Zeitreihenprognose in relationalen Datenbanksystemen*

Ulrike Fischer

Professur für Datenbanken
Fakultät Informatik
Technischen Universität Dresden
ulrike.fischer@tu-dresden.de

Abstract: Die Zeitreihenprognose ist eine wichtige Analysetechnik für eine Vielzahl von Anwendungsgebieten. Prognosen werden dabei oft von fachfremden Nutzern auf großen, multidimensionalen Datensätzen angefragt, die eine geringe Antwortzeit erwarten. Aktuelle Datenbanksysteme unterstützen jedoch gar nicht oder nur sehr eingeschränkt Prognoseverfahren, sodass Prognosewerte manuell von Experten in dedizierten Softwareumgebungen berechnet werden müssen. Im Rahmen dieser Dissertation stellen wir einen neuartigen Ansatz vor, der die Integration der Zeitreihenprognose in ein relationales Datenbanksystem verfolgt. Mittels SQL können Nutzer ohne statistische Vorkenntnisse deklarative Prognoseanfragen stellen, die dann transparent und automatisch vom Datenbanksystem verarbeitet werden. Dabei diskutieren wir Optimierungstechniken für drei verschiedene Arten von Prognoseanfragen: Ad-hoc Anfragen, wiederkehrende Anfragen und kontinuierliche Anfragen. Alle Optimierungsansätze reduzieren die Ausführungszeit von Prognoseanfragen und gewährleisten eine hohe Prognosegenauigkeit.

1 Einleitung

Die Zeitreihenprognose ist seit langer Zeit ein wichtiger Bestandteil der Forschungslandschaft und bietet immer noch viele offenen Forschungsfragen [GH06]. Die Prognose zukünftiger Daten ist in einer Vielzahl von Anwendungsgebieten unverzichtbar, zum Beispiel bei der Produktionsplanung, der Regulierung intelligenter Stromnetze oder im Online Marketing Bereich. Die Nutzung von Prognosetechniken ermöglicht es, vorausschauend zu agieren, anstatt nur auf vergangene Ereignisse zu reagieren.

In der Vergangenheit wurde die Zeitreihenprognose von hoch qualifizierten Experten mit langer Erfahrung im Unternehmen oft manuell in dedizierten, statistischen Programmen durchgeführt. In den letzten Jahren können wir einen Paradigmenwechsel beobachten, der nicht nur die Zeitreihenprognose sondern generell komplexe statistische Verfahren betrifft. Traditionelle Datenbanksysteme müssen immer größer werdende Datenmengen verarbeiten, die kontinuierlich aus einer Vielzahl von Datenquellen generiert werden. Diese Entwicklung bietet nicht nur große Herausforderungen sondern auch vielfältige Möglichkeiten

*Englischer Titel der Dissertation: “Forecasting in Database Systems”

hinsichtlich der Analyse komplexer Datenbestände in Echtzeit. Folglich beinhaltet die Datenanalyse in modernen Datenbankmanagementsystemen mehr und mehr komplexe statistische Methoden, die weit über klassische Roll-Up und Drill-Down Operationen hinausgehen. Dabei werden Prognosewerte oft von Nicht-Mathematikern und Nicht-Statistikern abgefragt, die sich nicht mit Aufgaben wie der Auswahl oder der Schätzung des Prognosemodells beschäftigen möchten. Auf der anderen Seite müssen Prognosewerte aber auch automatisiert und gänzlich ohne menschliches Eingreifen berechnet werden können, beispielsweise bei der Regulierung intelligenter Stromnetze.

Diverse Forschungsgruppen und kommerzielle Datenbankanbieter beschäftigen sich mit der Integration von statistischen Verfahren wie der Zeitreihenprognose in klassische Datenbankmanagementsysteme. Jedoch verfolgen die meisten existierenden Verfahren einen Black-Box Ansatz und versuchen, die Änderungen am Datenbanksystem so minimal wie möglich zu gestalten, beispielsweise durch die Verwendung benutzerdefinierter Funktionen (UDFs) [CDD+09] oder die Verknüpfung des Datenbanksystems mit externen, statistischen Programmen [GLW+11]. Zwar sind derartige Ansätze einfacher zu realisieren, jedoch entgehen ihnen vielfältige Optimierungsmöglichkeiten. Weiterhin erfordern die meisten existierenden Ansätze immer noch die explizite Erstellung und Nutzung von Modellen, und ignorieren damit das Konzept deklarativer Datenbankanfragen.

