

CityGML Quality Interoperability Experiment des OGC

VOLKER COORS¹ & DETLEV WAGNER²

Zusammenfassung: Die Bedeutung der Datenqualität von 3D-Stadtmodellen nimmt mit der Komplexität von Analyseaufgaben immer mehr zu. Die Prüfung der Daten hinsichtlich gewünschter Anforderungen an deren Struktur und Beschaffenheit ist dazu unabdingbar. Das CityGML Quality Interoperability Experiment soll die Spezifikation von Anforderungen vereinheitlichen sowie die Vergleichbarkeit von Prüfergebnissen und Prüfroutinen ermöglichen.

1 Einleitung

Zunehmend komplexere Analyseanwendungen im Bereich dreidimensionaler Stadtmodelle erfordern einen zuverlässigen und einheitlich spezifizierten Dateninput, um verlässliche und aussagekräftige Analyseergebnisse zu erzielen. Dadurch hat die Bedeutung des Datenqualitätsmanagements in den letzten Jahren auch im Bereich der 3D-Stadtmodelle stark zugenommen. Klare Richtlinien für Datenhersteller und -nutzer zur Festlegung der Anforderungen an ein CityGML-Modell werden nötig. Das CityGML Quality Interoperability Experiment (CityGML QIE) hat sich zum Ziel gesetzt, Vorschläge zur Spezifikation von Anforderungen, zur Strukturierung des Prüfprozesses und des Prüfergebnisses zu machen. Dabei wird eine auf Konsens basierende Standardisierung und Vereinheitlichung des Vorgehens angestrebt, die best-practice-Empfehlungen zur Anforderungsspezifikation, Validierung von CityGML-Modellen und dem Fehler-Reporting erarbeiten soll.

2 Methodik

Wird ein CityGML Dokument mit verschiedener Prüfsoftware geprüft, dann ist es unbedingt erforderlich, dass die Prüfergebnisse unabhängig von der eingesetzten Prüfsoftware sind. Ansonsten ist eine Vergleichbarkeit der Prüfergebnisse nicht gegeben. Grundvoraussetzung für eine Qualitätsprüfung muss also sein, dass definierte Prüfkriterien, nach denen ein CityGML Dokument geprüft wird, bei jeder Prüfsoftware zum gleichen Prüfergebnis führen. Um dies sicherzustellen zu können, müssen zunächst einheitliche Prüfkriterien definiert werden. Diese Prüfkriterien ergeben sich im CityGML QIE aus den im CityGML Standard definierten Anforderungen an ein CityGML Dokument.

Definiton 1. Ein CityGML Dokument d ist ein wohlgeformtes XML-Dokument, das der im CityGML-Standard definierten Schema-Struktur folgt. Die Menge aller CityGML-Dokumente wird mit D bezeichnet.

¹ Hochschule für Technik Stuttgart, Schellingstr. 24, 70174 Stuttgart;
E-Mail: volker.coors@hft-stuttgart.de

² University of Tehran, College of Engineering, Department of Surveying and Geomatics Engineering, Kh. Karagar Shomali, 14399-57131 Teheran, Iran; E-Mail: dwagner4@gmx.at

Definiton 2. Eine Anforderung r (requirement) ist ein überprüfbares Kriterium, das etwas über den Inhalt eines CityGML Dokuments bzw. die darin beschriebenen Daten aussagt. Die Menge aller Anforderungen wird mit R bezeichnet.

Als Beispiel einer Anforderung sei hier genannt, dass, wenn es sich bei der Geometrie eines Gebäudes (Building) um ein Solid-Element handelt, die Geometrie der Solid-Definition nach ISO 19107 entsprechen muss. Die Anforderungen werden in 2.1 detailliert betrachtet.

Definiton 3. Eine Prüfmethode $check_r: D \rightarrow \text{Boolean}$ prüft, ob ein gegebenes CityGML Dokument eine bestimmte Anforderung r erfüllt, d.h.

$$check_r(d) = \begin{cases} \text{true, falls } d \text{ die Anforderung } r \text{ erfüllt} \\ \text{false, sonst} \end{cases}$$

Die Menge aller Prüfmethode wird mit C bezeichnet.

