

3D-MODELL-BASIERTES LEBENSDAUERMANAGEMENT VON BRÜCKEN

Michael Kluth¹, André Borrmann¹, und Till Mayer²

¹ Lehrstuhl für Bauinformatik – Technische Universität München,
{Kluth | Borrmann}@bv.tum.de

² Lehrstuhl für Baustoffkunde und Werkstoffprüfung, Technische Universität München,
TMayer@cbm.bv.tum.de

Kurzfassung: Lebensdauermanagementsysteme ermöglichen eine Betriebsoptimierung von Bauwerken über die gesamte Lebensdauer bei gleichzeitiger Verstärkung der benötigten Finanzmittel. Im Rahmen eines Forschungsprojektes entwickeln wir ein modulares Software-Tool für das Lebensdauermanagement von Brücken, dessen zentraler Bestandteil ein 3D-Bauwerksinformationsmodell ist. Das Bauwerk wird in bis zu fünf Detaillierungsgrade untergliedert, wodurch eine sehr genaue Erfassung und Einschätzung des Bauwerkszustandes ermöglicht wird. Die Implementierung des Software-Tools erfolgt als Java-Applikation, die Brückengeometrie wird mittels Java 3D dargestellt. Für eine intuitive und nutzerfreundliche Interaktion mit der Brückengeometrie im Benutzer-Frontend wird ein Algorithmus vorgestellt, durch den wechselnde Zentren der Rotation durch Selektion einzelner Bauteile ermöglicht wird.

1 Einführung

1.1 Motivation

In Deutschland gibt es einen Bestand von ca. 120.000 Brücken [1]. Die Bauwerksbetreiber stehen vor der Aufgabe, diesen immer älter werdenden Bauwerksbestand mit immer stärker begrenzten Haushaltsmitteln zu erhalten [2]. Bereits vor mehreren Jahren wurden computerunterstützte Bauwerksmanagementsysteme (BMS) entwickelt, die die Ingenieure bei der Instandhaltungsplanung unterstützen sollen. Bei vielen dieser Systeme werden Instandsetzungsmaßnahmen erst dann geplant, wenn Schädigungen visuell bei Bauwerksprüfungen erkannt werden. Der optimale Zeitpunkt einer Instandsetzung liegt aber in vielen Fällen bereits vor dem Zeitpunkt, an dem die Schä-

digung visuell erkannt wird. Durch Anwendung von vollprobabilistischen Schädigungsmodellen kann die Zustandsentwicklung von Bauteilen prognostiziert und durch visuelle sowie ergänzende zerstörungsfreie Bauwerksuntersuchungen zugeschräfft werden. Mit einem solchen prädiktiven Lebensdauermanagementsystem (LMS), welches diese Modelle impliziert, ist eine Optimierung des Betriebes über die gesamte Lebensdauer des Bauwerks möglich [2]. Ein solches System unterstützt den Bauwerksbetreiber bei der langfristigen Planung von Inspektionen und Instandsetzungen auf Netzebene, wodurch auch eine Verstetigung der benötigten Finanzmittel erreicht werden kann.

Im Rahmen eines aktuellen Forschungsvorhabens entwickeln wir eine Software für das prädiktive Lebensdauermanagement von Stahlbetonbrücken. Im Zentrum steht das 3D-Bauwerksinformationsmodell, welches alle Informationen über Messungen, Aufnahmen und Zustandsprognosen enthält und somit dem Bauwerksbetreiber jederzeit ein Überblick über den Bauwerkszustand gewährt. Der Aufbau dieses LMS und die Umsetzung in ein Software-Tool soll nach einem Überblick über bestehende Systeme dargestellt werden.

1.2 Literaturrecherche

Im Rahmen einer Literaturrecherche wurden einige bestehende Lebensdauermanagementsysteme für Brücken identifiziert. In Deutschland wurde das System „SIB-Bauwerke“ von der Bundesanstalt für Straßenwesen entwickelt, welches auf Bundes- und Landesebene eingesetzt wird [3]. Die Landeshauptstadt Düsseldorf hat ein eigenes System entwickelt, welches für die Instandsetzungsplanung von Brücken und Tunneln im Stadtgebiet Anwendung findet [4]. Im Ausland werden beispielsweise die folgenden Systeme verwendet: Bridgelife (Finnland) [5], Danbro (Dänemark) [6], Eirspan (Irland) [7], Kuba-MS (Schweiz) [8], Pontis [9] und Bridgit [10] (USA) sowie das Ontario Bridge Management System (Kanada) [11]. In Kanada wird zurzeit ein „mobile model-based bridge lifecycle management“ entwickelt [12].

