

Untersuchungen zum Entwurf und der prototypenhaften Implementierung eines Koordinatentransformationsdienstes

Ulrich Lenk, Sebastian Scholl & Peter Fleischer

Zusammenfassung

Der Beitrag befasst sich mit der Spezifizierung und prototypenhaften Implementierung eines Koordinatentransformationsdienstes (*coordinate transformation service*, CTS). Dabei werden aktuelle (Industrie bzw. de Facto)-Standards untersucht, die Anforderungen an einen entsprechenden CTS aufgestellt und diskutiert. Eine prototypenhaft Implementierung wird vorgestellt.

Abstract

The article is related to specifications and a prototype implementation of a coordinate transformation service (CTS). In this course up-to-date industry and de-facto standards are discussed as well as requirements for a respective CTS are presented. The prototype implementation that resulted from this investigation is discussed.

1 Einleitung

Dass sich Geodaten oft auf verschiedene Koordinatensysteme und unterschiedliche Referenzsysteme beziehen, ist ein nur zu bekanntes Problem in der Geodäsie und Geoinformatik. Die Transformation von Geodaten zwischen verschiedenen Referenzsystemen zwecks Homogenisierung, sei es zu Analysezwecken oder »nur« für die Visualisierung, gehört zum alltäglichen Leben im Umgang mit digitalen Geodaten. Insbesondere vor dem gegenwärtig sehr aktuellen Hintergrund der Einrichtung von Geodateninfrastrukturen (GDI; z.B. Bernard et al. 2005) wird in vielen Veröffentlichungen erwähnt, dass es notwendig ist, (mindestens) einen Koordinatentransformationsdienst in einer derartigen Infrastruktur zur Verfügung zu haben. Das ist nicht nur auf regionaler bzw. nationaler Ebene der Fall (GIBa 2006; Müller und Portele 2005; Donaubaue et al. 2005; Aumann et al. 2003), sondern insbesondere im Kontext globaler Geodateninfrastrukturen (vergleiche Nebert 2004). In konkreten Konzepten zu GDIs der Länder werden auch Koordinatentransformationsdienste andiskutiert (z.B. GDI-NRW, aus der ein Entwurf eines *Web Coordinate Transformation Service* resultierte, siehe OGCE 2005). Aber auch global agierende Geoinformationszentren (gegebenenfalls verbunden mit weiteren Nutzern), die in dem Sinne als Mikro-Geodateninfrastruktur bezeichnet werden können, haben einen derartigen Bedarf nach entsprechenden Diensten, da sie Daten aus verschiedensten Quellen zu einheitlichen Produkten zusammenführen.

Interessanterweise ist die Thematik der Koordinatentransformationsdienste (aus subjektiver Sicht der Autoren) vom technischen Standpunkt aus bislang relativ wenig in der Literatur behandelt worden. Es gibt einige Spezifikationen auf dem Gebiet (diese werden nachfolgend behandelt werden), es gibt eine kleine Übersicht von Poth (2005) mit Einsatzszenarien des Koordinatentransformationsdienstes, es gibt einen Artikel von Rösch und Zimmermann (2005), der Entwurfsmuster bei Koordinatentransformationen behandelt, dabei aber bereits verfügbare internationale Standards nicht erwähnt. Jedoch ist über weiterführende Erfahrungen bei der Implementierung eines derartigen Dienstes nach Kenntnis der Autoren bislang nichts Wesentliches veröffentlicht worden. Dabei gibt es frei verfügbare Implementierungen, die prinzipiell genutzt werden können. Dieser Beitrag soll in diesem Sinne über Erfahrungen berichten, die im Rahmen des Entwurfs und der prototypenhaften Implementierung eines Koordinatentransformationsdienstes gemacht wurden. Der Artikel befasst sich nicht mit dem theoretischen Hintergrund von Koordinatentransformationen bzw. Koordinatenkonversionen, hierzu sei auf Standardliteratur wie Heck (2003) oder Kuntz (1990) bzw. auch auf die unten vorzustellende Quelle bei der *International Association of Oil and Gas Producers* (OGP; OGPc 2006) verwiesen.

