

VISION EINES MITWACHSENDEN GEOMETRIEMODELLS FÜR DIE COMPUTERGESTÜTZTE BAUAUFNAHME

14. FORUM BAUINFORMATIK, BOCHUM

Prof. Dr.-Ing. Dirk Donath, Dipl. Inf. Torsten Thurow
Bauhaus-Universität Weimar

***Kurzfassung:** Die aktuelle Situation im Bauwesen ist dadurch geprägt, daß der überwiegende Teil an Bauleistungen in enger Beziehung zu der bestehenden Bausubstanz realisiert wird. Die Voraussetzung für die Planung sind exakte Planungsunterlagen. Die Praxis der computergestützten Bauaufnahme zeigt, daß die meisten Applikationen Adaptionen der Bauplanung entstammen, in erster Linie das Bauaufmaß unterstützen und sich meist auf ein Aufmaßverfahren beschränken. Eine durchgängige Unterstützung des Aufnehmenden von der Erstbegehung bis zur Übergabe des erstellten Geometriemodells an die betreffenden Fachplaner erfolgt nicht. Die Autoren haben die Vision eines mitwachsenden Geometriemodells für die computergestützte Bauaufnahme, welches den Aufnehmenden von der Erstbegehung bis zum Datenexport begleitet.*

1. Der Charakter der Bauaufnahme

1.1 Der Charakter der Bauaufnahme

Die Voraussetzung für die Planung im Bestand sind exakte Planungsunterlagen. In der Regel stellen diese, falls sie überhaupt vorliegen, jedoch nur eine ungenaue und / oder unvollständige Basis für alle weiteren Untersuchungen dar. Aufgrund dieser Tatsache muß eine Bestandsaufnahme durchgeführt werden, die entweder eine Ergänzung, Prüfung oder vollständige Neuanlage der Planungsunterlagen zum Ziel hat.

Obwohl Umfang und Ablauf einer Bauaufnahme sehr stark von ihrem Verwendungszweck und der zu erfassenden Bausubstanz abhängig sind, bildet die Abbildung der Geometrie der Bausubstanz in der Regel einen wichtigen Bestandteil der Aufnahme.

Der allgemeine Ablauf einer Bauaufnahme kann wie folgt beschrieben werden: Ein Architekturbüro, das auch Bauaufnahmen durchführt oder sich auf diesen Bereich spezialisiert hat, wird vom Kunden kontaktiert. Aufgrund dessen Angaben zu Gebäude und Ziel der Bauaufnahme erstellt das Büro ein Angebot. Wird dieses angenommen, kann die Bauaufnahme beginnen.

Vorhandene Pläne des Gebäudes bilden einen guten Startpunkt. Die Pläne werden inhaltlich geprüft, nicht eingezeichnete Öffnungen oder Wände etc. ergänzt, nicht mehr vorhandene Elemente gelöscht. Vor allem aber werden entsprechend der Aufgabenstellung Maße der entsprechenden Genauigkeit aufgenommen. Vor allem das Handmaß spielt dabei eine große Rolle, verbunden mit tachymetrischen Messungen und in einigen Fällen photogrammetrische Aufnahmen. Als Ergebnis erwarten die Auftraggeber je nach Fall Grundrisse und Schnitte als CAD-Zeichnungen oder CA(A)D-Modelle, aber auch Flächen- oder Mengenkalkulationen.

Die Idee

Gerade die Frühphase der Bauaufnahme ist von einer skizzenbasierten Annäherung des Aufnehmenden an das aufzunehmende Bauwerk geprägt. In dieser Phase ist der Umfang der aufzunehmenden Informationen in der Regel noch nicht geklärt. Der Aufnehmende wird besonders in dieser Phase mit dem Unerwarteten konfrontiert. Die Skizze ist dabei bis heute das Werkzeug schlechthin. Mit ihr lassen sich selbst abstrakteste Zusammenhänge sehr anschaulich Widerspiegeln. Die Skizze ist dabei nicht an bestimmte Modelle oder festdefinierte Symboliken gebunden, sondern lebt geradezu von ihrer Dynamik. Sollen bis dahin nicht festgelegte Objekte in der Skizze erfasst werden, so können diese bildlich, stilistisch oder symbolisch repräsentiert werden, wobei es dem Aufnehmenden selbst überlassen ist, neue Wiedergabeformen einzuführen. Selbstverständlich existieren daneben umfangreiche Standards der Wiedergabe bestimmter Objekte.