Im Folgenden sollen einige Aspekte ausgeführt werden, die bestehende Systeme von dem von uns zu entwickelnden System unterscheiden:

- Bei den meisten der genannten Systeme wird keine explizite Geometrie verwendet. Eine Ausnahme bildet [12].
- Der Level of Details-Ansatz für die Strukturierung des Bauwerks ist unterschiedlich ausgebildet, wobei bei allen genannten Systemen keine Untergliederung in Unterbauteile und HotSpots erfolgt.

- Die Zustandsprognose wird entweder über deterministische Modelle (beispielsweise [3], [4], [6]) oder über Markov-Ketten abgebildet (zum Beispiel [5]). Vollprobabilistische Verfahren für bekannte Schädigungsmechanismen werden nicht angewendet.
- Die Zustandserfassung erfolgt bei den untersuchten Systemen visuell ohne Verwendung von zerstörungsfreien Mess- und Prüfverfahren.

2 Prädiktives Lebensdauermanagementsystem

2.1 Aufbau

Der Aufbau des von uns zu entwickelnden prädiktiven Lebensdauermanagementsystems ist in Abb. 1 dargestellt.

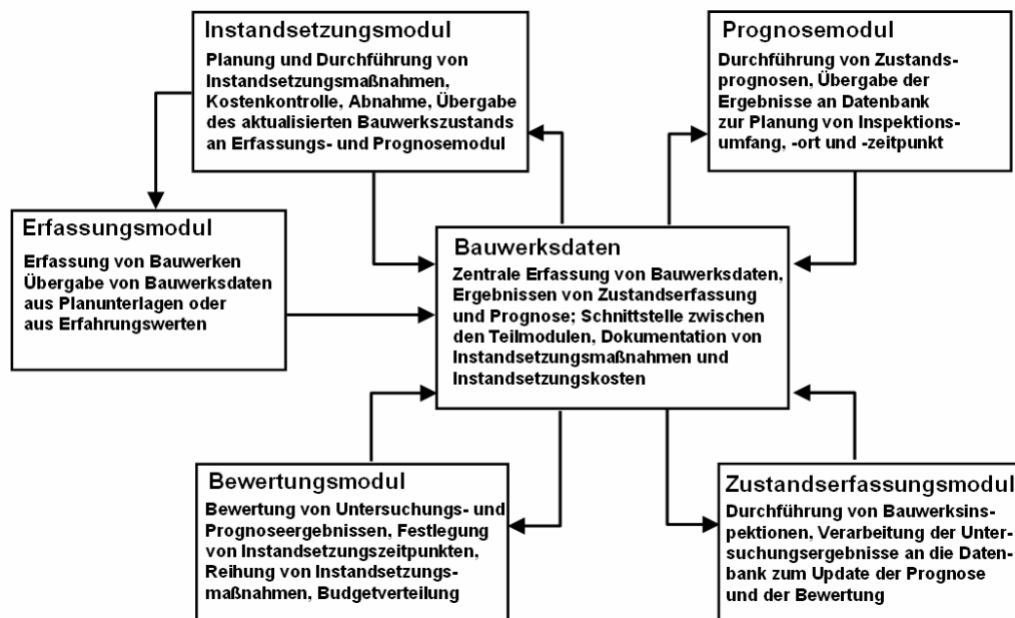


Abb. 1: Aufbau des prädiktiven Lebensdauermanagementsystems [2]

Im Mittelpunkt des LMS steht eine Datenbank, in der alle Bauwerksdaten erfasst werden. Hierzu zählen allgemeine Informationen und Geometriedaten eines Bauwerks sowie Angaben zu den charakteristischen Eigenschaften der verwendeten Baustoffe, Ergebnisse von Messungen und Instandsetzungen. Die Datenbank stellt gleichzeitig die Schnittstelle zu allen anderen Modulen des LMS dar. Über das Erfassungsmodul können weitere Bauwerke in das LMS integriert werden.

