

Implementierung und Analyse eines effizienten Benachrichtigungsdienstes für zusammengesetzte Ereignisse

Sven Bittner, Steven König
{bittner, koenig}@inf.fu-berlin.de
Arbeitsgruppe Datenbanken und Informationssysteme
Institut für Informatik, Freie Universität Berlin
Betreuung: Prof. Dr. H. F. Schweppe, Dipl. math. tech. A. Hinze

In den letzten Jahren erhalten *Benachrichtigungsdienste* immer mehr Aufmerksamkeit, z.B. in der Gebäudesteuerung oder der Verkehrskontrolle. Diese Systeme informieren ihre *Abonnenten* beim Auftreten von bestimmten *Ereignissen*, in der Gebäudesteuerung beispielsweise Änderungen von Sonneneinfallswinkeln bei Sonnenblenden. Die Ereignisse werden dem System von *Anbietern*, beispielsweise Lichtsensoren einer Jalousie, zur Verfügung gestellt. Die Darstellung der Ereignisse erfolgt durch *Attribut-Wert-Paare*. Die Ereignisstruktur beschreibt der eindeutig zugeordnete *Ereignistyp*. Bevor ein Anbieter Ereignisse an das System sendet, gibt er deren Art mit Hilfe von *Quellenprofilen* bekannt. Das System kann dadurch die potentiell möglichen Ereignisse eines Anbieters bestimmen. Die Abonnenten eines Benachrichtigungssystems sind an einer Teilmenge der Ereignisse interessiert, diese beschreiben sie in Form von *Profilen*. Das Finden aller passenden Profile zu einem Ereignis wird als *Filterung* bezeichnet. Die Auslieferung dieses Ereignisses an die jeweiligen Abonnenten als *Benachrichtigung*.

Profile können *einfache* oder *zusammengesetzte Ereignisse* beschreiben und werden demnach *einfache* bzw. *zusammengesetzte Profile* genannt. Einfache Profile werden durch ihren Typ und eine Menge an *Prädikaten* für die Attribute dieses Typs beschrieben. Werten alle Prädikate eines Profils auf einem Ereignis mit *Wahr* aus, so *passt* das Ereignis zum Profil und der Abonnent wird benachrichtigt. Zusammengesetzte Profile beschreiben eine temporale Komposition mehrerer Ereignissen, z.B. starke Sonneneinstrahlung, gefolgt von Regen. Benachrichtigungssysteme werden heutzutage von unterschiedlichen Abonnenten genutzt und müssen deshalb für vielfältige Einsatzmöglichkeiten konzipiert sein - die Möglichkeit der *Parametrisierung* von Profilen schafft diese Erweiterung der Ausdrucksstärke von Profildefinitionen.

Im Rahmen der Studienarbeit von Sven Bittner wurde das System PrimAS [1] entwickelt, welches eine effiziente Filterung von einfachen Ereignissen ermöglicht. In unseren Diplomarbeiten wird PrimAS um die Möglichkeit zur Filterung zusammengesetzter Ereignisse und deren Parametrisierung erweitert. Der bisherige, zentralisierte Filteransatz für zusammengesetzte Ereignisse wird zu einem verteilten, internetweit nutzbaren Benachrichtigungssystem ausgebaut. Dabei stehen die Aspekte der Skalierbarkeit und Effizienz im Vordergrund, welche für grosse Profil- und Ereignismengen mit zentralisierten Ansätzen nicht zu erreichen sind.

Unser System ist das erste Benachrichtigungssystem, in dem die Semantik von zusammengesetzten Ereignissen mit Hilfe von Parametern an die Bedürfnisse der Benutzer und Applikationen angepasst werden kann. Die Verteilung des Systems steigert die Effizienz und die Skalierbarkeit unter Beibehaltung der genannten Systemeigenschaften. Es werden zum einen verschiedene Strategien zur Filterung zusammengesetzter Ereignisse implementiert und miteinander verglichen. Zum anderen werden mehrere Verteilungsstrategien untersucht, implementiert und hinsichtlich Netzlast und Filterzeit verglichen. Als Ergebnis soll ein effizienter, skalierbarer, parametrisierbarer Benachrichtigungsdienst für zusammengesetzte Ereignisse entstehen, welcher insbesondere die Anforderungen für einen Einsatz in der Gebäudesteuerung erfüllt.