Der Algorithmus, mit dem die Prüfmethode implementiert wird, wird dabei nicht vorgegeben. Natürlich sollen verschiedene Implementierungen einer Prüfmethode für alle CityGML Dokumente zum selben Ergebnis führen.

Definiton 4. Die Implementierung einer Teilmenge C^A von C in einer Softwareumgebung wird als Prüfsoftware A bezeichnet.

Definiton 5. Zwei verschiedene Prüfsoftwares A und B implementieren die Teilmengen C^A und C^B der Prüfmethode. A und B sind interoperabel, wenn für jede Prüfmethode $check_r \in C^A \cap C^B$ gilt, dass die Implementierung der Prüfmethode $check_r^A$ in Prüfsoftware A und die Implementierung derselben Prüfmethode $check_r^B$ in Prüfsoftware B für alle CityGML Dokumente dasselbe Ergebnis liefern:

$$\forall check_r \in C^A \cap C^B, d \in D: check_r^A(d) = check_r^B(d)$$

Um ein aussagekräftiges Fehlerprotokoll zu ermöglichen und im Hinblick auf automatisierte Korrektur eines CityGML Dokuments, das bestimmte Anforderungen nicht erfüllt, wird als Ergebnis einer Prüfmethode oft nicht nur ein boolescher Wert ausgegeben, sondern ein Prüfergebnis mit zusätzlichen Informationen, inwieweit die Anforderung verletzt wurde (WAGNER et al. 2014b).

2.1 Definition der Anforderungen

Zur Strukturierung der Anforderungen werden im CityGML QIE die folgenden fünf Kategorien festgelegt:

- Schema
- Geometrie
- Semantik
- Konformitätsregeln
- Referenzielle Integrität

Die Kategorie Schema enthält alle Anforderungen an das CityGML Dokument, die sich aus dem CityGML Schema ergeben. Dazu gehört, dass das Dokument wohlgeformt und valide sein muss. Eine weitere Anforderung ist, dass die *gml:id* eines Elements eindeutig sein muss.

Anforderungen an die im CityGML Dokument beschriebene Geometrie sind im Wesentlichen die Planarität von Polygonen und die „Wasserdichtigkeit“ von Solids. Diese sind bereits ausführlich in WAGNER et al. (2013) und LEDOUX (2014) diskutiert.

In der Kategorie Semantik werden sowohl die Anforderungen die Elementstruktur im CityGML Modell als auch die Anforderungen an Attribute zusammengefasst. Im Vordergrund stehen hierbei Attribute und Elemente, die abhängig von der Geometrie sind. Hier sei exemplarisch das Attribut *measuredHeight* eines Gebäudes genannt.

Die Konformitätsregeln beziehen sich auf die im CityGML Standard definierten sog. *conformance requirements*. Häufig hängen die Konformitätsregeln mit der Semantik zusammen. Als Kriterium zur Abgrenzung dieser Kategorie gilt folgende Festlegung: Ist eine Anforderung durch eine Konformitätsregel im CityGML-Standard festgelegt, wird sie dieser Kategorie zugeordnet. Ist dies nicht der Fall, wird die Anforderung einer anderen Kategorie zugeordnet.

Anforderungen an die im CityGML Dokument verwendeten internen und externen Referenzen werden der Kategorie Referenzen zugeordnet. Externe Referenzen lassen sich ohne Zugriff auf einen externen Datenbestand in der Regel nicht überprüfen. Daher sind externe Referenzen nicht Bestandteil des CityGML QIE.

Jede Anforderung wird durch folgende Parameter beschrieben:

- Eindeutiger Name
- Informelle Beschreibung
- Formale Definition
- Ausnahmen
- Vorbedingungen, d.h. Liste von Anforderungen, die erfüllt sein müssen
- CityGML Version
- Prüfergebnis, siehe oben

Der Name der Anforderung wird durch die Kategorie der Anforderung, das CityGML Element, auf das sich die Anforderung bezieht und eine eindeutige Kennung innerhalb der Kategorie zusammengesetzt:

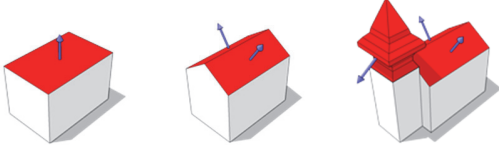
XX-namespace:YY-ZZZZ

Dabei steht XX für die Kategorie (SC – Schema, GE – Geometrie, SE – Semantik, CR – Konformitätsregeln, RI – Referenzielle Integrität).