2 Standards, Normen, Produkte und Daten für Koordinatentransformationsdienste

Dieser Abschnitt befasst sich mit der Vorstellung von existierenden Standards und Normen und entsprechenden Produkten, die jedoch nicht notwendigerweise auf den Standards aufbauen.

2.1 Open Geospatial Consortium und ISO

Das bekannte, 1994 gegründete *Open Geospatial Consortium* (OGC) ist eines der anerkanntesten Gremien, das sich mit dem Thema Standardisierung im Geoinformationsbereich befasst. Die hohe Akzeptanz zeigt sich allein schon daran, welche Firmen und Organisationen dem OGC angehören (z.B. die Firmen Oracle, ESRI, Intergraph Corporation, Infoterra Ltd. als Tochter der EADS Astrium, die Europäische Raumfahrtagentur ESA) und welche Verbreitung die Standards der OGC haben. Durch Kooperation der OGC mit der ISO sind einige OGC-Standards auch

gleichzeitig ISO-Standards, so zum Beispiel die *Geography-Mark-Up-Language* (GML)-Spezifikation (ISO 19136) oder der *Web Map Service* (ISO 19128, vgl. Kresse und Fa-daie 2004).

Beim Studium der im Internet verfügbaren *Baseline* für OGC-Dokumente finden sich etliche Dokumente, die sich allein schon aus dem Namen heraus mit dem Thema Koordinatentransformation befassen, zusätzlich gibt es Dokumente, die sich nicht auf den ersten Blick hin mit Koordinatentransformation befassen. Von der Vereinigungsmenge dieser Dokumente sind einige allerdings bereits überholt und haben in der Liste entsprechende Vermerke, die sie als *deprecated* ausweisen, so dass an gültigen aktuellen Dokumenten nur die folgenden Dokumente übrig bleiben (geordnet nach der Reihenfolge ihrer Erscheinung).

- »OpenGIS® Implementation Specification: Coordinate Transformation Services«; eine Implementationspezifikation (IS) in der Version 1.0 (Dokumenten-Nummer 01-009) vom 12.01.2001 (OGCa 2001). Dieses Dokument spezifiziert einen *Coordinate Transformation Service* (CTS) mit seiner Schnittstelle auf relativ technischem Niveau, mit Details zu den Schnittstellen, über die man den Service benutzen kann.
- »Recommended XML encoding of coordinate reference system definitions«; ein Empfehlungsdokument (Recommendation Paper) in der Version 2.1.0 (Dokumenten-Nummer 03-010r9) vom 16.10.2003 (OGCb 2003). Das Dokument beschreibt, wie Koordinatenreferenzsysteme mit XML spezifiziert werden können, wobei es stark auf dem nachfolgend aufgeführten Dokument basiert. Die Grundlage bildet jedoch noch die GML in der Version 3, die inzwischen durch jüngere Versionen aktualisiert ist. Prinzipiell wird das Dokument als veraltet aufgelistet, jedoch wird in dem Nachfolgedokument (OGCd 2005), das auf der GML 3.1.1 basiert, darauf verwiesen, dass nur Teile dieses Dokuments durch das neue ersetzt werden.
- »OGC Abstract Specification Topic 2, Spatial referencing by coordinates«; eine abstrakte Spezifikation (*Abstract Specification*) ohne Versionsangabe (Dokumenten-Nummer 04-046r3) vom 16.08.2004 (OGCc 2004). Der Zweck von abstrakten Spezifikationen von OGC ist es, Modelle in hinreichender Ausführlichkeit zu entwickeln und zu dokumentieren, damit auf der Grundlage dieser Modelle Implementierungen spezifiziert werden können. Dieses Dokument wurde auch der ISO als Entwurf einer Überarbeitung von ISO19111 vorgelegt.
- »Recommended XML/GML 3.1.1 encoding of common CRS definitions«; ein Empfehlungsdokument (*Recommendation Paper*) in der Version 1.0.0 (Dokumenten-Nummer 05-011) vom 28.01.2005 (OGCd 2005). Dieses Dokument basiert auf dem GML 3.1.1 (OGC Recommendation Paper 04-092r4).
- »Web Coordinate Transformation Service (WCTS) draft Implementation Specification«; ein Diskussionsbeitrag (*Discussion Paper*) in der Version 0.3.0 (Dokumenten-

