

Entwurf einer IT-gestützten Balanced Scorecard zum medizinischen Qualitätsmanagement: Verwendung von UML, XML und Java

Holger Kunz, Jürgen Braun und Thomas Tolxdorff

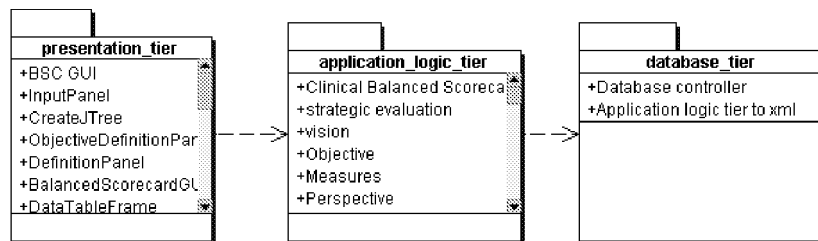
Institut für Medizinische Informatik, Biometrie und Epidemiologie
Charité – Universitätsmedizin Berlin, Campus Benjamin Franklin
Email: holger.kunz@medizin.fu-berlin.de

Zusammenfassung. Inhalt der Arbeit ist die Konzeption und Realisierung eines Systems für die computergestützte Balanced Scorecard (BSC) zur Unterstützung des Qualitätsmanagements im Krankenhauswesen. Mit Hilfe eines objektorientierten Case-Tools wurde das Software-design in der Modellierungssprache UML realisiert. Die Implementation des im Entwurf und Analyse festgelegten Designs wurde in Java umgesetzt. Die Datenbank wurde in XML realisiert sowie durch ein XML Schema definiert.

1 Einleitung

Das Gesundheitswesen steht derzeit unter enormen ökonomischen Druck, der zusätzlich durch gesetzgeberische Maßnahmen verstärkt wurde. Zur Entscheidungsunterstützung und -überwachung sowie zur Qualitätssicherung kann das von Robert Kaplan und David Norton entwickelte Konzept der Balanced Scorecard auch im Krankenhaus verwandt werden [1,2]. Es ist ein Instrument des strategischen Managements und als solches ein Kennzahlensystem, das der strategischen Kontrolle dienen soll. Das Konzept der BSC beinhaltet Entwicklung und Umsetzung einer Strategie, in der sämtliche Aktivitäten auf die vier Dimensionen Finanzen, Kunden (Patienten), Prozesse und Innovation ausgerichtet werden [3]. Aufgrund neuer Konzepte wie Managed Care, Gesundheitszentrum, vernetzte Versorgung und evidenzbasierte Medizin bietet sich die BSC als strategisches Instrumentarium auch im Krankenhaus an [4]. Außerdem belegen Untersuchungen, dass das in der USA in unterschiedlichen Branchen bewährte Konzept auch erfolgreich im Krankenhausmarkt eingesetzt werden kann [5,6]. Zur Unterstützung einer Balanced Scorecard im Krankenhaus leistet eine leistungsfähige Datenverarbeitung eine nicht zu unterschätzende Hilfestellung. Da die BSC ein mehrdimensionales Kennzahlensystem ist, soll in dieser Arbeit primär die softwaretechnische Durchführung vorgestellt werden die flexibel, plattformübergreifend, und wartungsarm ist.

Abb. 1. Die Dreischichtenarchitektur des Softwareprojektes. Hierbei erfolgt eine strikte Trennung in die Darstellungsschicht (presentation tier), die Anwendungsschicht (application tier) sowie in die Datenbankschicht (database tier).