Das CityGML Element, auf das sich diese Anforderung bezieht, wird durch den code YY beschrieben. Gültige Werte sind beispielsweise AB (*AbstractBuilding*), BU (*Building*), BP (*BuildingPart*), BS (*BoundarySurface*), RS (*RoofSurface*), WS (*WallSurface*), GS (*GroundSurface*) und PO (*Polygon*). Da einige Elemente wie *BoudarySurface* in verschiedenen CityGML-Modulen vorkommen, wird zur Unterscheidung der in CityGML übliche Namensraum des jeweiligen Moduls vorangestellt. Der dritte Teil des Namens (ZZZZ) ist eine eindeutige Kennung innerhalb der jeweiligen Kategorie.

Die Anforderung SE-bldg:RS-FN ist demnach eine Anforderung der Kategorie Semantik, die sich auf *RoofSurface*-Elemente im Modul Building bezieht. Eine vollständige Beschreibung der Anforderung ist in Tabelle 1 gegeben.

Tabelle 1: Anforderung SE-bldg:RS-FN

Parameter	Wert
ID	SE-bldg:RS-FN
Beschreibung	Die Flächennormalen aller Polygone, die zu einem <i>RoofSurface</i> -Element gehören, weisen nach oben.
Definition	Ein <i>RoofSurface</i> rs ist eine Menge von Polygonen: $rs = \{F_0, F_1, \dots, F_m\}$, $m \geq 0$. Sei n_i^0 die normalisierte Flächennormale des Polygons F_i ($i=0 \dots m$). Für alle Polygone F_i ($i=0 \dots m$) muss gelten: der Winkel α zwischen n_i^0 und dem Vektor $(0,0,1)$ liegt zwischen $-90^\circ + \tau$ und $90^\circ - \tau$. Für τ ist der Wert $\tau=10^\circ$ empfohlen.
Ausnahmen	Sonderdachformen wie Zwiebdach und Ringpultdach, falls über Codelistenwert <i>rooftype</i> definiert.
Vorbedingung	Alle Anforderungen an Polygoneometrie (GE-gml:PO-*) müssen erfüllt sein.
CityGML Version	Seit 1.0
Illustration	
Prüfergebnis	RET_SE-bldg:RS-FN

2.2 Testdaten

Die Überprüfbarkeit der Prüfergebnisse kann nur dann gewährleistet werden, wenn die zu prüfenden Testdaten genau spezifizierte Mängel enthalten. Dazu wurden verschiedene Modelle erzeugt, die jeweils für genau eine Prüfmethode *check* das Ergebnis *false* ausgeben. In allen Fällen sollten neben der Check-ID weitere Informationen zur Ausprägung des gefundenen Fehlers ausgegeben werden. Das können z.B. die ID des defekten Elements (*LinearRing*, *Polygon*, *Solid*) oder die genaue Lage (Koordinaten) des Fehlers sein.

Im Rahmen des QIE wurden von der TU Delft 54, von der HFT Stuttgart 45 und von der SIG3D 3 Testdatensätze bereitgestellt. Diese Modelle bestehen aus je einem manuell erstellten Einzelgebäude. Außerdem stehen synthetisch erzeugte, absichtlich fehlerhafte Stadtmodelle zur Verfügung, die viele Gebäude umfassen (BILIECKI et al. 2014). Im Rahmen des QIE sind vor allem überlappende Objekte und falsche Semantik interessant. Darüber hinaus stehen reale Datensätze kompletter Städte bereit. Sie dienen vor allem zur Bestimmung der Leistungsfähigkeit der Tools und ihrer Algorithmen bei der Verarbeitung großer Datenmengen.

Die Untersuchungen im Rahmen des CityGML QIE erstrecken sich auf Gebäudemodelle der LoD 1 und 2. Um die Vielzahl an möglichen Modellierungsvarianten abzudecken, die CityGML ermöglicht, wurden 13 geometrisch nahezu identische Modelle eines einfachen *Building*-Elements erstellt (Abb. 1). Dabei unterscheiden sich 3 Modelle lediglich durch den Dachüberstand (Tabelle 2). Nicht alle entsprechen den Conformance Requirements des Standards, jedoch sind sie Schema-konform.