Mit dem Prognosemodul werden anhand vollprobabilistischer Schädigungsmodelle unter Berücksichtigung der Ergebnisse zerstörungsfreier Messverfahren die zeitlichen Zustandsänderungen der Bauteile bestimmt. Da derzeit nur für die Modellierung einiger zentraler Schädigungsmechanismen (z.B. Bewehrungskorrosion) ausreichend quantifi-

zierte Schädigungsmodelle existieren, muss zur Abbildung anderer Mechanismen (z.B. Frost-Tausalz-Angriff, AKR) vorerst auf einfache Modellierungsansätze wie Markov-Kettensysteme zurückgegriffen werden. Diese können jedoch zu einem späteren Zeitpunkt durch geeignete Schädigungsmodelle ersetzt werden. Die Prognosen dienen als Grundlage für die Planung von Inspektionen. Messwerte, die im Laufe von Inspektionen aufgenommen werden, werden über das Zustandserfassungsmodul den entsprechenden Bauteilgeometrien zugeordnet. Anschließend kann unter Berücksichtigung der Untersuchungsergebnisse ein Update der Prognosen durchgeführt werden. Das Bewertungsmodul ermittelt auf Grundlage der aktualisierten, prognostizierten Zustandsänderungen den optimalen Instandsetzungszeitpunkt und die benötigten Instandsetzungsmaßnahmen. Die Abnahme einer Instandsetzung erfolgt durch das Instandsetzungsmodul, so dass der aktualisierte Bauwerkszustand wiederum dem Prognosemodul zur Verfügung gestellt wird.

Ein weiterer Bestandteil sind Kataloge für Materialien, Expositionsklassen, Messverfahren, Instandsetzungsverfahren, Schädigungsmodelle und Bauteile. Diese Listen sollen ein effektives Arbeiten mit dem Software-Tool ermöglichen. Weiterhin werden Methoden zur Verfügung gestellt, damit der Anwender diese Kataloge jederzeit selbständig erweitern kann.

2.2 3D-Bauwerksmodell

Zentraler Bestandteil für die Datenhaltung und Datenerfassung innerhalb des LMS ist das 3D-Bauwerksmodell. In diesem Modell werden sowohl die Geometrie der Bauteile als auch alle Informationen über Materialeigenschaften, Inspektionen, Instandsetzungen und Zustandsänderungen mit Bezug auf die entsprechende Geometrie abgelegt. Bei allen geometrieabhängigen Informationen mit Zeitbezug werden in der Datenbank auch Datum und Uhrzeit vorgehalten.

Für eine möglichst genaue Zuordnung von Messungen, Schädigungen, Instandsetzungen usw. zu Elementen des Bauwerksmodells erfolgt eine Untergliederung des Bauwerks. Hierfür werden bis zu fünf Detaillierungsgrade (Levels of Detail, LoD) verwendet, die in Abb. 2 dargestellt sind [13].

Ein Bauwerk (Ebene 1) wird zunächst in Module (Ebene 2) untergliedert, wobei diese Untergliederung nach funktionalen oder organisatorischen Gesichtspunkten vorgenommen wird [13]. Für eine Brücke bieten sich die Module Gründung, Widerlager, Pfeiler und Überbau an. Innerhalb eines Moduls werden zunächst Bauteile (Ebene 3) identifiziert, die wiederum in Unterbauteile (Ebene 4) gegliedert werden. Diese Untergliederung erfolgt in Abhängigkeit von unterschiedlichen Einwirkungen oder Material-

widerständen. Jedes Unterbauteil kann weiterhin HotSpots (Ebene 5) aufweisen. HotSpots bezeichnen Bereiche, in denen lokal geringe Materialwiderstände oder hohe Einwirkungen auftreten. Diese Bereiche können vom Tragwerksplaner oder Bauwerksbetreiber vorgegeben werden, oder sie ergeben sich infolge von Untersuchungen, wenn lokal Änderungen der Einwirkungen während der Nutzungsdauer festgestellt werden [13]. Nur auf den Ebenen 3 bis 5 werden den Bauwerkselementen Geometrieinformationen sowie Materialeigenschaften und Bauteilwiderstände zugewiesen.