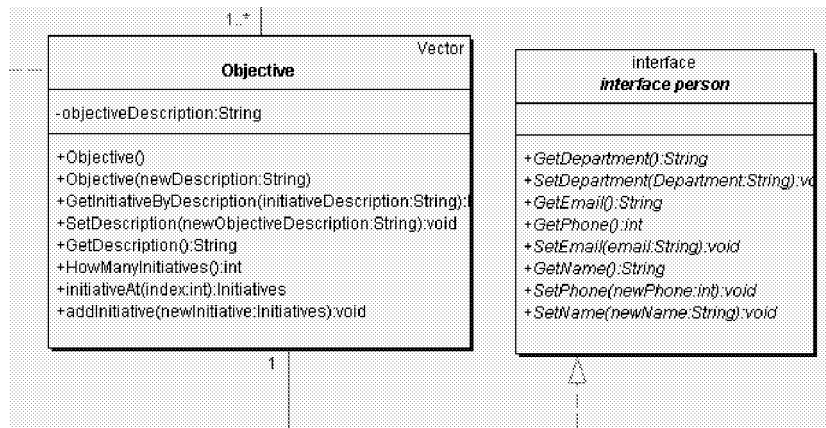
2 Methoden

Zur Erstellung eines Softwareedesigns ist zweckmäßig den objektorientierten Analyse- und Entwurfsansatz mittels UML (unified modelling language) einzusetzen. Die in der Analyse geklärten Anforderungen können mit Hilfe von Objekten im Entwurf in eine Systemstruktur umgesetzt werden. Der Entwurf bildet die Basis für die weitere Entwicklung und die anschließende Wartung. Diagramme gehören zu den wichtigsten Elementen der UML, die es erlauben, einzelne Aspekte von Modellen in einem angemessenen Detaillierungsgrad darzustellen. Dazu gehören beispielsweise das Klassen-Diagramm, Anwendungsfall-Diagramm, Reihenfolge-Diagramm, Zustands-Diagramm, Aktivitäten-Diagramm und Kooperations-Diagramm [7,8,9,10,11,12]. Ein weiterer Vorteil liegt in der breiten Unterstützung von Case-Tools, wie beispielsweise Rational Rose oder Borland Together Control Center. Zur Implementation des Entwurfs wurde Java verwendet. Java's Vorteile sind, dass es verteilt, robust, sicher, architekturneutral, multithreaded, interpretierbar, übertragbar und dynamisch ist. Insbesondere die Definition von Interfaces und Packages prädestiniert Java für die Entwicklung dieses Softwareprojektes. Da die Datenbank für zukünftige Applikationen erweiterbar, strukturiert und verifizierbar sein sollte wurde XML (extensible markup language) mit seiner XML Schema Definition gewählt. Einer der größten Stärken von XML Schemas liegt darin, dass Dokumenteninhalte auf Korrektheit validiert werden können und selbst in XML geschrieben sind [13]. Die GUI-Realisierung (graphical user interface) erfolgt mit Java Swing erfolgen.

3 Ergebnisse

Die Entwicklung des SW-Prototypen wurde nach dem Rational Unified Process (RUP) durchgeführt [14]. Er ist mittlerweile der Standard bei der Erstellung von Softwareprojekten. Er stellt eine Erweiterung des bisher genutzten Wasserfallmodells dar. Der RUP ist inkrementell, iterativ und besitzt in sich überlappende Phasen. Außerdem gibt es eine breite Tool-Unterstützung und eine starke Verknüpfung mit der UML-Notation. Er wird unterteilt in die (1) Inception, (2) Elaboration, (3) Construction und (4) Transition-Phase. Daher wurde zunächst der

Abb. 2. Ein kleiner Ausschnitt aus dem entwickelten UML-Diagramm. Links ist eine Klasse und deren zugehörige Variablen und Methoden in der UML-Notation zu sehen. Rechts sieht man beispielhaft die Definition eines Interface in der UML-Notation. Klassen, die dieses Interface implementieren, müssen entsprechend die im Interface definierten Methoden implementieren.



Softwarebezugsrahmen und die Projektgrenzen innerhalb der BSC-Anwendung festgelegt. Dabei wurden die Use-Cases des Gesamtsystems innerhalb der klinischen Anwendung erstellt. Mit Hilfe von primären und typischen Szenarios wurde das Verhalten simuliert. In der weiteren Phase wurde die Analyse des Problembereiches der Balanced Scorecard sowie die Festlegung auf die Architektur getroffen. Die Systemanforderungen wurden spezifiziert und die signifikanten Use-Cases implementiert. In der *Construction Phase* wurde die iterative und inkrementelle Entwicklung eines vollständigen und anwendungsreifen Produktes für die klinische BSC durchgeführt. Dabei wurde eine vollständige Herausarbeitung des Designs sowie Fertigstellung der Implementation erreicht. Die Entwicklung eines Klassendiagramms, Use-Case-Diagramm, Komponentendiagramms erfolgt mittels Borland Together Control Center.