Im Rahmen der Dissertation [Fis14] stellen wir einen Ansatz vor, der eine vollständige Integration der Zeitreihenprognose in ein traditionelles, relationales Datenbanksystem verfolgt. Eine enge Kopplung zwischen dem Datenbanksystem und den statistischen Prognosemethoden sorgt für Konsistenz zwischen Modellen und Daten, und erhöht sowohl die Nutzbarkeit als auch die Effizienz des Prognoseprozesses. Im Gegensatz zu sogenannten Flash-Back Anfragen, die eine vergangene Sicht auf die Daten ermöglichen, unterstützen wir *Flash-Forward* Anfragen. Mittels SQL können Nutzer ohne statistische Vorkenntnisse deklarative Prognoseanfragen stellen, die dann transparent und automatisch vom Datenbanksystem verarbeitet werden.

Im folgenden Kurzbeitrag zur Dissertation gibt Abschnitt 2 zunächst einen Überblick über die Grundlagen der Zeitreihenprognose. Abschnitt 3 stellt dann die konzeptionelle Architektur unseres Gesamtsytems, des *Flash-Forward Datenbanksystems* (F^2DB), genauer vor. Die Abschnitte 4 bis 6 diskutieren unterschiedliche Komponenten und Optimierungstechniken von F^2DB. Abschließend fassen wir das Erreichte in Abschnitt 7 zusammen.

2 Grundlagen der Zeitreihenprognose

Die Prognose von Zeitreihen basiert auf einem generalisierten modellbasierten Prozess, aus diesem wir Anforderungen an und architektonische Umsetzungsmöglichkeiten für ein Prognosesystem ableiten können [FL14].

Modellbasierter Prognoseprozess Eine *Zeitreihe* bezeichnet eine zeitliche Folge von Beobachtungen in äquidistanten Zeitabständen. Die *Zeitreihenprognose* beschäftigt sich mit der Schätzung neuer, bisher ungesehener Beobachtungen, sogenannter *Prognosewerte*, für zukünftige Zeitpunkte. Der *Prognosehorizont* beinhaltet dabei das gesamte Intervall

vom aktuellen Zeitpunkt bis zu einem gegebenen zukünftigen Zeitpunkt. Zur Berechnung der Prognosewerte werden im Allgemeinen modellbasierte *Prognosemethoden* eingesetzt, die die historische Entwicklung der Zeitreihe in einem *Prognosemodell* abbilden. Weit verbreitete und oft eingesetzte Vertreter von Prognosemethoden sind beispielsweise das exponentielle Glätten und die ARIMA-Modelle [BJR08]. Die modellbasierte Zeitreihenprognose besteht aus drei wesentlichen Grundschritten. Der erste Schritt, die *Modellerstellung*, beinhaltet die Definition der relevanten Variablen, der Prognosemethode sowie die Schätzung der *Modellparameter*. Für die Parameterschätzung sind dabei oft aufwendige Optimierungsverfahren erforderlich, sodass dieser Schritt die zeitaufwändigste Komponente bei der modellbasierten Prognose darstellt. Ein einmal erstelltes Modell kann nun wiederholt und kostengünstig für die Berechnung der Prognosewerte verwendet werden (*Modellnutzung*). Treffen neue Zeitreihenwerte im System ein, ist gegebenfalls eine *Modellwartung* erforderlich. Dies beinhaltet zum einen eine Evaluierung des vorhandenen Modells und zum anderen eine Anpassung des Modells an die neuen Zeitreihencharakteristiken, die oft eine komplette Neuschätzung der Modellparameter nach sich zieht.

Systemanforderungen Die Zeitreihenprognose ist eine wichtige Analysetechnik für eine Vielzahl von Anwendungsgebieten. Damit sollte ein System, das sowohl den vorgestellten Prognoseprozess als auch typische Applikationsanforderungen unterstützt, gewisse Voraussetzungen erfüllen. Zunächst müssen weit verbreitete Zeitreihenprognosemethoden für die Modellerstellung verfügbar sein. Weiterhin ist eine generische Schnittstelle wünschenswert, die es ermöglicht, beliebige, domänenspezifische Prognosemethoden zum System hinzuzufügen. Eine einfache und deklarative Anfragesprache erlaubt es jedem Nutzer, auch ohne statistisches Hintergrundwissen Prognosen zu berechnen. Echtzeitanforderungen (z.B. weniger als eine Sekunde im Online Werbebereich) auf großen multidimensionalen Daten erfordern eine effiziente Erstellung von Prognosemodellen. Da aber gerade dies sehr zeitaufwändig ist, sollten Mechanismen zur Speicherung und Wiederverwendung von Prognosemodellen zur Verfügung gestellt werden, möglichst auch für andere Anfragen und Nutzer. Schlussendlich ist eine automatisierte Wartungsumgebung erforderlich, die Prognosemodelle transparent an neue Zeitreihencharakteristiken anpasst und die teure Modellwartung nur bei Bedarf ausführt.