Die flexiblen Möglichkeiten der Gestaltungsmöglichkeiten von Skizzen haben allerdings auch ihren Preis. Eine Interpretation von Skizzen ist in großen Teilen nur im Kontext mit entsprechendem Hintergrundwissen möglich. Eine automatische Interpretation von Skizzen durch Computer ist nur bedingt möglich und bildet den Schwerpunkt einer eigenen Wissenschaftsdisziplin.

Trotz der beschriebenen Widrigkeiten halten die Autoren es durchaus für möglich, das eine Einbettung der Skizzenerstellung in den Frühphasen der Bauaufnahme vorteilhaft

sein könnte. Der Rechner erlaubt im Gegensatz zur Skizze auf Papier eine beliebig oft durchführbare Manipulation von Skizzen und kann ihre Verwaltung unterstützen. Für die Verwaltung der Skizzen werden Kennzeichner eingeführt. Ein Kennzeichner hat einen Typ und einen Namen. Typ und Name sind dabei flexibel wählbar. Jedem Typ eines Kennzeichners kann ein Symbol zugeordnet werden. Symbole sind dabei flexibel erstellbar.

Erstellen einer Skizze

Zur Erstellung einer Skizze ruft der Nutzer das Icon „Neue Skizze“ auf. Das System bietet eine Auswahl von Skizzentypen an, die vom Nutzer beliebig erweitert werden kann. Der Nutzer wählt einen Typ und bekommt dabei gleichzeitig einen Namen angeboten, welcher vom ihm beliebig veränderbar ist, solange der Name noch nicht im System existiert.

Der Nutzer kann nun mit dem Pen beliebig auf der Skizzenfläche zeichnen. Er kann dabei Strichart, Dicke und Farbe verändern. Außerdem kann er Beschriftungen einführen.

In einem Übersichtsbaum werden alle neu angelegten Skizzen an den Startknoten gehangen. Der Nutzer kann sie von dort strukturiert im Baum ablegen. Es ist dabei auch möglich, eine Skizze an mehreren Knoten zu referenzieren. Die Darstellung des Knotens der Skizze ändert sich dabei in die Form eines Links, bis wieder nur noch eine Referenz existiert. Wird auch diese gelöscht, so wird auch die Skizze gelöscht. Eine weitere Möglichkeit ist die Nutzung von Ordnern, ähnlich den Ordnern im Dateisystem, um Gruppen von Skizzen zu bilden.

Alle diese Funktionen bis auf die Typvergabe der Namen ließen sich bisher auch mit einem Dateieexplorer und einem besseren Grafikprogramm lösen. In diesem Fall jedoch ermöglichen Typ und Name Abfragen nach Datenbankschema.

Nun sollen auch Elemente der Skizzen miteinander in Beziehung gebracht werden. Dazu können Gruppen der grafischen Elemente, also der Striche und Beschriftungen selektiert und einem Kennzeichner zugeordnet werden. Dazu lassen sich neue Kennzeichner generieren oder vorhandene auswählen. Gleiches gilt für ihre Typen analog der Kennzeichnung der Skizzen allgemein. Einem Kennzeichnertyp kann ein Symbol zugeordnet werden. Ein Symbol ist wieder eine Gruppe von Zeichnungselementen, quasi wie eine Miniaturskizze. Dieses Symbol ist Standardsymbol für jeden Kennzeichner dieses Typs, kann aber bei Bedarf für jeden Kennzeichner modifiziert werden. Beim Einfügen eines Kennzeichners in eine Skizze kann das Symbol des Kennzeichners und sein Name angezeigt werden, muß jedoch nicht. Die Position des Symbols und des Namens sind beliebig veränderbar. Änderungen des Namens des Kennzeichners, sei

nes Symbols oder des Namens seines Typs wirken sich dabei global auf alle Referenzen aus.

Alle Kennzeichner werden ebenfalls in dem oben beschriebenen Übersichtsbaum angezeigt und lassen sich ebenso vom Nutzer strukturiert umsortieren.

Durch die Kennzeichner ist es möglich, zwischen den Skizzen zu navigieren. Angedacht ist desweiteren die Verknüpfung zwischen Kennzeichnern und Objekten in späteren Schritten.

IT-internes Konzept zur dynamischen Bauteildefinition

Wird eine Person gebeten, eine verbale Beschreibung eines Raumes oder einer Etage zu geben, so wird sie dazu in der Regel Substantive wie „Räume“, „Etage“, „Wände“, „Stützen“, „Decke“ etc. benutzen. Der Mensch gliedert demnach unvoreingenommen Bauwerke nach räumlichen und bauteilorientierten Gesichtspunkten. Inhalt dieses Artikels ist die bauteilorientierte Sichtweise auf ein Bauwerk und ein vorgeschlagenes Konzept zur IT-internen Abbildung.