Grundsätzlich wurde eine durchgehende Dreischichtenarchitektur verwendet, wie sie momentan in der Softwaretechnik üblich ist. In Abb. 1 ist die entsprechende Einteilung in die drei Packages dargestellt.

Die Konzeption zur Speicherung der BSC in objektorientierter Form als Datenbank wurde durch ein entwickeltes XML Schema durchgeführt. In der *Transition Phase* wurde eine prototypische Entwicklung einer graphischen Oberfläche durch JAVA aufgrund der UML-Konzeption realisiert. Die Erstellung der GUI wird momentan in enger Kooperation mit medizinischen Partnern durchgeführt.

4 Diskussion

Es zeigte sich sehr deutlich, dass das Softwaredesign mittels UML, eine Datenbankrealisierung mittels XML und die Verwendung der Implementationssprache Java auch im klinischen Bereich angewendet werden kann. Die Verwendung des

RUP bei der Erstellung von größeren Softwareprojekten kann vorteilhaft auch im Bereich der medizinischen Informatik eingesetzt werden. Durch die sauber definierte Struktur im Entwurf war es möglich, dass verschiedene Programmierer einzelne Komponenten entwickelten, da diese mit vorher festgelegten Schnittstellen der UML-Definition kommunizierten. Gerade im Hinblick auf heterogene Entwicklungssysteme im Krankenhaus sollte UML eine verstärkte Gewichtung zukommen. Bei der Wahl von XML zur Speicherung der Kennzahlen als Datenbankbasis zeigte sich, dass XML-basierende Datenbanken einfacher zu handhaben sind und bei geeigneter Implementation übersichtlicher als konventionelle Datenbanklösungen wie beispielsweise SQL sind. Die hohe Flexibilität des Entwurfsansatzes und die leichte Übertragbarkeit einer Implementation auf heterogene Systeme basiert auf neuen Modellierungsansätzen. Zusammenfassend lässt sich feststellen, dass die Verwendung moderner Softwareentwicklungskonzepte und die objektorientierte Implementierung hervorragend für medizinische Anwendungen genutzt werden können.

Literaturverzeichnis

1. Kaplan RS, Atkinson AA, Transforming the balanced scorecard from performance measurement to strategic management: Part I, *Accounting Horizons*, 87–104, 2001.
2. Kaplan RS, und Atkinson AA, Transforming the balanced scorecard from performance measurement to strategic management: Part II, *Accounting Horizons*, 15, 147–160, 2001.
3. Heberer M., Erfolgsfaktoren der Krankenhausführung, *Der Chirurg*, 60, Springer Verlag, 1305–1312, 1998.
4. Conrad HJ, Balanced Scorecard als modernes Management-Instrument im Krankenhaus, *Baumann-Fachverlag*, Kulmbach, 2001.
5. Rimar S. und Garstka SJ, The Balanced Scorecard, Development and Implementation in an Academic Clinical Department, *Academic Medicine* (74): ff. 114, 1999.
6. Zellman W., Issues for Academic Health Centers to Consider before Implementing a Balanced-scorecard Effort, in: *Academic Medicine* (74): ff. 1269, 1999.
7. Balzert, H. Methoden der objektorientierten Systemanalyse. *Spektrum Akademischer Verlag*, 1996.
8. Balzert H. Lehrbuch der Objektmodellierung. *Spektrum Akademischer Verlag*, 1996.
9. Booch G. *The Unified Modelling Language. User Guide* Addison-Wesley, 1999.
10. Fowler M. *Analysis Patterns: Reusable Object Models*. Addison Wesley Longman, Reading, Massachusetts 1997.
11. Oestereich B. *Objektorientierte Softwareentwicklung. Analyse u. Design mit der Unified Modeling Language* Oldenbourg 1997.
12. Quatrani T. *Visual Modeling with Rational Rose and UML*. Addison Wesley, 1998.
13. XML schema, available at <http://www.w3schools.com/schema>, 2003.
14. Rational Unified Process: Best Practices for Software Development Teams: http://www.rational.com/media/whitepapers/rup_bestpractices.pdf