Architektonische Umsetzung Existierende Ansätze zur Umsetzung statistischer Verfahren in Datenbankmanagementsystemen können in (1) externe Verfahren, (2) partielle Integrationsverfahren und (3) vollständige Integrationsverfahren klassifiziert werden. Externe Verfahren nutzen dedizierte statistische Programme (Matlab, R) und unterstützen dabei eine Vielzahl komplexer Prognosemethoden. Allerdings ist ein Export der Daten aus der Datenbank notwendig, es gibt keine Möglichkeit, deklarative Prognoseanfragen zu stellen und auch keine Mechanismen zur Speicherung, Wiederverwendung und automatisierten Wartung von Prognosemodellen. Partielle Integrationsansätze [CDD+09, GLW+11] gehen einen Schritt weiter und bieten eine eingeschränkte Integration der Zeitreihenprognose an. Jedoch wird hier der Prognoseprozess oft als Black-Box Ansatz innerhalb einer benutzerdefinierten Funktion oder eines R-Skriptes betrachtet, womit keine effiziente Optimierung, Wiederverwendung und Wartung von Modellen möglich ist. Vollständige Integrationsansätze, wie beispielsweise MauveDB [DM06], schlagen bereits eine vollständige Integration von statistischen Modellen als native Datenbankobjekte vor und bieten bei-

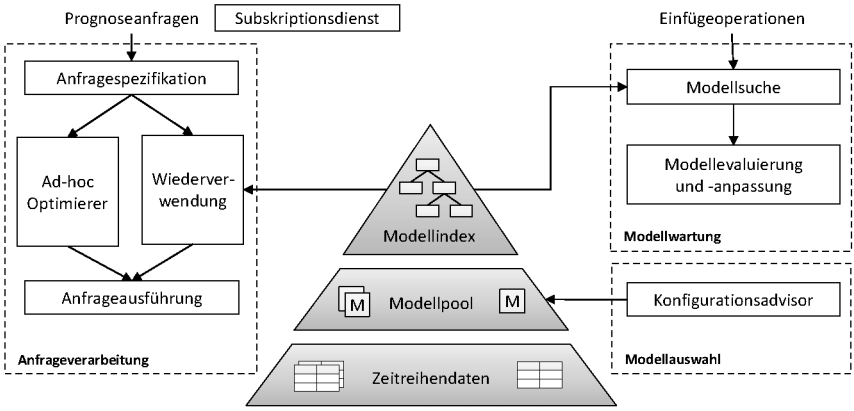

Abbildung 1: Flash-Forward Datenbanksystem F^{2}DB

spielsweise auch Wartungsmechansimen an. Existierende Ansätze adressieren die Zeitreihenprognose dabei jedoch nicht oder nur sehr eingeschränkt und erfordern weiterhin die explizite Spezifikation eines Modells in einer Anfrage.

3 Konzeptionelle Systemarchitektur

Basierend auf den vorgestellten Systemanforderungen und den Limitierungen existierender Ansätze erfolgte im Rahmen der Dissertation die Konzeption und prototypische Umsetzung eines Gesamtsystems mit dem Namen F^{2}DB (*Flash-Forward Datenbanksystem*) [FRL12a, FDS+12], welches eine vollständige Integration der modellbasierten Zeitreihenprognose in ein relationales Datenbanksystem realisiert. Der Prototyp wurde dabei auf Basis des frei verfügbaren Datenbanksystems PostgreSQL entwickelt. Die wesentlichen konzeptionellen Komponenten von F^{2}DB sind in Abbildung 1 dargestellt.

Zeitreihen stellen eine spezielle Sicht auf die Daten dar und können mittels beliebiger Anfragen auf den klassischen Datenbanktabellen erstellt werden, ähnlich wie bei klassischen Sichten. Prognosemodelle werden als native Datenbankobjekte in einem gemeinsamen *Modellpool* abgelegt. Ein *Modellindex* sichert das schnelle Auffinden existierender Modelle für eintreffende Prognoseanfragen und Einfügeoperationen.

