

Neue Ansätze zur Planung von Klinikkommunikationssystemen mittels stochastischer Leistungsmodellierung

Markus Siegle*, Bernhard Wentz**, Andreas Klingler**, Markus Simon**

Universität Erlangen-Nürnberg

IMMD 7* bzw. IVmed**

Martensstraße 1/3, 91058 Erlangen

siegle@informatik.uni-erlangen.de

1. Einleitung

Das Klinikum der Universität Erlangen-Nürnberg besteht aus über 30 klinischen Einrichtungen und Abteilungen, die über einen weiten Bereich des Erlanger Stadtgebiets verteilt sind. Die erforderliche Verkabelung dieser Einrichtungen untereinander mittels eines Glasfaserrings ist bereits abgeschlossen. Auch die Innenverkabelung der genannten Einrichtungen ist zum größten Teil realisiert. Durch die verteilte und dezentrale Lage der einzelnen Einrichtungen ergeben sich besondere Anforderungen an die Netzstruktur im Klinikum. Aus diesem Grund wurde in Erlangen im Rahmen des sog. Netzwerk-Investitions-Programmes (NIP) ein leistungsfähiges Glasfaser-Backbone-Netz in FDDI- Technologie aufgebaut, das im wesentlichen die einzelnen Kliniken in der Erlanger Innenstadt mit den Dienstleistungszentren (Informationsverarbeitung Medizin und Regionales Rechenzentrum) im Universitätssüdgelände verbindet. Dies wurde durch Vermittlungsknoten (Router) im Universitätssüdgelände und in der Innenstadt erreicht, an die die lokalen Netze der verschiedenen Einrichtungen angeschlossen sind. Vom Innenstadt-Router verzweigen verschiedene Glasfaserstränge zu sog. Aufpunkten in den Kliniken der Innenstadt, die Ausgangspunkte der lokalen Verkabelung in den einzelnen Kliniken darstellen. Dabei wird eine sternförmige Verkabelungsstruktur mit ATM-fähigen Twisted-Pair-Kabeln bevorzugt, wobei ältere mit diesen Vorgaben nicht konforme Netze (z.B. über Koaxialkabel) dennoch weitgehend integriert werden [1].

Als Hardware für die Server dienen neben einigen noch existierenden Mainframes im wesentlichen UNIX-Rechner sowie PCs unter Novell bzw. Windows NT. Die meisten Client-Endgeräte sind bereits PCs. Neben dem Patientenmanagementsystem IS-H der Firma SAP, das seit Mitte 1996 in Erlangen eingeführt wird, müssen nach und nach klinische Subsysteme (z.B. Laborsysteme, aber auch Abteilungssysteme) über eine Integrationsplattform zu einem Klinikkommunikationssystem zusammengeschlossen werden. Durch das Fehlen einer klinikumbergreifenden DV-Lösung haben sich in einigen Bereichen des Erlanger Klinikums bereits eine ganze Reihe Subsysteme etabliert. Diese Systeme werden seit Ende 1995 über die Erlanger Kommunikationsdrehscheibe integriert [1]. Die Kommunikationsdrehscheibe besteht im wesentlichen aus zwei Komponenten, der sog. Kommunikationsdatenbank und einem Kommunikationsserver.

In der Kommunikationsdatenbank werden die Patientenstammdaten, klinische Befunde, Leistungsdaten, etc. zwischengespeichert. Etwa 20 Subsysteme haben via Standard-SQL oder ODBC Zugang zu bestimmten Datenbereichen in dieser Datenbank und sind somit proprietär angebunden.

Mit der Ablösung der alten Subsysteme durch neuere DV-Lösungen mit Standard-Schnittstellen ist zunehmend der Einsatz eines sog. Kommunikationsservers erforderlich, der Kommunikationsstandards wie HL7, EDIFACT und

DICOM3 unterstützt. Daher wurden in Erlangen seit Ende 1996 erste Tests mit dem System DataGate der Firma STC durchgeführt.

Die Gesamtheit aller Systeme die im Medizinischen Versorgungsnetz über die Kommunikationsdrehscheibe angeschlossen sind, wird hier als Erlanger Klinikkommunikationssystem (EKKS) bezeichnet (siehe Abbildung 1).

Um z.B. die Größenordnung des Nachrichtendurchsatzes im EKKS abzuschätzen sowie Aussagen über Performance und Auslastung eines Kommunikationsservers zu treffen, müssen neue Ansätze entwickelt werden. In diesem Papier wird vorgeschlagen, dafür stochastische Modellierung mit Prozeßalgebren einzusetzen, und auf dieser Basis Leistungs- und Zuverlässigkeitsprognosen zu erstellen.