Zusammengesetzte Ereignisse und Parametrisierung. Benutzer können an einfachen Ereignissen oder an zusammengesetzten Ereignissen interessiert sein, z.B. an der *Sequenz* zweier Ereignisse, d.h. nach dem Ereignis e_1 tritt das Ereignis e_2 auf. Mögliche Operationen einer Ereignis-Algebra sind die *temporale Sequenz* $e_3 = (e_1; e_2)_t$, die *temporale Disjunktion* $e_3 = (e_1|e_2)_t$, die *temporale Konjunktion* $e_3 = (e_1, e_2)_t$, der *Zähleroperator* $e_3 = e_1^{[n]}$ und die *Negation* $e_3 = (\bar{e}_1)_t$. Die *temporale Sequenz* beschreibt z.B. das Auftreten eines Ereignisses e_1 vor einem Ereignis e_2 , wobei die Zeitpunkte von e_1 und e_2 nicht weiter als die Zeitspanne t entfernt sein dürfen. Nähere Informationen zur Algebra finden sich in [4].

Es gibt verschiedene Möglichkeiten, aufeinander folgende Ereignisse zu bearbeiten. Je nach Anwendungsgebiet ist es wünschenswert bestimmte Parameter einzustellen, welche die Abarbeitung der Events festlegen. Zum Beispiel ist ein Abonnent eines Aktientickers an *jeder* Kursänderung interessiert, wogegen ein Statistiker eher an dem *ersten* Auftreten eines Ereignisses (z.B. erstes Golden Goal im Fußball) interessiert ist. Die folgenden drei Parametrisierungen für zusammengesetzte Ereignisse [4] werden im Rahmen der Diplomarbeit implementiert:

1. *Auswahl der Ereignisinstanz:* Bei mehrmaligem Auftreten eines Teilereignisses wird die *erste*, *letzte* oder *x-te* Instanz des Teilereignisses verwendet.
2. *Ereignis-Instanz-Verarbeitung:* Treten mehrere Instanz-Paare eines zusammengesetzten Ereignisses auf, können verschieden Strategien zu deren Bearbeitung eingestellt werden.
3. *Auswertungsstrategie der Ereignisse:* Zusammengesetzte Ereignisse werden entweder am Ende ihres Zeitrahmens ausgewertet oder sofort nach Eintreffen aller Teilereignisse.

Verteilter Benachrichtigungsdienst. Ein verteiltes Benachrichtigungssystem besteht aus mehreren Netzknoten, welche über Punkt-zu-Punkt Verbindungen Nachrichten untereinander austauschen. Die Anbieter und Abonnenten, auch *Klienten* genannt, kommunizieren mit dem Dienst über einen beliebigen Netzknoten, ihrem *Vermittler*. Die Verteilung des Dienstes ist dadurch transparent für die Nutzerseite. In der Literatur [3, 7] existieren verschiedene Strategien zur Verteilung eines Benachrichtigungssystems:

Rendezvousknoten: Rendezvousknoten, vorgestellt von Pietzuch und Bacon [7], sind eine spezielle Art von Vermittlungsknoten, welche für bestimmte Ereignistypen zuständig sind. Alle Quellenprofile und Profile dieses Typs werden dort gespeichert, Ereignisse solchen Typs gefiltert.

Verteilte Auswertung: In der verteilten Auswertung, beschrieben von Carzaniga, Rosenblum und Wolf [3], werden Quellenprofile an alle Vermittler des verteilten Dienstes weitergeleitet. Profile werden an die Vermittler gesendet, welche mit Anbietern potentiell passender Ereignisse verbunden sind. Die Ereignisse werden vom Vermittler ihres Anbieters gefiltert und Benachrichtigungen über deren Netz weitergeleitet.

Die Effizienz der vorgestellten Verteilungsstrategien kann durch die im folgenden genannten Optimierungen [6] verbessert werden. Dabei treten die Vermittler des Systems als Stellvertreter für ihre Klienten auf, so dass Vermittler nur mit ihnen verbundene Klienten und ihre Nachbarvermittler im Netz kennen. Ansatzpunkt der Optimierungen ist die Verminderung der Mächtigkeit der zu filternden Profilmenge.

Gleichheit: Zwei Profile sind gleichartig, wenn die Menge der zu ihnen passenden Ereignisse gleich ist. Ein Profil braucht nur dann an einen Nachbarknoten weitergeleitet werden, wenn an diesen noch kein gleichartiges gesendet wurde.

Bedecken: Ein Profil p_x bedeckt ein Profil p_y , wenn die Menge der passenden Ereignisse zu p_x eine Obermenge der passenden Ereignisse zu p_y ist. Profile für Nachbarknoten müssen nur dann gefiltert werden, wenn noch keine bedeckenden für diesen gefiltert werden.