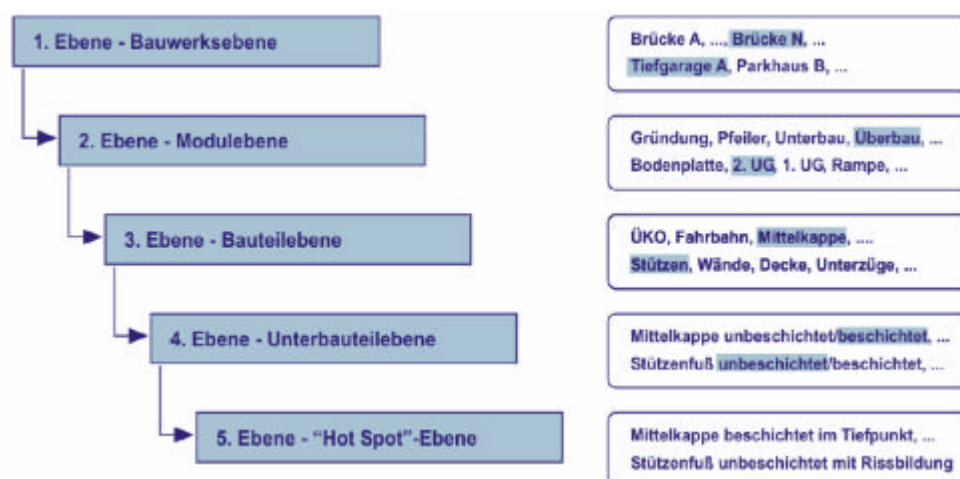


Abb. 2: Detaillierungsgrade für die Bauwerkserfassung [13]

Alle in der Datenbank vorgehaltenen Bauwerke bilden die Netzebene. Inspektionen und Instandsetzungen können auf diese Weise auf Bauwerks- oder Netzebene geplant werden.

Die Zustandsbestimmung des Gesamtbauwerks erfolgt durch Aggregation der Zustände der einzelnen Bauwerksebenen. Dieser Levels of Detail-Ansatz ermöglicht dem Bauwerksbetreiber eine sehr genaue Erfassung und Einschätzung des Bauwerkszustandes, die vom HotSpot bis hin zum Gesamtbauwerk auf Netzebene durchgeführt werden kann.

3 Umsetzung des LMS in ein Software-Tool

3.1 Allgemeines

Die Implementierung des Systems erfolgt durch Kopplung einer Java-Applikation [14] mit einer relationalen Datenbank, in der die Bauwerksinformationen inklusive Bauwerksgeometrie abgespeichert werden. Als Datenbankmanagementsystem wurde eine relationale MySQL-Datenbank gewählt [15], die dreidimensionale Darstellung der Geometrie wird über die Java 3D-Bibliothek umgesetzt [16]. Ein Screenshot der Applikation ist in Abb. 3 dargestellt.

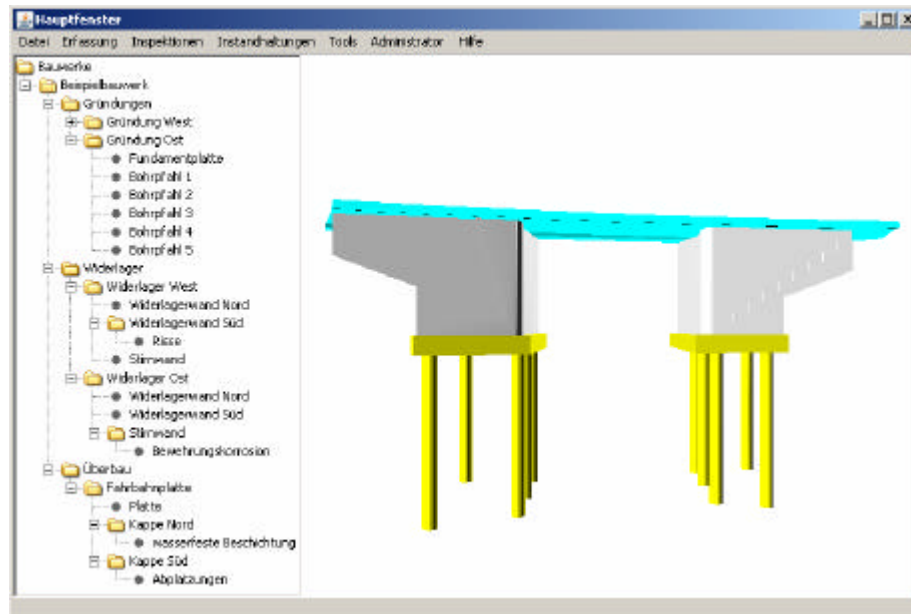


Abb. 3: Oberfläche des Software-Tools (in Entwicklung)