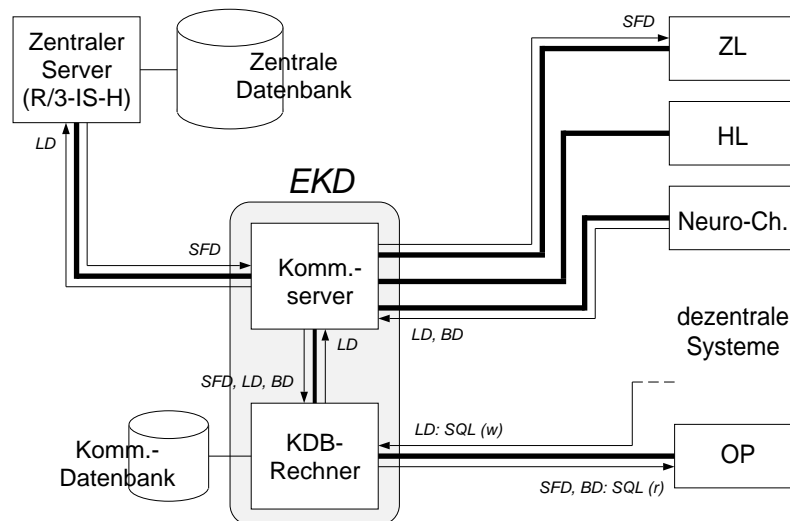


Abbildung 1: Das Erlanger Klinikkommunikationssystem (EKKS)

2. Modellierung mit Stochastischen Prozeßalgebren

Stochastische Prozeßalgebren (SPA) sind ein moderner Ansatz für die Modellierung des Verhaltens verteilter, nebenläufiger Systeme, an dem in Erlangen intensiv gearbeitet wird [2]. Neben der Überprüfung funktionaler Eigenschaften sind sie dafür geeignet, Leistungs- und Zuverlässigkeitsmaße zu ermitteln.

Stochastische Prozeßalgebren basieren auf klassischen Prozeßalgebren wie CCS [3] oder CSP [4], sind jedoch gegenüber diesen um Zeitaspekte erweitert, so daß sie auch eine Untersuchung des temporalen Systemverhaltens ermöglichen.

Zur Modellbeschreibung wird eine LOTOS-ähnliche [5] formale Sprache verwendet. Der Modellierer beschreibt das Verhalten der Prozesse, sowie die Kommunikations- und Synchronisationsbeziehungen zwischen den beteiligten Prozessen. Prozesse sind die Aktivitätsträger, die Aktionen entweder autonom oder in Kooperation mit anderen Prozessen ausführen. Zur Beschreibung eines Prozesses stehen Sprachkonstrukte für sequentielle Ausführung von Aktionen, für Verzweigung und rekursives Verhalten zur Verfügung. Wichtigstes Sprachkonstrukt für den Aufbau des Gesamtmodells aus einzelnen Prozessen ist der Paralleloperator, mit dem Prozesse parallel geschaltet und dabei gegebenenfalls über bestimmte Aktionen synchronisiert werden können.

Für die Analyse eines SPA-Modells wird die sprachliche Beschreibung mit Hilfe semantischer Regeln in ein beschriftetes Transitionssystem transformiert. Auf dieser automatenähnlichen Repräsentation können funktionale Analysen (z.B. Erreichbarkeits- oder Deadlockanalyse) durchgeführt werden. Gleichzeitig ist das Transitionssystem die Basis für eine Analyse des zeitlichen Verhaltens. Durch eine numerische Auswertung können stationäre oder transiente Zustandswahrscheinlichkeiten und daraus abgeleitete Leistungs- und Zuverlässigkeitsgrößen (z.B. mittlere

Anzahl von wartenden Aufträgen, Wahrscheinlichkeit für den gleichzeitigen Ausfall zweier Server) berechnet werden. Das in Erlangen entwickelte TIPP-Tool bietet dem Anwender eine durchgängige Unterstützung in allen Phasen der Modellierung, angefangen von der Modellerstellung (textuelle oder graphische Modelleingabe) bis hin zur numerischen Auswertung und Ergebnispräsentation.