Die Verarbeitung deklarativer Prognoseanfragen erfordert zunächst die Erweiterung der Anfrageverarbeitung eines traditionellen Datenbanksystems. Die *Anfragespezifikation* erkennt neue prognosespezifische Schlüsselwörter und überführt die Anfrage in einen internen, logischen Anfragebaum. Ausgehend von der initialen Analyse der Prognoseanfrage ergeben sich zwei grundsätzliche Pfade zur Anfrageverarbeitung. Auf der einen Seite können neue Prognosemodelle zur Laufzeit erstellt werden, wobei die Modellerstellung im Rahmen der *Anfrageausführung* mittels neuer physischer Prognoseoperatoren umgesetzt ist. Der *Ad-hoc Optimierer* hat dabei die Aufgabe, die Laufzeit und den Prognosefehler

derartiger Ad-hoc Prognoseanfragen zu minimieren. Auf der anderen Seite können existierende Modelle aus dem Modellindex geladen und für die Berechnung der Prognose *wiederverwendet* werden. Neben der Anfrageverarbeitung ergeben sich Änderungen bei der Verarbeitung von Einfügeoperationen, da nun eine Wartung der materialisierten Prognosemodelle notwendig ist. Dazu müssen die von einer Einfügeoperation betroffenen Modelle im Modellindex *gesucht*, *evaluiert* und bei Bedarf *angepasst* werden. Daraus ergibt sich die Frage, welche Modelle im Modellindex abgelegt werden sollten, um zum einen die Wartungskosten zu minimieren, zum anderen aber eine möglichst effiziente und genaue Wiederverwendung zu ermöglichen. Basierend auf einer gegebenen Anfragelast schlägt der *Konfigurationsadvisor* eine möglichst optimale Konfiugration von Modellen für den Modellpool vor. Eine letzte Komponente, der *Subskriptionsdienst*, ermöglicht die einmalige Registrierung kontinuierlicher Prognoseanfragen. Neu eintreffende Zeitreihenwerte führen dann zu automatischen Notifikationen an den Subskribenten.

4 Verarbeitung und Optimierung von Ad-hoc Prognoseanfragen

In diesem Abschnitt diskutieren wir zunächst die Grundlagen der Verarbeitung und Optimierung von Ad-hoc Prognoseanfragen.

Spezifikation und Ausführung von Prognoseanfragen Prognoseanfragen werden mittels eines einfachen Schlüsselwortes `AS OF` spezifiziert, welches den Prognosehorizont bestimmt [FRL12a]. Eine einfache Prognoseanfrage könnte beispielsweise Verkaufsprognosen für Telefone im nächsten Monat abfragen:

```
SELECT    orderdate, quantity
FROM      facts
WHERE     pgroup = 'phones'
GROUP BY  orderdate
AS OF     now() + interval '3 months'
```

Das SQL Konstrukt `AS OF` ist dabei bereits Teil des SQL:2011 Standards und wurde bis jetzt verwendet, um eine Sicht auf die Daten in der Vergangenheit zu ermöglichen (Flash-Back Anfragen). In unserem System ist es nun möglich, mit dem selben Sprachkonstrukt eine Sicht auf die zukünftigen Daten abzufragen (Flash-Forward Anfragen). Optionale zusätzliche Schlüsselworte ermöglichen die Spezifikation von abhängigen und unabhängigen Variablen oder der Prognosemethode.

Analog zur klassischen Anfrageverarbeitung wird die Prognoseanfrage in einen internen, logischen Anfragebaum transformiert, wofür wir einen neuen, logischen *Prognoseoperator* eingeführt haben. Auf physischer Ebene wird der logische Prognoseoperator durch zwei wesentliche, physische Operatoren repräsentiert - den Operator zur Modellerstellung und den Operator zur Modellnutzung. Die entwickelten Operatoren sind dabei unabhängig vom eigentlichen Prognoseverfahren und nutzen vordefinierte Funktionen einer generischen Schnittstelle. Diese erlaubt es Datenbankadministratoren, beliebige domänenspezifische Prognosemethoden in F^2DB zu integrieren.

Im Gegensatz zur klassischen Anfrageverarbeitung sind Prognosewerte allerdings per se

ungenau. Dies führt zu einer neuen Semantik bei der Restrukturierung und Optimierung von Anfrageplänen. Unterschiedliche Anfragepläne beeinflussen nicht nur die Laufzeit sondern auch die Genauigkeit des Anfrageergebnisses. Im Rahmen der Dissertation haben wir deshalb sowohl exakte als auch approximative Restrukturierungsregeln untersucht, neue Kostenmodelle für die physischen Operatoren eingeführt und spezielle stichprobenbasierte Optimierungstechniken für zwei Gruppen von Prognoseanfragen entwickelt.