Verschmelzen: Profil p_x ist eine Verschmelzung der Profile p_y und p_z , wenn die Menge passender Ereignisse von p_x eine Obermenge der Vereinigung passender Ereignisse von p_y und p_z ist. Werden mehrere Profile verschmolzen, so vermindert sich die zu filternde Profilanzahl.

Als Netzstruktur des Benachrichtigungssystems wird eine Graphstruktur gewählt. Es wird nicht direkt auf dem physischen Netz aufgebaut, sondern vielmehr wird ein logisches Zwischennetz angelegt. Auf diesem können effektive und applikationsrelevante Routingalgorithmen implementiert werden. Dadurch ist es ausreichend ein azyklisches Netz anzunehmen, da die physische Netzstruktur nicht widergespiegelt wird. Probleme, wie das Fehlerrisiko einer Verbindung ergeben sich nicht, da die Netzwerkschicht den besten Weg zwischen zwei physischen Knoten findet - sofern dieser existiert. Die gewählte azyklische logische Netzstruktur verlagert die Probleme der Verhinderung von Kreisen und doppelter Sendungen auf die vorhandenen Netzwerkschichten.

Ziele und Planung. Ziel der Diplomarbeiten ist ein effizientes, verteiltes und skalierbares Benachrichtigungssystem für parametrisierbare, zusammengesetzte Ereignisse. Beide Arbeiten entstehen im Zusammenhang mit dem Projekt MediAS [8] an der Freien Universität Berlin. Steven König's Arbeit [5] entwickelt ein effizientes zentralisiertes System (CompAS) zur Filterung zusammengesetzter Ereignisse. Die Erweiterung zu einem verteilten Benachrichtigungsdienst (DAS) zur Verbesserung der Skalierbarkeit findet in Sven Bittner's Diplomarbeit [2] statt.

In beiden Systemen werden detaillierte Analysen zum Platzbedarf und zur Filterzeit bzgl. der Komposition von Ereignissen durchgeführt. Unterschiedliche Implementierungsvarianten werden dazu gegenübergestellt, verglichen und analysiert. Im weiteren werden verschiedene Ansätze zur Verteilung implementiert und in Bezug auf Filterzeit und Netzlast verglichen. Durch den Einsatz des gleichen zentralen Filteralgorithmus können die verschiedenen Verteilungsvarianten in Hinblick auf ihre Effizienz analysiert, die Schwachstellen gefunden und verbessert, sowie Randbedingungen für optimale Einsatzmöglichkeiten erarbeitet werden. Somit wird als Ergebnis dieser Diplomarbeiten der Prototyp eines effizienten, verteilten und skalierbaren Benachrichtigungssystems für parametrisierbare, zusammengesetzte Ereignisse vorliegen. Dieser wird die Anforderungen für einen Einsatz im Rahmen der Gebäudesteuerung erfüllen und die heute dort vorhandenen Effizienz- und Darstellungsprobleme von Anforderungsdefinitionen lösen können.

Literatur

- [1] S. Bittner. Implementierung eines effizienten Matchingverfahrens für Benachrichtigungssysteme. Studienarbeit, Freie Universität Berlin, verfügbar in [8], 2002.
- [2] S. Bittner. Implementierung und Analyse eines verteilten Benachrichtigungsdienstes. Extended Abstract der Diplomarbeit, Freie Universität Berlin, verfügbar in [8], 2003.
- [3] A. Carzaniga, D. S. Rosenblum, and A. L. Wolf. Interfaces and Algorithms for a Wide-Area Event Notification Service. Technical Report CU-CS-888-99, University of Colorado, October 1999.
- [4] A. Hinze and A. Voisard. A Parameterized Algebra for Event Notification Services. In *Proceedings of TIME 2002*, Manchester, UK, 2002.
- [5] S. König. CompAS: Ein effizientes, parametrisierbares Benachrichtigungssystem f. zusammengesetzte Ereignisse. Extended Abstract der Diplomarbeit, Freie Universität Berlin, verfügbar in [8], 2003.
- [6] G. Mühl, L. Fiege, and A. Buchmann. Filter Similarities in Content-Based Publish/Subscribe Systems. In *Proceedings of ARCS '02*, 2002.
- [7] P. Pietzuch and J. Bacon. Hermes: A Distributed Event-Based Middleware Architecture. In *Proceedings of DEBS'02*, 2002.
- [8] Project MediAS: Efficient Internet-wide Notification on Composite Events. Homepage des Projektes: http://page.inf.fu-berlin.de/~hinze/projects/project_medias.html.