3. Ein einfaches Modell des EKKS

In diesem Abschnitt wird ein stark vereinfachtes Grundmodell des Erlanger Klinikkommunikationssystems beschrieben, mit dem Ziel, unsere Vorgehensweise exemplarisch darzustellen.

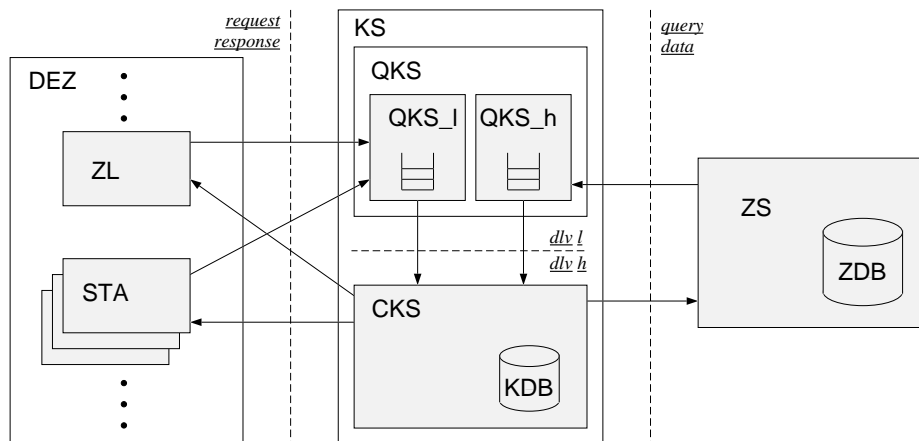


Abbildung 2: Prozeß-Interaktionsmodell des EKKS

Abbildung 2 zeigt einen Überblick über die beteiligten Prozesse und ihre Synchronisationsbeziehungen. Man erkennt in der Abbildung drei Blöcke: Links die dezentralen Systeme (DEZ), in der Mitte der Kommunikationsserver (KS) und rechts der zentrale Server (ZS). Jeder dieser Blöcke kann als Prozeß aufgefaßt werden, der sich, wie am Beispiel des KS zu erkennen ist, aus parallelgeschalteten Unterprozessen zusammensetzt. Im Fall des KS sind dies ein Unterprozeß, der zwei Eingangswarteschlangen repräsentiert (QKS für “queueing”) und ein weiterer Unterprozeß, der die eigentliche Bearbeitung von Nachrichten im KS durchführt (CKS für “computing”). Dieser modulare Aufbau eines Prozesses kann sich hierarchisch nach unten fortsetzen. So besteht beispielsweise der Prozeß QKS aus zwei weiteren Unterprozessen, nämlich einer Warteschlange mit niedriger (CKS_l für “low”) und einer mit hoher Priorität (CKS_h für “high”). Bei den dezentralen Systemen stellt ZL das Zentrallabor und STA eine typische Station dar.

Die Synchronisationsbeziehungen werden in Abbildung 2 durch Pfeile angedeutet. An den Prozeßschnittstellen (durch gestrichelte Linien dargestellt) wird die Menge der synchronisierenden Aktionen angegeben. Die Synchronisation zwischen DEZ und KS geschieht beispielsweise über die beiden Aktionen request und response. In unserem Beispiel wird aus Gründen der Einfachheit synchrone Kommunikation verwendet. Asynchrone Kommunikation läßt sich durch Zwischenschaltung von zusätzlichen Kanalprozessen modellieren.

Das Blockbild in Abbildung 2 enthält noch keinerlei Information über das interne Verhalten der einzelnen Prozesse. Dieses wird mit Hilfe der TIPP-Eingabesprache spezifiziert. Als Beispiel betrachten wir die folgende Beschreibung:

```
process KS := QKS | {dlv_l, dlv_h} | CKS endproc
```

Sie drückt aus, daß der Prozeß KS aus einer Parallelschaltung der beiden Partner QKS und CKS besteht. Dabei wird über die Aktionen dlv_l und dlv_h synchronisiert. Prozeß QKS ergibt sich als Parallelschaltung zweier parametrisierter FIFO-Prozesse, die die Funktionalität von Warteschlangen haben:

```
process QKS := FIFO[request,dlv_l](0,3) ||| FIFO[data,dlv_h](0,3) endproc
```

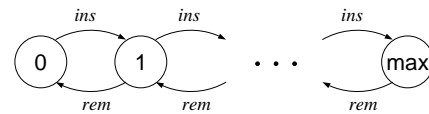
Das Verhalten eines allgemeinen FIFO-Prozesses wird folgendermaßen beschrieben

```
process FIFO[ins,rem](n,nmax) :=
```

```
  [n < nmax] -> (ins.FIFO[ins,rem](n+1,nmax)) []
```

```
  [n > 0]    -> (rem.FIFO[ins,rem](n-1,nmax))
```

```
endproc
```



Auf der rechten Seite sehen wir das zugehörige semantische Modell, d.h. das Transitionssystem, bestehend aus Zuständen und Transitionen, die den Aktionen ins (insert) und rem (remove) entsprechen.

