und im Netzwerkzugangspunkt gepufferten Nachrichten in Empfang nehmen und in ein Warteschlangensystem zur Weiterverarbeitung einreihen. Systeme mit harten Echtzeitbedingungen müssen garantieren, dass für alle eintreffenden Nachrichten ein Empfangskontrollfaden die Puffer der Eingangsschnittstelle lesen und Nachrichten an das Verarbeitungssystem weiterreichen kann. Dies muss zeitnah, bevor die Daten durch neue überschrieben würden, geschehen und kann ebenfalls durch Hardwarecharakeristika in der Systembeschreibung definiert sein.

Nachrichten in den Warteschlangen werden von einem Managerkontrollfaden an wartende Kontrollfäden mit passenden Prioritäten und Systemattributen weitergereicht und so die Behandlung in vorgegebenen Zeitgrenzen ermöglicht.

Die Abbildung 2 zeigt die hier beschriebene Interaktion beim Empfang von Nachrichten über mehrere physikalische Kommunikationsnetzwerke. Der Versand von Nachrichten wird nicht weiter betrachtet, weil neben der Definition von Veröffentlichungsattributen bei Nachrichtenkanälen lediglich eine entsprechende Bereitstellung über die Netzzugangsschnittstelle notwendig ist.

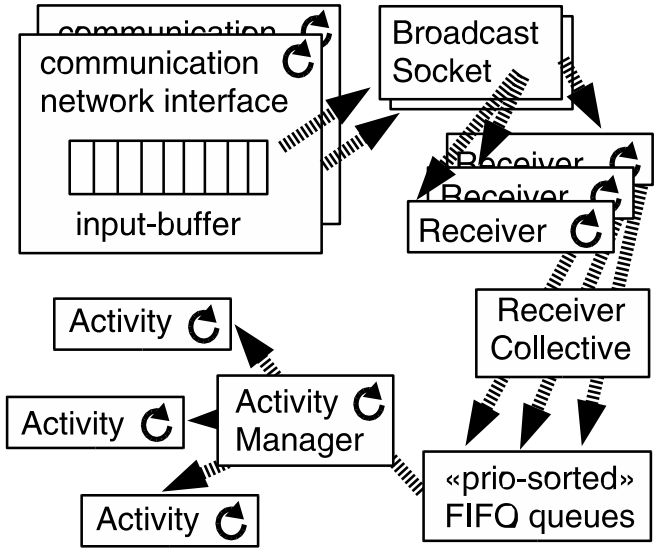

Abbildung 2. Struktur und Interaktion beim Nachrichtenempfang

2.2. Ablaufkoordinierung

Der Nachrichtendienst bietet eine asynchrone, themenbasierte Publiziere/Abonniere-Kommunikation und unterstützt ein direktes "Push"-Nachrichtenübertragungsmodell ohne zentrale Diensteinheiten. Weitere Anforderungen und Informationen zu Kommunikation und Synchronisation bietet [Sch05a]. Die mit einem Nachrichtenkanal verbundenen Nachrichten können in periodischer oder sporadischer[1] Häufigkeit eintreffen und mit dieser Einschränkung wird eine Abbildung auf die in RTSJ verfügbare periodischen und sporadischen Kontrollfäden ermöglicht. Die zugeordneten Empfangskontrollfäden sind für den direkten Empfang oder die zeitnahe Leerung des Empfangspuffers der Netzwerkzugangsschnittstelle und die Einreihung der Nachrichten in ein nach Prioritäten sortiertes Warteschlangensystem verantwortlich.

Durch die Nutzung einer prioritätsbasierten Ablaufkoordinierung kann ausgehend von diesen Anforderungen auch für Systeme mit harten Echtzeitbedingungen statisch die Erfüllung von Zeitanforderungen garantiert werden. Die für jedes physikalische Netzwerk existieren periodisch und kurzzeitig aktiven Empfängerkontrollfäden nutzen die Attribute der Nachrichtenkanäle für die Festlegung ihrer Ausführungsparameter und ermöglichen die Verabeitung periodischer Nachrichten. Bei sporadischem Auftreten der Nachrichten wird das Verfahren von "sporadic servers" ein-

[1] mit garantierten Empfangszwischenzeiten

gesetzt, das zusammen mit den periodischen Kontrollfäden eine statische Analyse für die gemeinsame Ablaufkoordinierung ermöglicht [Sch05c].

2.3. Deskriptive Softwareentwicklungsmethode

Die statische Beschreibung der Nachrichtenkanäle und Kommunikationsknoten unter harten Echtzeitbedingungen erlaubt neben der statischen Analyse der Ablaufkoordinierung auch die Nutzung der Programmierkonzepte von Sprachen der 5. Generation. Mit der Beschreibung von Systemvoraussetzungen, z.B. der Pufferkapazität in Netzwerkszugangsschnittstellen oder Prioritäten für die Verarbeitung von Nachrichten und Anwendungszielen, z.B. bei der Verarbeitung selbst, wird eine deskriptive Softwareentwicklung ermöglicht. Erste Beispiele dazu finden sich in [Sch05a] oder [Sch05c].

Für den Einsatz unter flexiblen Echtbedingungen schlägt [SH04] auch eine auf XML und XSLT basierende Generierung des Programmcodes zur Nutzung der Nachrichtendienst-Infrastruktur vor.

3. Verwandte Arbeiten

Verwandte Arbeiten für die Entwicklung verteilter, sicherheits- oder geschäftskritischer Systeme mit echtzeitfähiger Kommunikation sind von Seiten verteilter und plattformübergreifenden Programmiersysteme wie CORBA, aber auch beim Hardware-nahen Entwurf von Mikrokernbasierten Echtzeitbetriebssystemen zu identifizieren. Die Anforderungsanalyse [Sch05a] beschreibt und bewertet beide Ansätze insbesondere unter Berücksichtigung der gegebenen Netzwerkinfrastruktur bei eingebetteten Systemen und einer objektorientierten Entwicklungsmethode.