Nummer 05-013) vom 30.01.2005. Dabei handelt es sich nicht um einen verabschiedeten OGC-Standard, sondern um einen Diskussionsbeitrag für einen *Web Service* für die Transformation von Geodaten über ein Computernetz, der aus dem Projekt Geodateninfrastruktur Nordrhein-Westfalen (GDI-NRW) heraus entstanden ist (OGCe 2005).

- »GO-1 Application Objects«; eine Implementationspezifikation (IS) in der Version 1.0 (Dokumenten-Nummer 03-064r10) vom 04.05.2005 (OGCf 2005). Diese Spezifikation ist das Ergebnis einer Initiative des OGC für Geographische Objekte (GO-1; *Geographical Objects Phase 1*). Ins Leben gerufen wurde diese Initiative mit dem Ziel, eine frei verfügbare Sammlung weit verbreiteter, leichtgewichtiger, sprachunabhängiger Abstraktionen zu entwickeln. Dass sich dahinter u. a. auch eine Spezifikation für einen Transformationsdienst bzw. die Modellierung von dafür notwendigen Objekten verbirgt, zeigt sich erst, wenn man sich mit dem Dokument näher befasst, da sich keine Hinweise darauf in der kurzen Beschreibung in der *Baseline*-Liste von OGC-Dokumenten finden.

Sucht man auf den Webseiten des Open Geospatial Consortiums jedoch nach entsprechenden Produkten, so finden sich gerade mal fünf Produkte, die den CTS implementieren, und zwei Produkte, die den WCTS in seiner *Draft*-Spezifikation 0.3 (OGCe 2005) implementieren. Gegenüber den bekanntesten OGC-Standards, dem *Web Map Service 1.1.1* und dem *Web Feature Service 1.0* (nicht transaktional, d. h. ohne Fähigkeit zur Aktualisierung von Daten), die mit 144 bzw. 86 Implementierungen aufgelistet werden, sind diese Zahlen doch etwas mager (Stand: 03.04.2006). Allerdings wird der WCTS teilweise auch als nicht so dringend erforderlich betrachtet (GIBa 2006).

Wichtig ist im Zusammenhang von Koordinatenoperationen und OGC/ISO hinsichtlich der Terminologie, dass differenziert wird zwischen Koordinatentransformation (*coordinate transformations*), bei denen ein Datumsübergang stattfindet, und Koordinatenumformungen (*coordinate conversions*, daher gegebenenfalls auch Koordinatenkonversionen), bei denen eine Umrechnung der Koordinaten bei gleichbleibendem Datum vorgenommen wird. Letzteres ist z. B. der Fall, wenn geografische Koordinaten in UTM-Koordinaten umgerechnet werden oder umgekehrt. Als Überbegriff für Transformationen und Konversionen wird der Begriff der Operation, d. h. Koordinatenoperation (*coordinate operation*), verwendet.