Stichprobenbasierte Anfrageoptimierung Stichprobenverfahren führen zu einer Reduktion der zu verarbeitenden Datenmenge und wurden in einer Vielzahl von Bereichen erfolgreich angewandt. Im Falle von Prognoseanfragen nutzen wir Stichprobenverfahren auf zwei verschiedene Arten: *vertikales Sampling* und *horizontales Sampling*. Beide Verfahren haben zum Ziel, die Datenmenge und damit die Anfragelaufzeit mit keinem oder nur sehr geringem Genauigkeitsverlust zu reduzieren. Vertikales Sampling adressiert Aggregationsanfragen, die eine aggregierte Prognose über eine Vielzahl von Einzelzeitreihen berechnen [FRL12b]. Die Verwendung von Einzelmodellen ist dabei oft sehr teuer, ein Modell über die aggregierte historische Zeitreihe liefert eine geringere Genauigkeit. Vertikales Sampling verwendet stattdessen nur eine Stichprobe der Einzelmodelle und gewinnt die Prognose durch eine Hochrechnung der Einzelprognosen. Spezielle Stichprobenschätzer, die auf den historischen Entwicklungen der Zeitreihen basieren, stellen eine hohe Genauigkeit selbst bei kleinen Stichprobengrößen sicher. Horizontales Sampling reduziert die Länge historischer Zeitreihen in Gruppierungsanfragen mit dem Ziel der Beschleunigung der Modellerstellung. Dazu erfolgt eine Analyse der optimalen Länge für eine kleine, ausgewählte Anzahl von Zeitreihen. Anschließend wird der aktuelle Plan neu parametriert, sodass für spätere Zeitreihen im Plan eine kürzere Historie verwendet werden kann.

Evaluierung Im Rahmen der Evaluierung haben wir sowohl den Laufzeitgewinn der vollständigen Datenbankintegration der Zeitreihenprognose gegenüber alternativen Integrationsansätzen als auch die Vorteile der vorgeschlagenen Optimierungsverfahren untersucht. Um möglichst unterschiedliche Anfragecharakteristiken abzubilden, haben wir dafür den klassischen TPC-H Benchmark angepasst. Dies erforderte zum einen Anpassungen bei der Datengenerierung, sodass Zeitreihendaten anstatt zufälliger Werte erstellt werden, und zum anderen Anpassungen bei der Anfragegenerierung, sodass Prognoseanfragen anstatt historischer Anfragen an das Datenbanksystem gesendet werden. Die Evaluierung zeigte eine signifikante Laufzeitreduzierung der vollständigen Integration gegenüber der externen Ausführung von Prognoseanfragen beziehungsweise partiellen Integrationsansätzen. Eine weitere Verbesserung der Laufzeit wurde durch die vorgestellten stichprobenbasierten Optimierungstechniken erreicht.

5 Materialisierung von Prognosemodellen

Die Vorauswahl und Materialisierung von Prognosemodellen ermöglicht es, sowohl die Laufzeit als auch die Genauigkeit von Prognoseanfragen weiter zu reduzieren. Dabei ist unser Ziel, die deklarative Verarbeitung von Prognoseanfragen beizubehalten, wofür wir zunächst einen Mechanismus für die transparente Wiederverwendung und Wartung von Prognosemodellen benötigen.

Wiederverwendung und Wartung von Prognosemodellen Die Materialisierung von Modellen wird durch zwei neue Datenstrukturen unterstützt. Ähnlich zu klassischen Datenbankindexen werden physische Prognosemodelle (Modellparameter und -zustand) in einem *Modellpool* gespeichert. Zur schnellen Identifikation passender Modelle haben wir einen *Modellindex* entwickelt, der komplett im Hauptspeicher gehalten wird [FRBL10]. Ein Prädikatenindex indexiert dabei atomare Prädikate, welche dann zu beliebigen Anfrageausdrücken verknüpft werden. Einmal erstellte und indizierte Modelle werden automatisch und transparent für Prognoseanfragen verwendet. Ein Suchalgorithmus findet hierbei während der Anfrageoptimierung passende Modelle für eine gegebene Anfrage. Wird kein passendes Modell gefunden, muss zur Laufzeit ein neues Modell erstellt werden, welches dann im Modellindex abgelegt wird und damit zukünftigen Anfragen zur Verfügung steht. Neu eintreffende Daten erfordern die Wartung der Prognosemodelle. Hierbei ist erneut eine Suche im Modellindex erforderlich, um die betroffenen Modelle zu identifizieren. Nach der Aktualisierung des Modellzustandes (interne Historie) erfolgt eine Evaluierung des Modells, um die Notwendigkeit einer Modellwartung festzustellen. Dafür haben wir unterschiedliche Evaluierungsstrategien in F^2DB integriert. Zeigt die Evaluierung, dass eine Neuschätzung der Modellparameter erforderlich ist, dann wird das Modell markiert und bei der nächsten Referenzierung durch eine Anfrage gewartet.