Die Grundelemente einer bauteilorientierten Sicht sind trivialerweise Bauteile. Bauteile besitzen verschiedene Eigenschaften, wie Funktion, Material, Kenngrößen und Geometrie. Beispielsweise besitzt eine Stütze immer eine Höhe. Verschiedene Bauteilarten bilden Bauteilgruppen, wie beispielsweise Stützen, zu denen runde, quaderförmige, komplexe Stützen usw. gehören können.

Lange Zeit wurde das Objektorientierte Paradigma als das quasi Allheilmittel der IT-Modellierung angesehen, nur langsam setzt sich die Erkenntnis durch, das auch das OOP nur ein Paradigma unter vielen mit seinen Stärken und Schwächen darstellt. In der Tat scheint das OOP grundsätzlich für die IT-interne Modellierung von Bauteilen sehr gut geeignet.

Die wesentlichsten Elemente des OOP's sind Klassen und Objekte. Dabei ist die Vorstellung, das alle Objekte exakte Instanzierungen ihrer Klassen darstellen. Um auf diese Weise reale Bauteile zu modellieren, müssen diese parametrisiert werden. Bei diesem Vorgang findet naturgemäß immer eine Idealisierung statt. Problematisch an dieser Idealisierung ist, das in der Realität beispielsweise die Geometrie eigentlich gleicher Stützen eben nicht genau gleich ausfallen kann, und das nicht nur in der Maßlichkeit. Nehmen wir an, in einem Bauwerk wurde eine große Zahl von baugleichen Stützen verwendet. Zwei dieser Stützen mußten im Einbau nachbearbeitet werden. Bei einer Stütze wurde ein Durchbruch eingefügt, um einen Kabelkanal durchzuführen, bei der anderen wurde an der Seite ein Einschnitt vorgenommen, da hier ein Längsträger zu weit nach außen ragte. Gehören diese beiden modifizierten Stützen nun einer neuen

Stützenklasse an ? Die Antwort auf diese Frage dürfte bei unterschiedlichen Lesern unterschiedlich beantwortet werden.

Klassenkonzept

Das Objektorientierte Paradigma wird heute geradezu als Allheilmittel zur IT-mäßigen Abbildung realer Vorlagen angesehen. Das Paradigma geht dabei von den Grundelementen Klasse und Objekt aus. Verschiedene Sprachen, wie Java oder C++ unterstützen dieses Paradigma auf der Ebene der Programmiersprache. Klasse und Objekt sind hier Begriffe während der Programmerstellung, das fertige Programm kennt diese Begriffe nicht. Der Nutzer kommt mit den vom Programmierer genutzten Klassen und Objekten nicht direkt in Berührung. Er kann auch keine neuen Klassen erstellen.

Das Objektorientierte Paradigma läßt sich aber genauso gut auf Nutzerseite einsetzen. Beispielsweise in der objektorientierten Nutzerführung. Der Nutzer selektiert ein Objekt und wählt danach eine der Methoden, die von dem Objekt angeboten werden. Auf programmiertechnischer Seite kann eine solche Applikation auch mit Programmiersprachen erstellt werden, welche nicht objektorientiert sind, etwa mit Basic. Das Paradigma drückt sich hier nur in der Oberfläche aus.

In den letzten Jahren wurde viel Arbeit in die Erstellung universeller Bauwerksmodelle investiert. Die meisten dieser Bauwerksmodelle haben den Neubau als Zielvorgabe. Bis heute ist kein allumfassendes Bauwerksmodell entstanden und wird aus Sicht der Autoren auch nicht entstehen. Zu unterschiedlich und vielseitig fällt Architektur in der Realität aus.

Die Autoren versuchen einen Ansatz, dem Nutzer die Möglichkeit zu geben, sein eigenes Bauwerksmodell zu erstellen oder zu modifizieren. Dazu muß er in der Lage sein, selber neue Bauteilklassen und ihre Instanzen generieren zu können. Zu diesem Zweck wird ein Modellverwaltungssystem genutzt. Ein Modellverwaltungssystem stellt quasi die Umsetzung des Objektorientierten Paradigmas auf API-Ebene dar. Zur Laufzeit können Klassen mit ihren Attributen erstellt, modifiziert und instanziiert werden. Diese Funktionalität kann durch eine geeignete Oberfläche an den Nutzer weitergegeben werden.