2.2 EPSG – European Petroleum Survey Group

Die ehemalige EPSG (*European Petroleum Survey Group*) wurde 1986 mit dem Ziel gegründet, *best practices* zu entwickeln und zu fördern, ein Forum für den Erfahrungs- und Wissenstransfer zu bilden, Einfluss zu nehmen auf entsprechende Standardisierungsgremien und

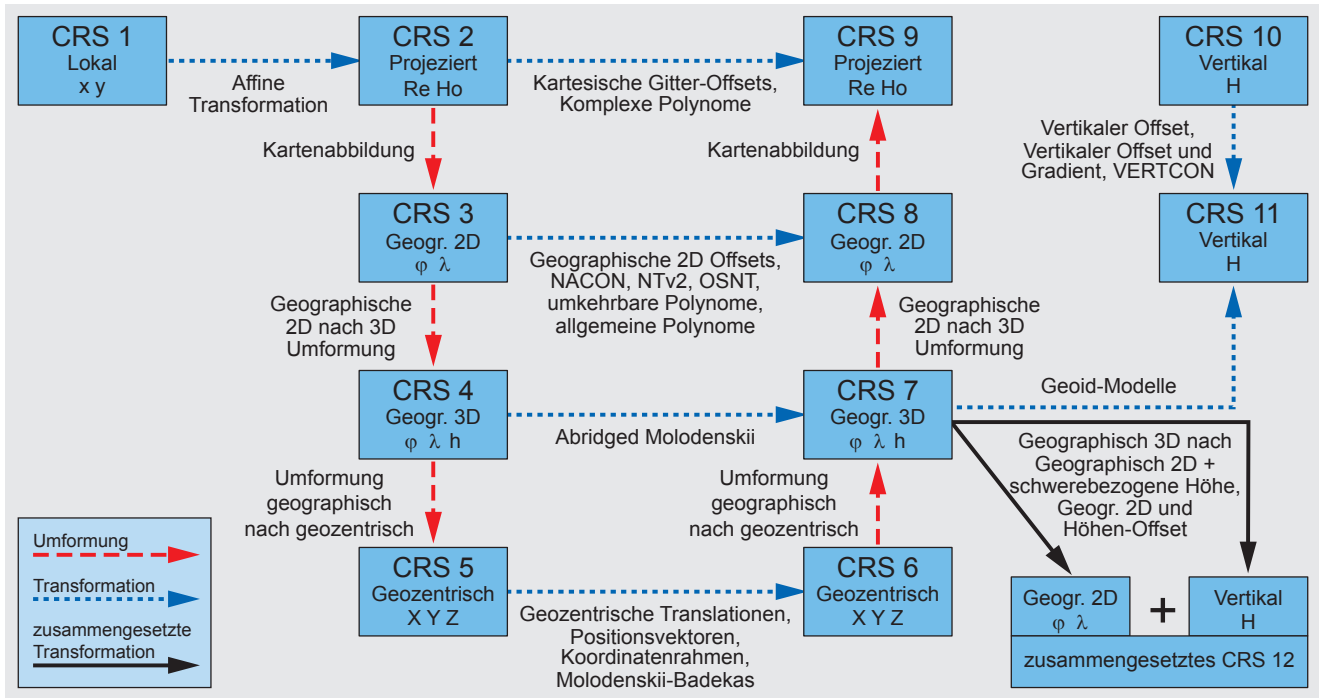


Abb. 1: Möglichkeiten, Koordinatentransformationen implizit zu erzeugen (nach OGPb 2006)

deren Standardisierungsprozesse sowie Kontakte zu den entsprechenden Industrie- und Fachverbänden zu halten (OGPa 2006). Sie ist inzwischen in das *Surveying and Positioning Committee* der *International Association of Oil and Gas Producers* (OGP) übergegangen.

Neben OGC gehört auch ISO zu den Partnern des Gremiums. Durch diese enge Vernetzung der Standardisierungsorganisationen und den an Normierung interessierten Organisationen kommt ein sehr hoher Standardisierungsgrad der Veröffentlichungen zustande. Diese hohe Standardisierung findet sich dadurch auch beim Hauptprodukt von EPSG, der geodätischen Parameterdatenbank (EPSG-Datenbank) und deren beiden Bedienungsanleitungen, den *Guidance Notes 7 Part 1* und *Part 2* (OGPb; OGPc 2006).