Optimierung von Modellkonfigurationen Zeitreihen in typischen Datenbanksystemen sind entlang einer Vielzahl von Dimensionen und Aggregationsebenen organisiert, auf denen traditionelle Anfragen mit Drill-Down und Roll-Up Operationen arbeiten. Die Erstellung und Wartung eines Modells für jede einzelne Zeitreihe ist extrem aufwändig und kaum praktikabel. Analog zu materialisierten Sichten können Ableitungsschemata genutzt werden, um Modelle einzusparen, beispielsweise durch Aggregation von Prognosewerten. Allerdings und damit im Gegensatz zu materialisierten Sichten haben derartige Ableitungsschemata Einfluss auf die Genauigkeit der Prognosen. Interessanterweise kann die Genauigkeit sogar durch Ableitungen erhöht werden. Beispielsweise haben empirische Studien gezeigt, dass bei verrauschten Zeitreihen, Modelle auf höherer Aggregationsebene zu bevorzugen sind, während niedrige Aggregationsebenen zu einer höheren Genauigkeit bei sehr regelmäßigen Zeitreihen führen können. In der Dissertation haben wir dazu ein verallgemeinertes Ableitungsmodell konzeptioniert, das die Ableitung einer Zielzeitreihe aus einer beliebigen Anzahl von Quellzeitreihen erlaubt.

Konfigurationsadvisor Basierend auf diesem Ableitungsmodell bestimmt der Konfigurationsadvisor für eine gegebene Anfragelast eine möglichst optimale Konfiguration von Modellen [FBL11, FSHL13]. Eine Modellkonfiguration besteht dabei aus einer Menge von Modellen und einer Menge von Ableitungsschemata, die beschreiben wie die Modelle zur Anfrageverarbeitung zu nutzen sind. Ziel des Konfigurationsadvisors ist nun zum einen die Maximierung der Prognosegenauigkeit für die gegebene Anfragelast und zum anderen die Minimierung der dafür notwendigen Modelle, d.h. der Modellwartungskosten. Während die Modellkosten einfach zu berechnen sind, lässt sich die Genauigkeit nicht mathematisch bestimmen. Hier müssen Modelle zunächst erstellt und empirisch evaluiert werden. Gerade bei großen Datensätzen ist die Erstellung aller Modelle und die Überprüfung aller möglichen Ableitungsschemata allerdings kaum praktikabel.

Der Konfigurationsadvisor arbeitet deshalb nach einem iterativen Prinzip (Abbildung 2).