Die Geometriedaten werden in Form eines Boundary Representation Modells (B-Rep) in der Datenbank abgelegt. Hierfür wird eine Datenstruktur verwendet, die an den Modellierkern ACIS angelehnt ist [17].

Für die Verwendung der Programmiersprache Java spricht vor allem die Portierbarkeit auf andere Betriebssysteme, da bei den potenziellen Anwendern dieses Software-Tools (Verwaltungen, Baufirmen mit PPP-Projekten) sowohl Windows- als auch Linux-Systeme im Einsatz sind.

Obwohl es bereits erste Schritte zur Entwicklung eines Produktdatenmodells für Brücken gibt (IFC-Bridge, [18]), steht bisher noch kein solches Modell für den praktischen Einsatz zur Verfügung. Eine spätere Verwendung eines solchen Produktdatenmodells ist vorgesehen.

Alle Bauwerksinformationen, die in der Datenbank abgespeichert sind, können über das grafische Benutzer-Frontend abgerufen und bearbeitet werden. Das Frontend ist derart gestaltet, dass eine intuitive und nutzerfreundliche Interaktion sowohl im Büro als auch während Brückeninspektionen gewährleistet werden kann.

3.2 Darstellung des 3D-Bauwerksmodells

3.2.1 Einführung in die Java 3D-Bibliothek

Die dreidimensionale Darstellung des Bauwerksmodells wird mit der Java 3D-Bibliothek umgesetzt [16]. Zunächst soll aufgezeigt werden, wie ein Szenegraph in dieser Bibliothek aufgebaut ist (siehe Abb. 4).

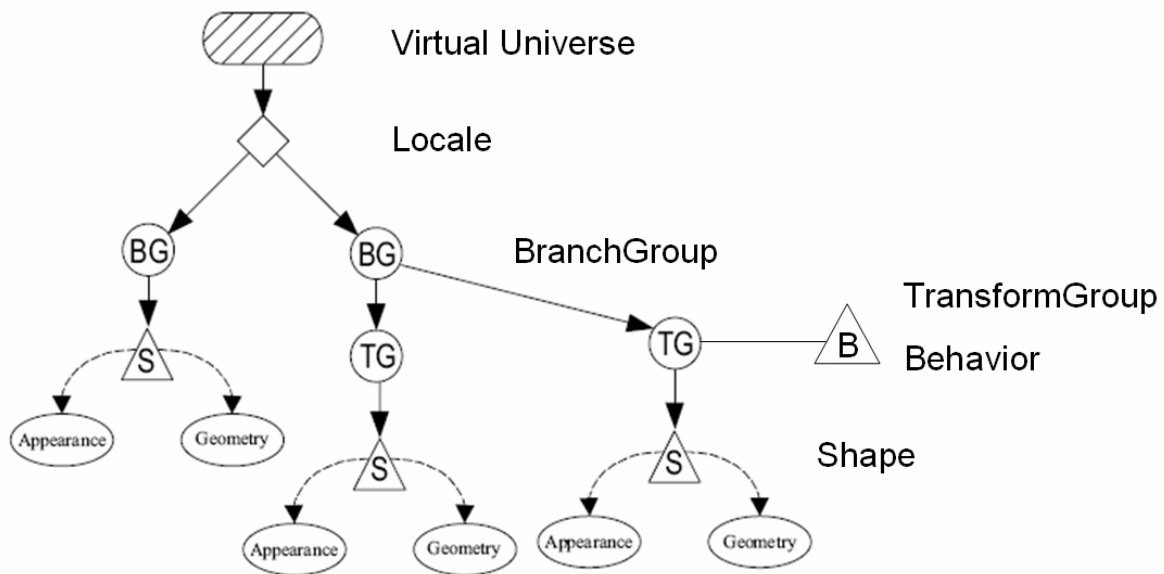


Abb. 4: Beispiel eines Szenegrafen in Java 3D bestehend aus drei Objekten, aus [19]