Im ersten Teil der Bedienungsanleitungen wird der allgemeine Umgang mit der Datenbank erläutert. Ausführlich erklärt werden das Konzept der Koordinatenreferenzsysteme und Koordinatentransformationen, der Umgang der Datenbank mit diesen Bereichen, der Aufbau der Datenbank und die Gewinnung und Verwendung von Daten aus der Datenbank. Dem Anwendungsentwickler werden Möglichkeiten aufgezeigt, durch geschickte Verkettung einzelner im Datensatz abgelegter Operationen mit eigenständig erzeugten Operationen die Anzahl der möglichen Transformationen gegenüber der Anzahl der abgelegten Operationen zu steigern. Durch korrekte Aneinanderreihung einzelner Transformationsschritte kann somit durch die Applikation eine sehr viel größere Menge an unterschiedlichen Koordinatentransformationen erzeugt werden. Abb. 1 veranschaulicht die Verkettungsmöglichkeiten der Entwickler, wodurch die Anzahl der möglichen Koordinatentransformationen gesteigert werden kann.

In der Datenbank ist nicht für jede Transformation zwischen existierenden geografischen Koordinatenreferenzsystemen (*coordinate reference systems*, CRS) ein Eintrag vorhanden. In der Anleitung für den EPSG-Datensatz sind einige Strategien erläutert, mit denen solche Probleme gelöst werden können. In Abb. 2 wird gezeigt, dass es möglich ist, einen Umweg über ein drittes Referenzsystem zu

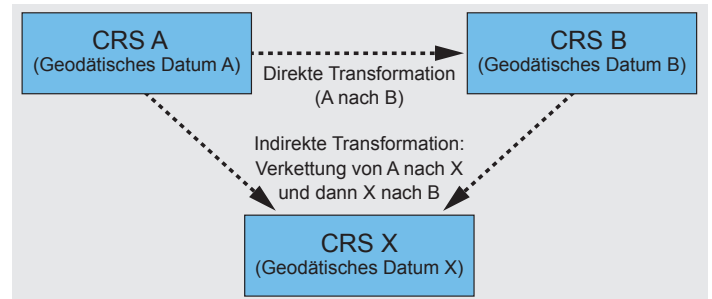


Abb. 2: Indirekte Koordinatentransformationen über ein Hub (nach OGPb 2006)

gehen. Damit diese Strategie erfolgreich ist und zu einer enormen Erhöhung der Transformationsanzahl führt, muss ein weit verbreitetes CRS wie ein *Hub* in einem Netzwerk als Transformationsschaltstelle genutzt werden. Dafür bietet sich das WGS84-CRS regelrecht an, da für sehr viele Referenzsysteme in der EPSG-Datenbank Transformationen zu diesem abgespeichert sind. Ein möglicher direkter Weg sollte dem Umweg über das WGS84-System jedoch immer vorgezogen werden.

Teil zwei der Bedienungsanleitung für die EPSG-Datenbank (OGPc 2006) befasst sich ausführlich mit den verschiedenen Algorithmen für Konversionen und Transformationen. Es sind die für eine Implementierung eines

Koordinatentransformationsdienstes wichtigsten Methoden und deren Formeln angegeben. Die Formeln sind aus dem Dokument auch in die Datenbank übernommen worden. Sie sind in der Tabelle *epsg_coordoperationmethod* in Spalte *formula* gespeichert.

Die EPSG-Datenbank unterstützt prinzipiell auch vertikale Datumsübergänge. Dies findet sich in den zur Verfügung stehenden zusammengesetzten (*compound*) CRS (CRS12, Abb. 1) sowie den 3-D-geografischen und den geozentrischen CRS (CRS4/7 und CRS5/6). Allerdings werden für Übergänge zu CRS12 und umgekehrt zusätzliche Daten benötigt, die nicht Teil der EPSG-Datenbank sind. Sie sind in der Datenbank referenziert und können daher in zu erstellende Anwendungen eingebunden werden. Es handelt sich dann um gitterförmige Daten, zwischen denen für konkrete Positionen (im Allgemeinen bilinear) interpoliert werden muss. Aus diesem Grund ist es klar, dass diese Transformationsparameter nicht direkt in der EPSG-Datenbank enthalten sein können. In den Beschreibungen der EPSG/OGP wird zudem darauf verwiesen, dass hier nur der genäherte additive Übergang durch die Summe von Geoidundulation N und der orthometrischen Höhe H mit dem Ergebnis der ellipsoidischen Höhe h betrachtet wird, der die doppelte Krümmung der Raumkurve der orthometrischen Höhe vernachlässigt (Heck 2003; Torge 2001). Dies entspricht auch der Vorgehensweise von OGC/ISO (OGC 2004).