Bei der Verwendung eines Modellverwaltungssystems ergeben sich verschiedene Fragen. Die im restlichen Programm fest kompilierten Algorithmen müssen mit den dynamischen Klassen und Objekten des MVS verbunden werden und existieren vor ihnen, nämlich zur Compilezeit.

Das Objektorientierte Paradigma stellt eine Idealisierung der Realität dar. Alle Instanzen der Klassen sind 100%ige Abbilder der Klassen. In der Realität bestehen oft Unterschiede zwischen formal gleichen Bauteilen. Als Beispiel sei eine größere Gruppe gleicher Stützen genannt. Die Stützen erhielten teilweise Durchführungen für Kabelkanäle

bzw. Einschnitte an sehr engen Stellen. Halterungen wurden angeschweißt, Seitenarme teilweise gekürzt bzw. umgearbeitet. Dieses Beispiel soll verdeutlichen, das die Frage, wann Bauteile noch der selben Bauteilklasse angehören und wann nicht mehr, gar nicht so einfach zu entscheiden ist.

An dieser Stelle soll das Objektorientierte Paradigma an mehreren Stellen „aufgeweicht“ werden. Anstatt einer festen parametrisierten Geometrie wird eine auf Objektebene flexibel änderbare Geometrie verwendet und dynamisch mit parametrisierten, festen Objektattributen wie Länge, Breite, Höhe verbunden. Die Geometrie wird quasi zu einem Attribut.