Die Java-Klassen *Virtual Universe* und *Locale* stellen Methoden zur Verfügung, damit Geometrien dreidimensional im Benutzer-Frontend angezeigt werden können. Alternativ kann auch die Klasse *SimpleUniverse* verwendet werden. Ein *Shape* enthält einerseits die Geometrie der Objekte (*Geometry*) und andererseits deren Eigenschaften wie beispielsweise Linienstärke, Farbe und Textur. Letztere Informationen werden mittels einer Instanz der Klasse *Appearance* den geometrischen Objekten zugewiesen. Durch Anwendung von *TransformGroups* auf ein *Shape* können Rotationen und Translationen durchgeführt werden. Mehrere Körper werden über eine *BranchGroup* zu einem Objekt zusammengefasst. Einer *TransformGroup* können Instanzen der Klasse *Behavior* zugeordnet werden, wodurch weitere Eigenschaften aller Elemente, die unterhalb dieser *TransformGroup* angeordnet sind, festgelegt werden. In der Java 3D-Bibliothek sind bereits *Behaviors* implementiert, die beispielsweise Rotationen, Translationen und Skalierungen von Java 3D-Objekten mittels Mausbewegung erlauben.

3.2.2 Rotation mit wechselnden Rotationszentren

Für eine nutzerfreundliche und intuitive Anwendung des Software-Tools ist von großer Bedeutung, dass sowohl Skalierungen und Translationen als auch Rotationen des Brückenmodells im grafischen Benutzer-Frontend ermöglicht werden. Der Anwender kann entweder über den TreeView oder direkt über das Java 3D-Grafikfenster jeweils ein Element des Bauwerks anwählen, dessen geometrischer Mittelpunkt dann zum Pol für Rotationen wird.

Die Problematik bei der Umsetzung von wechselnden Rotationszentren bei der Verwendung der Java 3D-Bibliothek und ein entsprechender Algorithmus für die Umsetzung sollen abschließend dargestellt werden.

Bei dem in Abb. 5 dargestellten Szenegraphen entspricht der Pol für Rotationen aller Objekte dem geometrischen Mittelpunkt des Objektes 1. Rotationen per Mausklick werden ermöglicht, in dem der *TransformGroup* TG_1 ein *MouseRotateBehavior* zugewiesen wird. Zur Visualisierung der dargestellten *BranchGroup* im Benutzer-Frontend, muss die Methode *compile()* auf die Instanz angewendet werden. Nach diesem Methodenaufruf kann die Struktur dieser *BranchGroup* nicht mehr verändert werden. Dies hat zur Folge, dass bei einem Wechsel des Rotationszentrums eine neue Instanz der Klasse *BranchGroup* erstellt werden muss.