Abb. 1: Einfaches CityGML Building

Tabelle 2: Modellierungsvarianten eines einfachen Building nach CityGML 2.0

Name	Beschreibung
SimpleSolid_SBS	Einfaches CityGML-Building mit <i>Solid</i> -Geometrie und <i>BoundarySurfaces</i> als <i>MultiSurface</i> -Geometrie (Konflikt mit <i>Conformance Requirement 10.3.9.4 (CityGML 2.0 spec, 10.3.9,S. 78)</i>)
SimpleSolid_SBSx	Einfaches CityGML-Building mit <i>Solid</i> -Geometrie und leeren <i>BoundarySurface</i> -Elementen
SimpleSolid_Sx	Einfaches CityGML-Building mit <i>Solid</i> -Geometrie ohne <i>BoundarySurface</i> -Elemente
SimpleSolid_SBSref	Einfaches CityGML-Building mit <i>Solid</i> -Geometrie und <i>BoundarySurfaces</i> , die die <i>Solid</i> -Geometrie referenzieren (Konflikt mit <i>Conformance Requirement 10.3.9.4 (CityGML 2.0 spec, 10.3.9,S. 78)</i>)
SimpleSolid_SrefBS	Einfaches CityGML-Building mit <i>Solid</i> -Geometrie und <i>BoundarySurfaces</i> , deren <i>MultiSurface</i> -Geometrie vom <i>Solid</i> referenziert wird
SimpleSolid_xBS	Einfaches CityGML-Building, bestehend nur aus <i>BoundarySurface</i> -Elementen
SimpleSolid_MSBS	Einfaches CityGML-Building mit <i>MultiSurface</i> -Geometrie und <i>BoundarySurfaces</i> ebenfalls als <i>MultiSurface</i> -Geometrie (Konflikt mit <i>Conformance Requirement 10.3.9.4 (CityGML 2.0 spec, 10.3.9,S. 78)</i>)
SimpleSolid_MSBSref	Einfaches CityGML-Building mit <i>MultiSurface</i> -Geometrie und <i>BoundarySurfaces</i> , die die <i>MultiSurface</i> -Geometrie referenzieren (Konflikt mit <i>Conformance Requirement 10.3.9.4 (CityGML 2.0 spec, 10.3.9,S. 78)</i>)
SimpleSolid_MSrefBS	Einfaches CityGML-Building mit <i>MultiSurface</i> -Geometrie und <i>BoundarySurfaces</i> , deren <i>MultiSurface</i> -Geometrie vom <i>Solid</i> referenziert wird
SimpleSolid_MSx	Einfaches CityGML-Building mit <i>MultiSurface</i> -Geometrie ohne <i>BoundarySurface</i> -Elementes
SimpleSolidOverhangs_SrefBS	Einfaches CityGML-Building mit Dachüberstand, modelliert mit <i>MultiSurface</i> -Geometrie und <i>BoundarySurfaces</i> , deren <i>MultiSurface</i> -Geometrie vom <i>Solid</i> referenziert wird
SimpleSolidOverhangs_MSrefBS	Einfaches CityGML-Building mit Dachüberstand, modelliert mit <i>Solid</i> -Geometrie und <i>BoundarySurfaces</i> , deren <i>MultiSurface</i> -Geometrie vom <i>Solid</i> referenziert wird
SimpleSolidOverhangs_SMSrefBS	Einfaches CityGML-Building mit Dachüberstand, modelliert mit <i>Solid</i> -Geometrie (Volumenelement) und <i>MultiSurface</i> -Geometrie (Dachüberstände), bei dem die <i>MultiSurface</i> -Geometrie der <i>BoundarySurfaces</i> vom <i>Solid</i> referenziert wird

2.3 Software

Zur Beteiligung am CityGML QIE waren alle Interessierten aus dem kommerziellen und akademischen Bereich aufgerufen, die entsprechende Prüfwerkzeuge oder Testdaten einbringen konnten bzw. Erfahrungen aus der Qualitätsprüfung von Stadtmodellen beitragen wollten. Folgende Softwaretools wurden hinsichtlich ihrer Interoperabilität getestet:

- CityDoctor (Hochschule für Technik Stuttgart)
- CityServer 3D (Fraunhofer IGD)
- FME 2014 SP4 (Safe Software)
 - mit CityDoctor Transformer
 - mit GeometryValidatorWorkspace
- IfcExplorer for CityGML (Karlsruher Institut für Technologie)
- val3dity (TU Delft)
- virtualcitysystems Validator (virtualcitySystems)

Die Auswertung der Tests ist noch nicht abgeschlossen. Erste Zwischenergebnisse liefern jedoch wertvolle Hinweise hinsichtlich der einheitlichen Nomenklatur der Anforderungen sowie der Fehleridentifikatoren (vgl. 2.1), da sonst ein Vergleich der Ergebnisse kaum möglich ist. Insbesondere die Systematisierung der Fehleridentifikationen und deren klare Abgrenzung bei ähnlichen Fehlertypen sind wichtige Schritte auf dem Weg zu verbesserter Interoperabilität.

3 Ergebnisse

Da das CityGML QIE voraussichtlich noch bis März 2015 läuft, stehen derzeit keine endgültigen Ergebnisse fest. Deshalb wird hier nur auf das CityDoctor Validation Tool (WAGNER et al. 2013) eingegangen.

Wie in 2.3 erwähnt, spielt die eindeutige Identifizierung der Fehler bei der Interoperabilität der Software eine wichtige Rolle. Bisher werden vom Validation Tool noch interne Fehleridentifikatoren ausgegeben. Sie wurden im Rahmen des QIE den dort verabschiedeten gegenübergestellt, so dass eine eindeutige Zuordnung auch derzeit problemlos möglich ist.

Das verwendete Geometriemodell des Validation Tools verarbeitet derzeit keine inneren Ringe. Modelle, die Polygone mit Löchern enthalten, können somit derzeit nicht validiert werden.

Als einziges der untersuchten Prüfwerkzeuge stellt CityDoctor Plausibilitätsprüfungen für die Semantik von CityGML-Modellen zur Verfügung. Diese erstrecken sich auf die geometrisch-semantische Kohärenz (STADLER 2007) sowie auf Attribute mit Geometriebezug. Das Funktionieren des Ansatzes zeigen u.a. die Prüfergebnisse des synthetischen Modells mit zufällig verteilten *BoundarySurface*-Typen. Die Validierung ergab 1795 Fehler für das LOD2-Modell an 394 von 900 Gebäuden.

Mit derselben Strategie lassen sich auch bestimmte Checks zur Überprüfung von Conformance Requirements implementieren. Da sich die Diskussion in diesem Bereich noch in der konzeptionellen Phase befindet (vgl. 4), wird bis zu deren Abschluss von weiteren Entwicklungen zur Prüfung von Conformance Requirements abgesehen.

Tabelle 3: Ergebnisse für das CityDoctor Validation Tool.

Testdatensatz	valid	invalid
1 TU Delft (54 Modelle)	8/14	30/33
2 HFT Stuttgart (45 Modelle)	15/15	25/25

CityDoctor verarbeitet keine inneren Ringe. Daher ergeben sich falsche Prüfergebnisse bei Datensätzen, die Polygone mit Löchern enthalten. Bei Datensatz 1 wurden daher von 14 validen Modellen 6 als fehlerhaft ausgegeben. Umgekehrt wurden von 33 nicht validen Modellen 3 Stück als fehlerfrei ausgegeben, da die Fehler bei ungültigen inneren Ringen liegen. Dies wird in CityDoctor derzeit nicht geprüft. Bei der Überprüfung toleranz-abhängiger Fehler sind 3 Modelle betroffen. Eine weitere Schwierigkeit stellt die Verarbeitung sehr großer Zahlenwerte bei Koordinaten dar. In einem Testdatensatz werden Koordinaten mit 13-stelligen Integerwerte pro Dimension verwendet, die im CityDoctor nicht verarbeitet werden können.

Im Falle des zweiten Datensatzes traten keine Probleme auf, alle Modelle wurden korrekt validiert.

4 Zusammenfassung und Ausblick

Für die Qualitätsprüfung von 3D Stadtmodellen stehen derzeit nur wenige Software-Werkzeuge zur Verfügung. Manche davon befinden sich noch in der Entwicklung oder werden nicht vermarktet. Dennoch ist das Interesse groß, was auch durch die Teilnahme großer Hersteller wie ESRI oder Safe Software unterstrichen wird. Die Bedeutung von Standardisierung und Interoperabilität sind unbestritten.