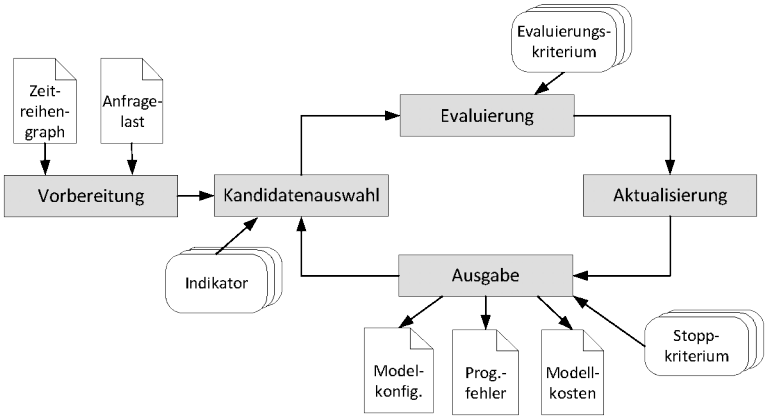

Abbildung 2: Iterative Ausführung des Konfigurationsadvisors

Er erhält als Eingabe den aktuellen Datensatz, repräsentiert als Zeitreihenhypergraph, und die Anfragelast. Nach der Erstellung einer Initialkonfiguration (*Vorbereitung*) wird eine iterative Prozesskette ausgeführt. Basierend auf Heuristiken, sogenannten Indikatoren, wird in jeder Iteration eine Menge von Modellen ausgewählt (*Kandidatenauswahl*) und für diese empirisch die Genauigkeit für die gegebenen Anfragen bestimmt (*Evaluierung*). Am Ende jeder Iteration werden die aktuellen Genauigkeiten und Wartungskosten der Modellkonfiguration ausgeben (*Ausgabe*), sodass ein automatischer oder manueller Abbruch des Advisors erfolgen könnte. Diverse Parameter (zum Beispiel die Reduktion der Kandidatenanzahl oder Ableitungsmöglichkeiten) erlauben die Skalierung des Advisors auch für größere Datensätze (*Aktualisierung*). Die Evaluierung auf realen und synthetischen Daten zeigte, dass der Advisor effizient und auch auf großen Datensätzen eine Modellkonfiguration mit hoher Genauigkeit und geringen Wartungskosten berechnet.

6 Subskriptionsbasierte Prognoseanfragen

Abschließend erweitern wir Prognoseanfragen um kontinuierliche Aspekte, die es einer Applikation ermöglichen, Anfragen einmalig im F^{2}DB System zu registrieren [FKK+12, FBLP12]. Dies erlaubt es Applikationen kontinuierliche Prognoseergebnisse zu erhalten und ermöglicht es, diese weiter zu verarbeiten. Ein wichtiger Anwendungsbereich sind beispielsweise intelligente Stromnetze, so genannte Smart Grids. Hier ist eine kontinuierliche Balancierung von Energienachfrage und -angebot notwendig, wofür kontinuierlich Notifikationen mit neuen Prognoseergebnissen an die Applikation verschickt werden müssen.

Eine subskriptionsbasierte Prognoseanfrage registriert sich am Datenbanksystem mit einem Prognosehorizont und einer Fehlergrenze. Das Datenbanksystem schickt nur dann neue Notifikationen, wenn der Prognosehorizont ausgeschöpft ist oder eine signifikante Abweichung von der Fehlergrenze festgestellt wurde. Aufgabe des *Subskriptionsdienstes* ist dabei die Optimierung derartiger Notifikationen zur Reduktion der Kommunikations-

und Applikationskosten. Dazu besteht die Möglichkeit, auch mehr Prognosewerte als gefordert, das heißt einen größeren Prognosehorizont, zu versenden. Dies hat den Vorteil, dass der Prognosehorizont länger gültig bleibt und damit weniger Notifikationen versendet werden müssen. Auf der anderen Seite zeigen sehr lange Prognosehorizonte einen höheren Prognosefehler, womit eine schnellere Überschreitung der Fehlergrenze zu befürchten ist.

Die Grundlage zur Bestimmung des optimalen Prognosehorizontes bildet ein Kostenmodell, welches die Verarbeitungskosten der Applikation abbildet. Weiterhin unterscheiden wir drei verschiedene Optimierungsansätze, die basierend auf dem entwickelten Kostenmodell und der historischen Zeitreihe einen festen Prognosehorizont, variable Prognosehorizonte für verschiedene Zeitscheiben sowie eine dynamische Anpassung des Prognosehorizontes berechnen. Die abschließende Evaluierung zeigte einen geringen Aufwand der Optimierungsansätze, eine signifikante Reduktion der Applikationskosten sowie die Gültigkeit des Kostenmodells auf realen Datensätzen.

7 Zusammenfassung

Wir haben einen Ansatz zur vollständigen Integration der Zeitreihenprognose in ein Datenbankmanagementsystem vorgestellt. Deklarative Prognoseanfragen ermöglichen dabei die einfache Abfrage von Prognoseergebnissen und erfordern keine statistischen Kenntnisse des Nutzers. Prognoseergebnisse werden direkt in der Datenbank berechnet und verarbeitet, und können mit anderen Quelldaten verknüpft werden (z.B. mittels Join-Operatoren). Analog zu klassischen Indexstrukturen werden Prognosemodelle als native Datenbankobjekte unterstützt. Dies erlaubt die transparente Speicherung, Wiederverwendung und Wartung von Prognosemodellen. Zusammenfassend haben wir im Rahmen dieser Dissertation ein neuartiges Flash-Forward Datenbanksystem entwickelt und viele weitere interessante Forschungsmöglichkeiten eröffnet. Zukünftige Forschungsarbeiten könnten beispielsweise neue Anfrageoptimierungstechniken integrieren, die Parallelisierung von Prognoseoperatoren betrachten oder die Online Wartung von Modellkonfigurationen untersuchen.