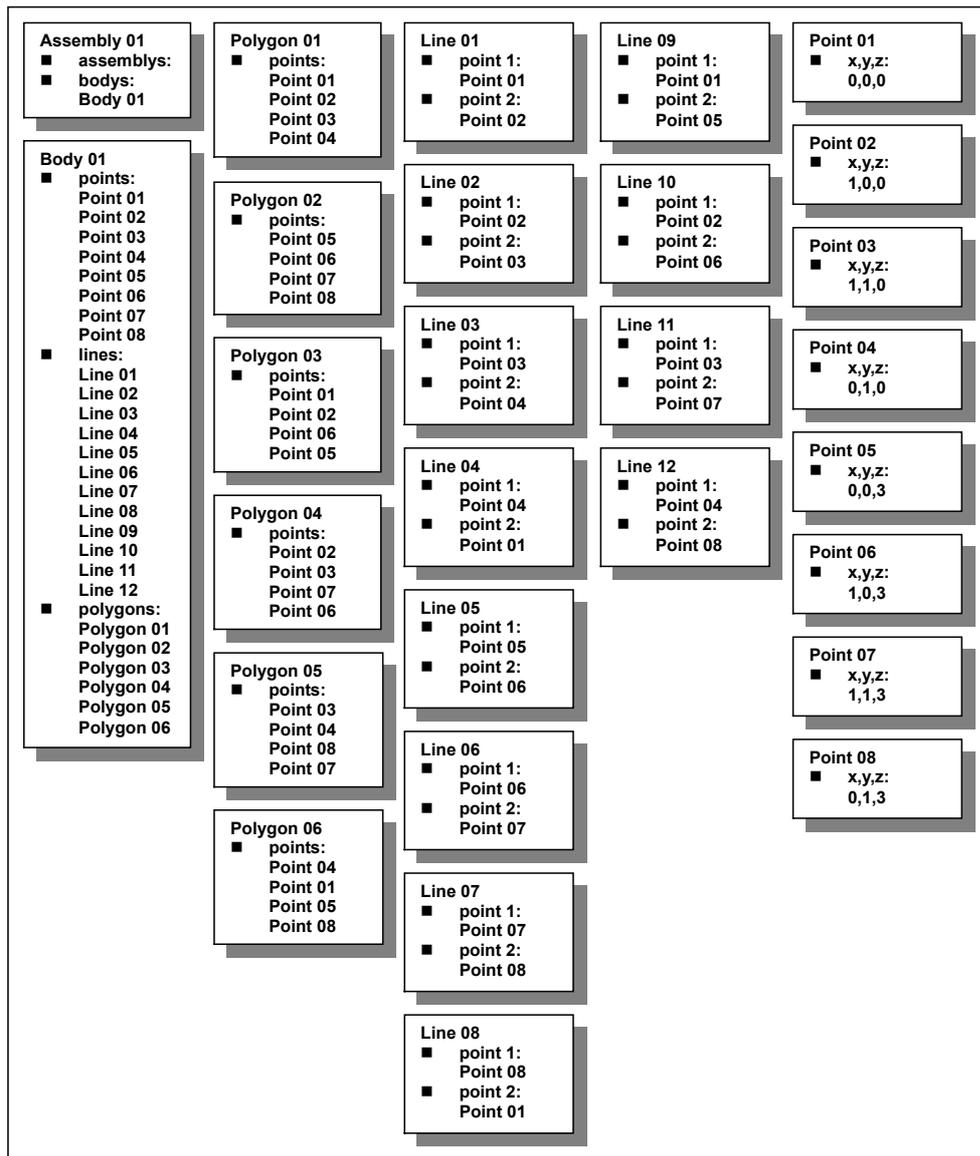


Abbildung 1: Beschreibung eines Quaders

Die Umsetzung dieser Grundidee ist denkbar einfach. Der Nutzer erstellt eine neue Bauteilklasse und weist ihr ein von ihm angelegtes oder modifiziertes BRep-Modell zu (Abbildung 1). Werden von der Bauteilklasse Instanzen gebildet, so wird bei jeder Instanziierung eine Kopie des BRep-Modells angelegt und der Instanz zugewiesen. Auf diese Weise erhalten alle Instanzen zunächst eine identische Geometrie, die aber im weiteren Verlauf beliebig verändert werden kann (Abbildung 2).

Die dynamische Erzeugung und Modifikation von Klassen durch den Nutzer zur Laufzeit bringt neue Probleme mit sich, die in der klassischen Programmierung so nicht bekannt sind. So ist es nun möglich, dass eine Klasse verändert wird, nachdem schon Instanzen von ihr generiert wurden, deren Laufzeit noch nicht beendet ist. Änderungen der Klasse müssen daher in geeigneter Weise auf die Instanzen übertragen werden.

Änderungen an existierenden Klassen können sein:

- Hinzufügen von Attributen
- Ändern der Defaultwerte der Attribute
- Löschen von Attributen
- Umbenennung von Attributen
- Ändern von Attributtypen

Änderungen von Attributtypen entsprechen dem Löschen und Neuanlegen von Attributen unter gleichen Namen, jedoch mit anderem Typ. Die folgenden Betrachtungen gelten der Übertragung der genannten Modifikationen auf die Geometrie.

Auf Geometrielevel können die einzelnen geometrischen Primitiven wie Punkte, Linien, Polygone etc. mit Attributen verglichen werden. Diese besitzen jedoch keinen Bezeichner. Von daher verbleiben die Operationen:

- Hinzufügen von geometrischen Elementen
- Ändern von geometrischen Elementen
- Löschen von geometrischen Elementen

Bei dem Hinzufügen geometrischer Elemente in der Defaultgeometrie einer Bauteilklasse ist es in einem großen Teil der Anwendungsfälle wünschenswert, diese Ergänzung auch in den schon existierenden Instanzen der Bauteilklasse vorzunehmen. Zu diesem

Zweck ist es günstig, jeder Instanz einer Bauteilklassse eine Tabelle mitzugeben, die es ermöglicht, geometrische Elemente der Klassendefinition und der Instanzen zuzuordnen. Auf diese Weise lassen sich Änderungen in den Geometriedefinitionen der Bauteilklassen auf die Instanzen übertragen. Diese Zuordnung wird am Beispiel eines Quadrates erläutert.