Als weiteres Beispiel betrachten wir noch eine Zeile aus der Spezifikation des Prozesses ZL:

```
process ZL := (init,lambda_init).ZL_working(0) ...
```

Hier sehen wir die Verwendung einer zeitbehafteten Aktion. Der Prozeß ZL führt zunächst die Aktion init aus, wofür eine stochastisch verteilte Zeit benötigt wird. Anschließend weist er das Verhalten ZL_working(0) auf. In diesem Fall wird eine exponentiell verteilte Zeit angenommen, die durch die Angabe der Rate (lambda_init) eindeutig charakterisiert ist.

Nach der Erstellung des Modells muß dieses mit Meßdaten parametrisiert werden, damit realistische Werte zugrundegelegt werden. Die Modellanalyse und die Berechnung der Leistungs- und Zuverlässigkeitskenngrößen wird vom TIPP-Tool automatisch durchgeführt. Abbildung 3 zeigt exemplarisch das Ergebnis einer Modellierungsstudie. Hier wird die Auslastung des zentralen Servers in Abhängigkeit von der Trefferrate der Kommunikationsdatenbank prognostiziert, wobei als zusätzlicher Parameter die Leistung des Servers berücksichtigt wird.

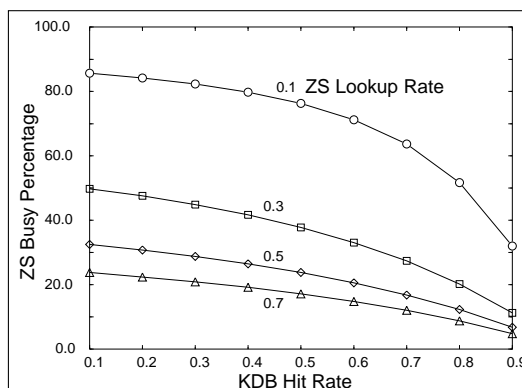


Abbildung 3: Ausgabe des TIPP-Tools

4. Diskussion

Das hier vorgestellte einfache Grundmodell stellt sicher noch keine adäquate Beschreibung der komplizierten Abläufe innerhalb des EKKS dar. Es dient jedoch als Basis, die durch schrittweise Verfeinerung ausgebaut wird, bis das Modell alle relevanten Aspekte des projektierten realen Systems wiedergibt.

Zwar haben viele Kliniken in Deutschland bereits z.B. einen Kommunikationsserver eingeführt, verlässliche Aussagen über den zu erwartenden Nachrichtendurchsatz und die davon abhängige Größenordnung der Hardware- und Software-Ressourcen sind bisher noch schlecht abschätzbar.

Die neuen Ansätze der Modellierung mittels stochastischer Prozeßalgebren sind daher sehr vielversprechend. Sie ermöglichen, sofern sie mit realistischen, aus Messungen an einer zugänglichen Konfiguration gewonnenen Daten parametrisiert werden, Prognosen des zu erwartenden Leistungsverhaltens projektierte Systeme.

Literatur

- [1] B. Wentz. Erlanger Kommunikationsdrehscheibe. Interner Bericht 10, Universität Erlangen–Nürnberg, Informationsverarbeitung Medizin, 1996.
- [2] N. Götz, H. Hermanns, U. Herzog, V. Mertsiotakis, and M. Rettelbach. *Quantitative Methods in Parallel Systems*, chapter Constructive Specification Techniques – Integrating Functional, Performance and Dependability Aspects. Springer, 1995.
- [3] R. Milner. *A Calculus of Communicating Systems*, volume 92 of *LNCS*. Springer, Berlin, New York, 1980.
- [4] C.A.R. Hoare. *Communicating Sequential Processes*. Prentice-Hall, Englewood Cliffs, NJ, 1985.
- [5] E. Brinksma. A tutorial on LOTOS. In *Protocol Specification, Testing, and Verification 5*, pages 171–194. IFIP WG 6.1, 1985.