Literatur

[BJR08] George E. P. Box, Gwilym M. Jenkins und Gregory C. Reinsel. *Time Series Analysis: Forecasting and Control, 4th ed.* Wiley, 2008.

[CDD+09] Jeffrey Cohen, Brian Dolan, Mark Dunlap, Joseph M. Hellerstein und Caleb Welton. MAD Skills: New Analysis Practices for Big Data. *PVLDB*, 2:1481–1492, 2009.

[DM06] Amol Deshpande und Samuel Madden. MauveDB: Supporting Model-based User Views in Database Systems. In *SIGMOD*, Seiten 73–84, 2006.

[FBL11] Ulrike Fischer, Matthias Böhm und Wolfgang Lehner. Offline Design Tuning for Hierarchies of Forecast Models. In *BTW*, Seiten 167–186, 2011.

[FBLP12] Ulrike Fischer, Matthias Boehm, Wolfgang Lehner und Torben Bach Pedersen. Optimizing Notifications of Subscription-Based Forecast Queries. In *SSDBM*, Seiten 449–466, 2012.

[FDS+12] Ulrike Fischer, Lars Dannecker, Laurynas Siksnys, Frank Rosenthal, Matthias Boehm und Wolfgang Lehner. Towards Integrated Data Analytics: Time Series Forecasting in DBMS. *Datenbank-Spektrum*, Seiten 1–9, 2012.

[Fis14] Ulrike Fischer. *Forecasting in Database Systems*. Dissertation, Technische Universität Dresden, 2014.

[FKK+12] Ulrike Fischer, Dalia Kaulakiene, Mohamed E. Khalefa, Wolfgang Lehner, Laurynas Siksnys Torben Bach Pedersen und Christian Thomsen. Real-time Business Intelligence in the MIRABEL Smart Grid System. In *BIRTE*, Seiten 1–22, 2012.

[FL14] Ulrike Fischer und Wolfgang Lehner. Transparent Forecasting Strategies in Database Management Systems. In *Business Intelligence*. Springer Berlin Heidelberg, to appear 2014.

[FRBL10] Ulrike Fischer, Frank Rosenthal, Matthias Böhm und Wolfgang Lehner. Indexing Forecast Models for Matching and Maintenance. In *IDEAS*, Seiten 26–31, 2010.

[FRL12a] Ulrike Fischer, Frank Rosenthal und Wolfgang Lehner. F2DB: The Flash-Forward Database System. In *ICDE*, Seiten 1245–1248, 2012.

[FRL12b] Ulrike Fischer, Frank Rosenthal und Wolfgang Lehner. Sample-Based Forecasting Exploiting Hierarchical Time Series. In *IDEAS*, Seiten 120–129, 2012.

[FSHL13] Ulrike Fischer, Christopher Schildt, Claudio Hartmann und Wolfgang Lehner. Forecasting the Data Cube: A Model Configuration Advisor for Multi-Dimensional Data Sets. In *ICDE*, 2013.

[GH06] Jan G. De Gooijera und Rob J. Hyndman. 25 Years of Time Series Forecasting. *International Journal of Forecasting*, 22:443–473, 2006.

[GLW+11] Philipp Große, Wolfgang Lehner, Thomas Weichert, Franz Färber und Wen-Syan Li. Bridging Two Worlds with RICE Integrating R into the SAP In-Memory Computing Engine. *PVLDB*, 4:1307–1317, 2011.

Ulrike Fischer hat von 2003 bis 2009 Informatik an der Technischen Universität Dresden studiert. Seit 2009 ist sie dort als wissenschaftliche Mitarbeiterin am Lehrstuhl für Datenbanken tätig. Zu ihren Forschungsschwerpunkten zählen die Zeitreihenprognose, insbesondere im Energiebereich, und die Integration von analytischen Verfahren in Datenbankmanagementsysteme. Im Jahr 2013 schloss sie ihre Dissertation mit dem Titel "Forecasting in Database Systems" mit Auszeichnung ab. Sie hat weiterhin zu einer Vielzahl von Forschungsprojekten beigetragen, unter anderem dem DFG-geförderten Grundlagenprojekt FFQ (Flash-Forward Query Framework) und dem EU-geförderten MIRABEL Verbundprojekt (Micro-Request-Based Aggregation, Forecasting and Scheduling of Energy Demand, Supply and Distribution). Letzteres beschäftigt sich mit der automatisierten Balancierung moderner Energienetze, wofür effiziente und genau Energieprognosen eine wichtige Voraussetzung bilden.