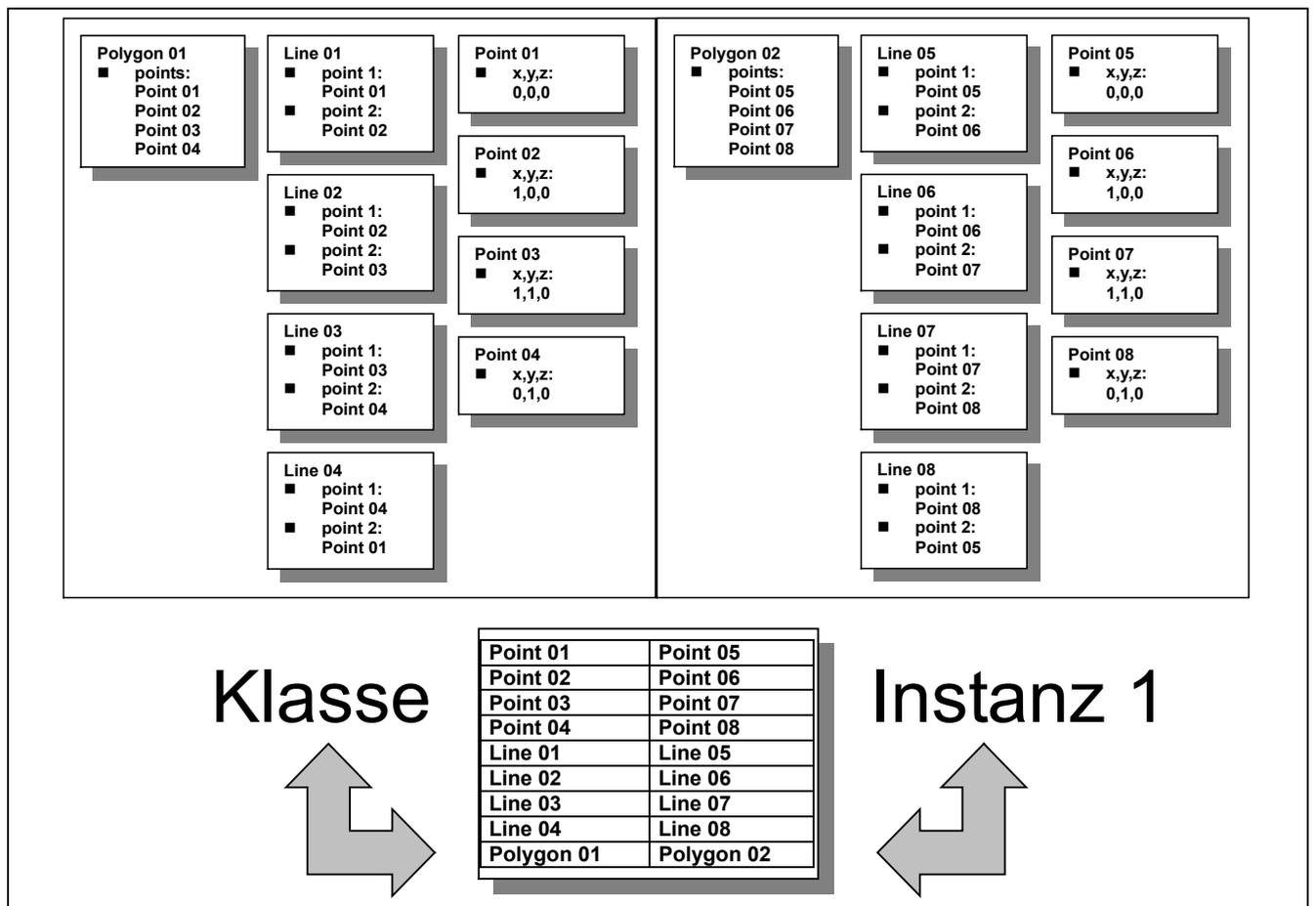


Abbildung 2: Beziehungen zwischen Geometriedefinitionen Klasse - Instanz

Abbildung 2 zeigt die Defaultgeometrie einer Bauteilklasse und ihre Kopie, welche einer Instanz dieser Klasse zugeordnet wurde. Alle Elemente der beiden Geometrien sind in einer Tabelle zugeordnet. Nun wird in der Defaultgeometrie der Klasse dem Quadrat eine Diagonale hinzugefügt. Anhand der Verweise aus der Tabelle der Instanz 1 kann diese entsprechend selbst verändert werden (Abbildung 3).

Das Löschen von Attributen ist entsprechend trivial. Wird ein geometrisches Element in der Defaultgeometrie gelöscht, so werden in den Geometrien der Instanzen die zugehörigen Elemente ebenfalls gelöscht. Die Zuordnung wird den Tabellen entnommen. Änderungen der Geometrie erfolgen analog.

Kopplung von Geometrie und Parametern

Mit Hilfe verknüpfter BRep-Modelle lassen sich vielfältige Geometrien realer Bauteile wirklichkeitsnah abbilden. Für bestimmte Anwendungszwecke werden aber auch formalisierte Werte wie Höhe, Breite, Tiefe benötigt. Der Nutzer muß in die Lage versetzt

werden, Kopplungen zwischen Geometrie und Attributen einzurichten, welche die Attribute entsprechend füllen und aktualisieren. Zu diesem Zweck werden Leseattribute eingeführt.