3.1. RT-CORBA

Die Object Management Group (OMG) hat ein Objekte/Dienste-Informationsmodell für heterogene Applikationen in verschiedenen Sprachen und auf unterschiedlichen Plattformen entwickelt. Um die “Common Object Request Broker Architecture” (CORBA) für Anforderungen unter Echtzeitbedingungen anzupassen, ist deshalb die Echtzeiterweiterung (RT-CORBA) definiert worden. RT-CORBA beschreibt auf Basis der synchronen CORBA-Kommunikationsplattform auch asynchrone Nachrichteninteraktion. Neben dem relativ hohen Protokoll-Overhead bei dieser Kommunikation ist aber besonders die TCP/IP-Orientierung der CORBA-Plattform ein Hindernis für die Nutzung im Umfeld sicherheits- oder geschäftskritischer Systeme, die oft Netzwerke mit Viele-zu-viele-Kommunikation verwenden [Sch05a].

3.2. OSA+

Die Integrationsplattform “Open System Architecture - Platform for Universal Services” (OSA+) ist von den Anforderungen der eingebetteten Hardware nach Stromsparsamkeit und effizienter Ausführung bestimmt. Im Gegensatz zu CORBA wird hier ein Mikrokern-Betriebsystem definiert und die Integration in objektorientierte Systeme erfordert erst noch die Implementierung einer entsprechenden Schnittstelle auf Basis der vorgegebenen elementaren Mikrokern-Funktionen [Sch05a].

4. Einordnung, Bewertung

Zur Bewertung der vorgestellten Methodik und Untersuchung der eingeführten Methoden werden in der Dissertation auf unterschiedlichen Ebenen Metriken und Vergleichskriterien definiert. Da das vorgestellte Verfahren asynchroner Nachrichten-Kommunikation sowohl unter harten, als auch flexiblen Echtzeitbedingungen Vorteile bietet, ist es Ziel, beide Versionen in realistischen Systemumgebungen auf ihre Einsatzmöglichkeit zu testen.

4.1. Statische Version unter harten Echtzeitbedingungen

Harte Echtzeitbedingungen verlangen statische Analyse um die Ablaufkoordinierung vor der Ausführung zu verifizieren. Für die Kommunikation mit Nachrichten bedeutet dies eine Deklaration aller notwendigen Systemattribute im voraus und diese Anforderung erlaubt die Nutzung einer deskriptiven Entwicklungsmethode in XML.

Für die Verifikation der Entwicklungsmethode und der Nachrichtenkommunikation soll neben einem Prototypen für ein sicherheits- oder geschäftskritisches Steuerungs- und Kontrollsystem insbesondere die statische Analyse der Ablaufkoordinierung untersucht werden.

4.2. Dynamische Version mit flexiblen Echtzeitbedingungen

Bei der Nutzung unter flexiblen Echtzeitbedingungen wird eine deterministische Behandlung möglicher Fehlerquellen zur Laufzeit interessant. In [SH04] wird basierend

auf unterschiedlichen möglichen physikalischen Kommunikationsnetzwerken ein Fehlermodell vorgestellt und zwei Verfahren der Fehlerbehandlung für periodische und sporadische Nachrichten vorgeschlagen. Der dynamische Charakter wird durch Nutzung eines Verwaltungsnachrichtenkanals, auf dem die Erzeugung, Suche und Bewerbung neuer Kommunikationskanäle möglich wird, ermöglicht.

Für die Nutzung im HIJA Forschungsprojekt soll auf Basis des vorgestellten Publiziere/Abonniere-Nachrichtendienstes eine Gleiche-zu-gleiche-Kommunikationsinfrastruktur (Pastry) implementiert und genutzt werden. Auch die Nutzung mit "Controller Area Network" (CAN) Netzwerken im Automobil wird untersucht. Aus beiden Testszenarien sollen realistische Bewertungskriterien und Erfahrungen bei der Nutzung in die Dissertation integriert werden.

5. Ausblick und Zusammenfassung

Durch die Integration in das europäische Forschungsprojekt versucht die vorgestellte Dissertation realistische Einsatzsszenarien als Testumgebung zu nutzen. Die Verifikation der Vorteile einer asynchronen nachrichtenbasierten Publiziere/Abonniere-Kommunikation für aktuelle und zukünftige eingebettete Systeme im Umfeld sicherheits- und geschäftskritischer verteilter Anwendungen ist Ziel der präsentierten Methodik. Hiermit und mit Integration einer Softwareentwicklungsmethode, die Programmiersprachenkonzepten der 5. Generation ausnutzt, soll die Entwicklung dieser Systeme vereinfacht und verbessert werden.

Literatur

[Sch05a] Marc Schanne. Anforderungsanalyse für asynchrone Kommunikation beim Entwurf von objektorientierten, verteilten Systemen unter Echtzeitbedingungen. 2005.

[Sch05b] Marc Schanne. Real-Time Communication with a Receiver Collective, Activity Manager, and Queues. In *Proceedings of IADIS Internatonal Conference Applied Computing 2005*, 2005.

[Sch05c] Marc Schanne. Real-Time Communication with Direct Publish/Subscribe Event Service. 2005.

[SH04] Marc Schanne and Dr. James J. Hunt. Remote Event Service Design. Technical report, FZI Forschungszentrum Informatik, 2004. Deliverable D4.2 describing the HIDOORS event channel network.