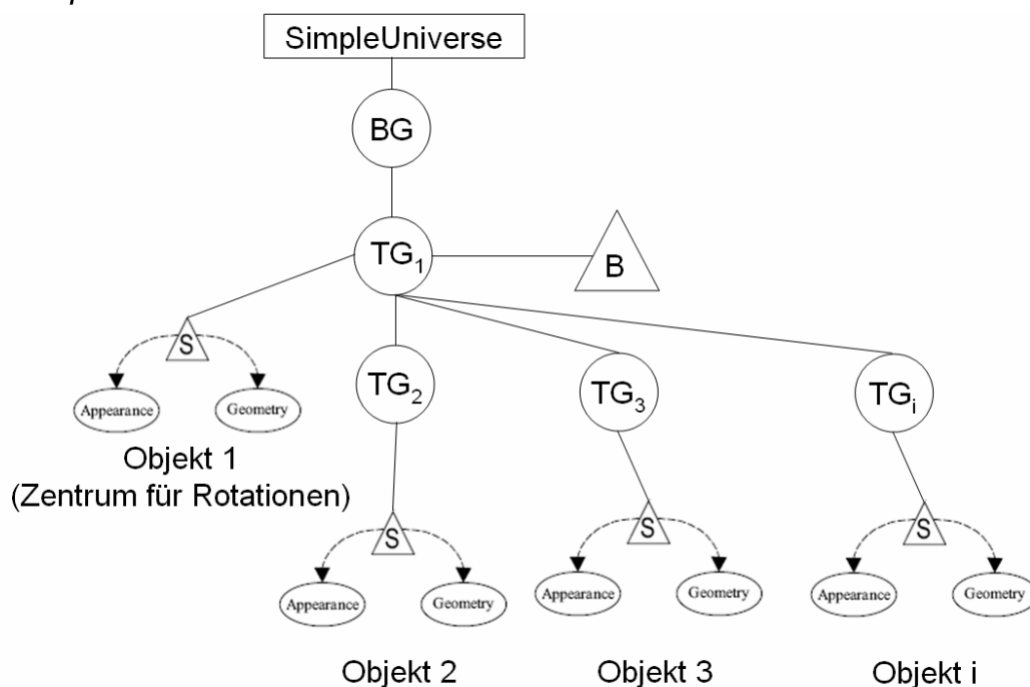


Abb. 5: Aufbau des Szenegraphen im LMS

Es seien n Objekte in der *BranchGroup* enthalten. Objekt k sei das zuvor ausgewählte Objekt (Objekt 1 in Abb. 5) und Objekt m das neu angeklickte Objekt (beispielsweise Objekt i in Abb. 5). Damit der Anwender den Aufbau der neuen *BranchGroup* nicht bemerkt, muss die Lage aller Objekte im Grafikfenster in der neuen Instanz identisch zur Lage in der alten Instanz sein. Hierfür sind die folgenden Operationen durchzuführen:

- Bei einem Mausklick auf Objekt m muss geprüft werden, ob gilt

$$\text{Objekt } k \neq \text{Objekt } m$$

- Wenn diese Bedingung erfüllt ist, wurde ein anderes Objekt angeklickt. Zunächst wird die *TransformGroup* TG_1 der *BranchGroup* bestimmt. Aus TG_1 wird die gl-

bale Transformationsmatrix \underline{M}_{TG_1} extrahiert, die die Rotationen und Translationen des Gesamtsystems enthält.

- Es werden die aktuellen Koordinaten der geometrischen Mittelpunkte aller n Objekte nach Gleichung (1) berechnet und lokal gespeichert.

$$\vec{v}_{j,neu} = \underline{M}_{TG_1}(\vec{v}_j - \vec{v}_k) + \vec{v}_k \quad \forall j \in n \quad (1)$$

Es bedeuten: $\vec{v}_{j,neu}$ Vektor mit den neuen Punktkoordinaten des Mittelpunkts von Objekt j ; \vec{v}_j , \vec{v}_k Vektor mit den zuletzt berechneten Punktkoordinaten des Mittelpunkts von Objekt j bzw. Objekt k .

- Die alte *BranchGroup* wird mittels der Methode *detach()* aus dem Szenegraph entfernt.
- Es wird eine neue Instanz der Klasse *BranchGroup* erstellt, in der die Objekte k (Objekt 1 in Abb. 5) und m (Objekt i in Abb. 5) vertauscht platziert werden. Alle Objekte werden wiederum an die *TransformGroup* TG_1 angehängt. Alle invarianten Rotationen und Translationen werden in den jeweiligen untergeordneten *TransformGroups* abgelegt. Nur bei Objekt k werden diese in der *TransformGroup* TG_1 mit abgespeichert.

Diese Vorgehensweise ist immer dann durchzuführen, wenn entweder im Grafikfenster oder im TreeView ein Objekt angeklickt wird.

Mit diesem Algorithmus ist es ebenfalls möglich, bei Veränderungen der Bauwerksgeometrie (zum Beispiel Anlegen eines neuen HotSpots) die Geometrieänderungen im Grafikfenster anzuzeigen, ohne dass der Betrachtungswinkel des Anwenders verändert wird.