Leseattribute sind Elemente im Geometriemodell, die Verweise auf Attribute besitzen und bei Änderungen an der Geometrie dorthin Werte schreiben. Solch ein Leseattribut kann die Distanz zwischen zwei Punkten sein. Das Leseattribut wird topologisch gesehen mit den beiden distanzbestimmenden Punkten verknüpft. Ändert sich die Position mindestens eines der beiden Punkte, so wird die Distanz zwischen beiden Punkten neu bestimmt und in ein Attribut eingetragen. Die Mechanismen zur Kopie der Leseattribute zwischen Defaultgeometrie der Bauteilklassse auf die neuen Geometrien der Instanzen geschieht wie bei normalen geometrischen Elementen analog dem vorherigen Kapitel.

Der eben beschriebene Weg geht davon aus, dass ein für den jeweiligen Verwendungszweck ausreichend exaktes BRep-Modell gegeben ist, welches nur noch ausgelesen werden muß. In anderen Fällen ist es jedoch

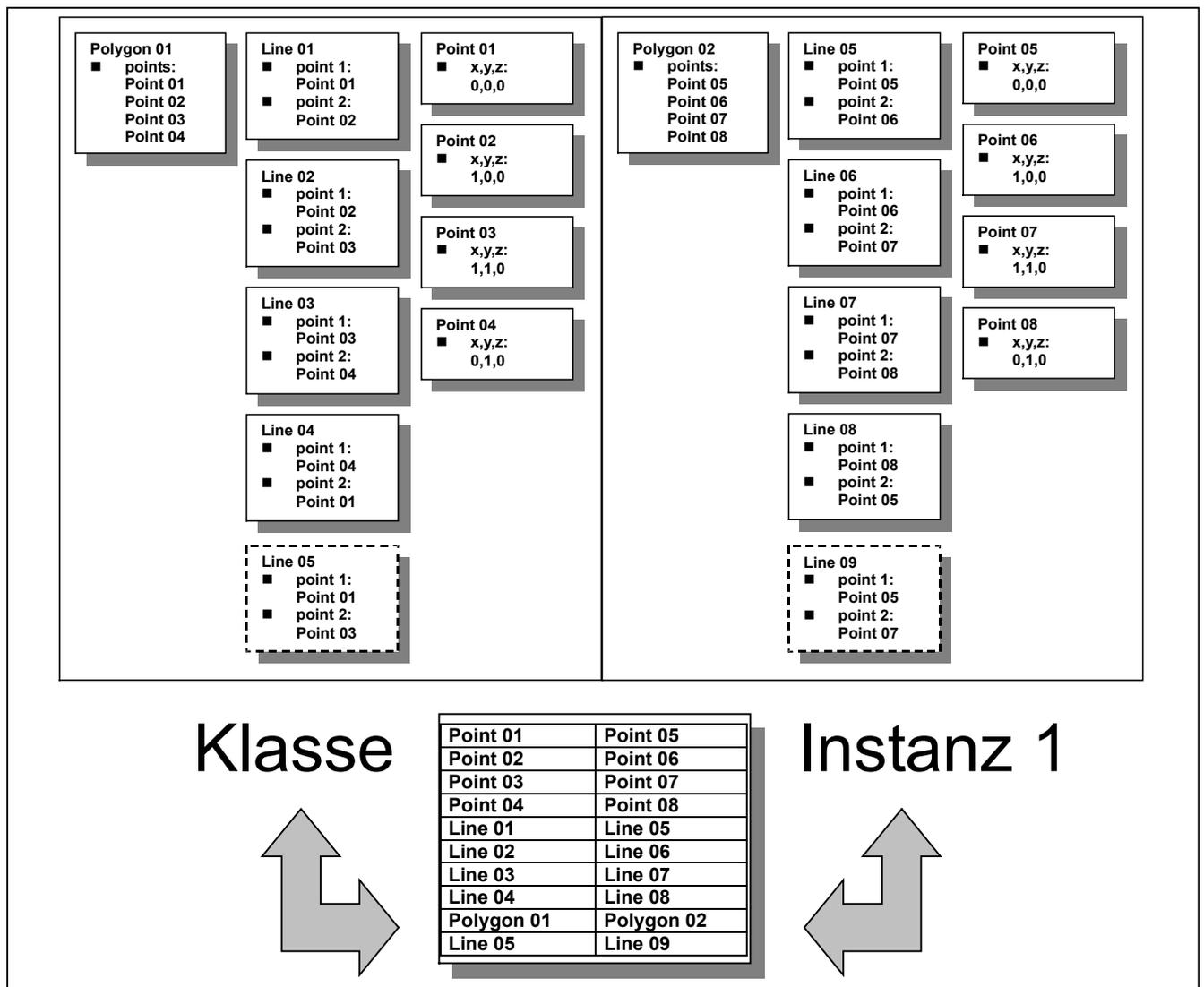


Abbildung 3: Ergänzung der Geometrie

wünschenswert, Parameter aus Attributen auf das BRep-Modell zu übertragen, beispielsweise, um bei einem Quader dessen Höhe festzulegen.