Erste Zwischenergebnisse sind die Systematisierung der Checks, die zur Verabschiedung eines einheitlichen Nomenklaturschemas für alle Prüfbereiche führte. Daraus ergeben sich verbindliche Fehleridentifikatoren, die künftig von Softwareherstellern bevorzugt verwendet werden sollten, um Interoperabilität gewährleisten zu können. Zudem wurde eine eindeutige Definition für ein Solid-Element nach ISO 19107 vereinbart (LEDOUX 2014) sowie die Checks zur Überprüfung dieser Anforderung spezifiziert.

Im Bereich Semantik hat sich gezeigt, dass die Prüfergebnisse im Unterschied zur Geometrie eher als Hinweise auf die Plausibilität des verwendeten semantischen Elements zu verstehen sind. Dies gilt insbesondere für die geometrisch-semantische Kohärenz. Semantische Mängel können also nicht immer als Fehler im engeren Sinne bezeichnet werden, zumal es eine Vielzahl von Ausnahmen gibt.

Conformance Requirements müssen zunächst aus der Freitextformulierung des Standards in eine geeignete Form übertragen werden. Dabei ist darauf zu achten, dass die Zielformulierung eindeutig aus dem Ursprungstext ableitbar ist. Es ergeben sich somit verbindliche und optionale Kriterien, die automatisiert verarbeitet werden können. In einigen Fällen ist eine Ableitung solcher Regeln nicht möglich, da die zugrunde liegenden Formulierungen zu schwammig sind.

Zur Strukturierung des Validierungsprozesses wurde im Rahmen dieser konzeptionellen Diskussion ein Prozessplan erstellt, der auch auf andere Prüfbereiche übertragbar ist und zur weiteren Standardisierung des gesamten Prüfprozesses beitragen kann.

Das CityGML QIE wird voraussichtlich Ende März mit einem OGC-Report abgeschlossen, der die kompletten Ergebnisse in anonymisierter Form enthalten wird.

Das CityGML QIE konnte nur Dank der Unterstützung durch die Initiatoren Delft University of Technology, Fraunhofer Gesellschaft, Ordnance Survey International, EuroSDR und SIG3D sowie des Open Geospatial Consortium. Besonderer Dank geht an dieser Stelle an den Manager des Experiments, Carsten Rönsdorf, sowie an alle Aktiven in den Arbeitsgruppen.

5 Literaturverzeichnis

- BILJECKI, F., LEDOUX, H. & STOTER, J., 2014: Error propagation in the computation of volumes in 3D city models with the Monte Carlo method. ISPRS annals of the photogrammetry, remote sensing and spatial information sciences. Proceedings of the ISPRS/IGU joint international conference on geospatial theory, processing, modelling and applications, Toronto, Canada.
- LEDOUX, H., 2014: Three-dimensional primitives in the context of the CityGML QIE. https://github.com/tudelft-gist/CityGML-QIE-3Dvalidation/blob/master/solid_definition/qie_geom.pdf?raw=true [accessed 19.1.2015]
- STADLER, A., 2007: Kohärenz von Geometrie und Semantik in der Modellierung von 3D Stadtmodellen. Entwicklerforum Geoinformationstechnik, S. 167-181.
- WAGNER, D., ALAM, M. & COORS, V., 2013: Geometric validation of 3D city models based on standardized quality criteria. Urban and Regional Data Management: UDMS Annual 2013, 197.
- WAGNER, D., COORS, V. & BENNER, J., 2014a: Semantic Validation of GML-based geospatial data. Proceedings of 9th 3D GeoInfo Conference 2014, Breunig, M. (ed.) 12-13 November, Dubai, United Arab. Emirates, S. 1-15.
- WAGNER, D., KOLBE, T. H. & COORS, V., 2014b: Spezifikation von Prüfplänen und Prüfergebnissen zur Validierung von 3D-Stadtmodellen. Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e.V., Band 23, Seyfert, E., Gülch, E., Heipke, C., Schiewe, J. & Sester, M. (Hrsg.), 34. Wissenschaftlich-Technische Jahrestagung der DGPF, 26.-28. März 2014 an der HafenCity Universität Hamburg, Tagungsband auf CD-ROM, Beitrag 188.