4 Zusammenfassung

Im Rahmen eines Forschungsvorhabens entwickeln wir ein Software-Tool für das prädiktive Lebensdauermanagement von Brückenbauwerken aus Stahlbeton. Ein wesentlicher Bestandteil ist das 3D-Bauwerksinformationsmodell, in welchem alle Informationen über die einzelnen Bauteile geometriebezogen enthalten sind. Für eine intuitive und nutzerfreundliche Interaktion mit dem dreidimensionalen Bauwerksmodell wurde ein Algorithmus vorgestellt, mit dessen Hilfe wechselnde Rotationszentren in einer Szenegraph-basierten 3D-Applikation umgesetzt werden können.

5 Danksagung

Die Ergebnisse dieses Artikels entstanden im Rahmen des BMBF-Verbundforschungsvorhabens „Nachhaltig Bauen mit Beton“, Teilprojekt D „Lebensdauermanagement“. Das Projekt wird vom BMBF gefördert.

Literaturverzeichnis

- [1] Bericht über die Qualität, Dauerhaftigkeit und Sicherheit von Spannbetonbrücken. Bundesministerium für Verkehr, Bau und Stadtentwicklung, Berlin 2006.
- [2] P. Schießl, T. Mayer: Lebensdauermanagement von Stahlbetonbauwerken. In: Deutscher Ausschuss für Stahlbetonbau im DIN e.V. (DAfStB) (Hrsg.). Statusseminar zum Verbundforschungsvorhaben „Nachhaltig Bauen mit Beton“. Berlin 2006.
- [3] P. Haardt: Entwicklung eines Bauwerks-Management-Systems für das deutsche Fernstraßennetz, Stufe 1 und 2, Schlussbericht zum AP-Projekt 99 245. Bundesanstalt für das Straßenwesen. Bergisch Gladbach 2002.
- [4] Lebensdauermanagement für Ingenieurbauwerke der Landeshauptstadt Düsseldorf
http://www.duesseldorf.de/verkehrsmanagement/bruecken_und_tunnel/index.shtml
- [5] E. Vesikari: BRIDGELIFE, User Manual
- [6] A. Henriksen: Bridge Management – Routine Maintenance: Recent Experience with the Routine Management Module in the DANBRo Bridge Management System. In: Proceedings of the 8th International Bridge Management Conference, Denver, USA 1999.
- [7] National Roads Authority: Development of Eirspan: Ireland's bridge management system
<http://nra.ie/RepositoryforPublicationsInfo/file,2164,en.pdf>
- [8] KUBA Benutzerhandbuch
<http://www.astra.admin.ch/dienstleistungen/00129/00183/00518/index.html?lang=de>

- [9] W. Robert, A. Marshall, R. Shepard, J. Aldayuz: Pontis Bridge Management System – State of Practice in Implementation and Development. In: Proceedings of the 9th International Bridge Management Conference, Orlando, USA 2003.
- [10] H. Hawk: BRIDGIT: User-Friendly Approach to Bridge Management. In: Proceedings of the 8th International Bridge Management Conference, Denver, USA 1999.
- [11] P. Thompson, T. Merlo, B. Kerr, A. Cheetham, R. Ellis: The New Ontario Bridge Management System. In: Proceedings of the 8th International Bridge Management Conference, Denver, USA 1999.
- [12] A. Hammad, C. Zhang, Y. Hu: Mobile Model-Based Bridge Lifecycle Management System. Computer-Aided Civil and Infrastructure Engineering 21 (2006), pages 530-547.
- [13] P. Schießl, T. Mayer: Forschungsbericht 31006/05, DAfStB-Verbundforschungsvorhaben „Nachhaltig Bauen mit Beton“, Teilprojekt A3 „Wesentliche Bausteine eines Bauwerksmanagementsystems“. München 2007.
- [14] Java, Sun Microsystems
<http://java.sun.com>
- [15] MySQL, MySQL GmbH
<http://www.mysql.de>
- [16] Java 3D, Sun Microsystems
<http://java.sun.com/products/java-media/3D/>
- [17] 3D Acis Modeler, Spatial
<http://www.spatial.com/products/acis.html>
- [18] N. Yabuki, E. Lebegue, J. Gual, T. Shitani, L. Zhantao: International Collaboration for Developing the Bridge Product Model "IFC-Bridge". In: Proceedings of the Joint International Conference on Computing and Decision Making In Civil and Building Engineering, Montréal, Canada 2006.
- [19] D. Bouvier: Getting Started with the Java 3D™ API
<http://java.sun.com/developer/onlineTraining/java3d/>