Das Gesamtkonzept der Autoren geht von einem BRep-Modell aus, welches auch zur Aufnahme von realen Meßwerten dient, um das BRep-Modell maßlich immer weiter der Realität anzunähern. Diese Messungen müssen mit der Übernahme von Attributwerten, welche die Geometrie betreffen, abgeglichen werden.

Für die beschriebene Aufgabe eignet sich die geodätische Ausgleichsrechnung. Die Messungen werden als Beobachtungen und die Attributwerte als Bedingungsgleichungen modelliert. Bei Unterbestimmtheit durch Datums- und Konfigurationsdefekte muß entsprechend regularisiert werden.

Referenz und Kopie

Oftmals sind in Bauwerken eine große Anzahl nahezu identischer Bauteile zu finden, wie etwa eine lange Fensterreihe oder eine Menge von im gleichmäßigen Raster eingebauten Stützen. In vielen Fällen ist hier ein Aufmaß aller vorgefundenen Bauteile nicht erforderlich. Vielmehr wird ein Bauteil oder eine kleine Auswahl der Bauteile aufgenommen und für die restlichen Bauteile als gleichbleibend erklärt. An dieser Stelle wird im folgenden zwischen Kopien und Referenzen unterschieden. Wird ein Bauteil zum Referenzobjekt erklärt, so erfolgt bei Änderungen an diesem Bauteil auch die entsprechende Änderung an den ihm zugeordneten Bauteilen, während einfache Kopien keine Bindungen mehr an ihr Ursprungsbauteil besitzen.

Referenzierungen können sowohl auf Geometrie- wie auf Attributebene geschehen. So wie die Geometrie eines Fensters in Form und Maßlichkeit referenziert werden kann, trifft dies auch auf Aussagen wie Farbe, Anstrich, Öffnungsrichtung etc. zu.

Objekte, die einem Referenzobjekt zugeordnet sind, stellen keine hundertprozentige Kopie dar. So befinden sich beispielsweise mehrere Fenstern nicht an der selben Position, ihre „Einfügepunkte“ sind verschieden. Gleiches gilt für Attribute, wie beispielsweise die fortlaufende Bezeichnung der Fenster („Fenster 1.OG, Raum 2, Nr. 3“). Die Referenzierung gilt demnach nur für bestimmte Attribute.

Es ist demnach eine Regelung erforderlich, welche Attribute referenziert werden. Zwei Ansätze sind denkbar:

1. Es wird in der Implementation fest vorgegeben, welche Attribute referenziert werden.
2. Die Steuerung geschieht auf Nutzerebene.

Einer festen implementativen Lösung steht entgegen, das zur Compilezeit die vom Nutzer angelegten Bauteilklassen unbekannt sind. In diesem Fall können lediglich bestimmte Attribute als referenziert festgelegt werden, oder es wird eine Komplimentehre Lösung gewählt, in der festgelegt ist, welche Attribute nicht referenziert werden.

Eine Lösung auf Nutzerebene kann durch Einführung von Flags zu jedem Attribut erfolgen. Je nach dem, ob das Flag gesetzt oder nicht gesetzt wird, wird das Attribut referenziert oder nicht. Auf Klassenebene werden günstige Standartwerte der Flags vergeben, so das die Flags der Instanzen nur im Bedarfsfall vom Nutzer geändert werden müssen.

Etwas komplexer als die Referenzierung auf Attributebene fällt die Referenzierung der Geometrie aus. Hier müssen maßliche oder strukturelle Änderungen geeignet weitergegeben werden. Strukturelle Änderungen lassen sich dabei analog der Mechanismen bei

Änderungen der Defaultgeometrien von Bauteilklassen verarbeiten. Die Zuordnung der Geometrielemente ist geklärt, da alle Instanzen, sowohl Referenzobjekt wie zugeordnete Objekte, über Zuordnungstabellen nach Abbildung 2 verfügen. Schwierig in diesem Sinne sind maßliche Änderungen, da diese nicht absolut, sondern relativ weitergegeben werden müssen.

noch unfertig !