2.3 Das Projekt deegree

Im Rahmen des *Open-Source*-Projektes *deegree* wird in einigen Veröffentlichungen auf die Existenz eines WCTS innerhalb dieses Paketes von Diensten hingewiesen (Fitzke et al. 2003; Fitzke et al. 2004). Initiiert wurde das Projekt *deegree* vom Geographischen Institut der Universität Bonn und wird auch weiter von dem Institut gemeinsam mit der Firma lat/lon, Bonn, gepflegt. Es gibt einige namhafte Referenzprojekte, in denen die Software eingesetzt wird (siehe Fitzke et al. 2003). *Deegree* implementiert neben dem WCTS gemäß der Spezifikation Version 0.3 auch die wichtigsten anderen *OGC-Web Services* (*Web Map Service*, *Web Feature Service*, *Web Coverage Service*, *Web Catalogue Service* sowie weitere) und ist damit neben einem Experimentalsystem der Firma PCI Geomatics die einzige auf den Internetseiten des OGC referenzierte Implementierung eines WCTS. Der WCTS von *deegree* wurde in seiner ersten Version im Rahmen der Untersuchungen zur Geodateninfrastruktur Nordrhein-Westfalen (GDI-NRW) spezifiziert und dann weiter entwickelt.

2.4 Das Projekt GeoTools mit GeoAPI

GeoTools ist ein *Open-Source-Projekt*, das sich in seinem Leitspruch selbst als den »frei zugänglichen Java-Werkzeugsatz für GIS« beschreibt. Das Ziel des Projekts ist es,

einen Kernsatz von Java-Objekten für GIS-Software in einem Framework bereitzustellen, mit dessen Hilfe Entwickler z.B. OGC-konforme, serverseitige Dienste entwickeln können. *GeoTools* besteht aus einem Kernsatz von API-Schnittstellen und deren Implementierung. Es handelt sich um eine Komplettimplementierung der GO-1-Spezifikation von OGC (OGCf 2005). *GeoTools* ist selbst ein Mitglied von OGC. Deshalb verpflichten sich die Entwickler dazu, OGC-Spezifikationen zu implementieren und nehmen an Treffen von OGC teil (*GeoTools* 2006). Ein Ziel des *Open-Source*-Projektes ist es, die Interoperabilität zwischen verschiedenen Anwendungen zu gewährleisten.

Es bestehen zwei Versionen von *GeoTools*: *GeoTools Lite* und *GeoTools*. *GeoTools Lite* ist eine Vorgängerversion von *GeoTools*, die in vielen Applikationen noch genutzt wird, da sie unter Java 1.1 lauffähig ist. Die neueste Version von *GeoTools* benötigt mindestens Java-Version 1.4.

GeoTools bietet umfangreiche Funktionalität im Bereich der GIS, u. a. für Koordinatentransformationen zwischen verschiedenen Referenzsystemen. Hierbei arbeitet *GeoTools* auf der EPSG-Datenbank. *GeoTools* ist sehr viel mächtiger als notwendig wäre, um Koordinatentransformationen zu ermöglichen. Durch seine modulare Architektur können Funktionalitäten aber problemlos hinzugefügt oder entfernt werden.

Die GO-1-Spezifikation des OGC (OGCf 2005) definiert eine umfassende Schnittstelle für das Arbeiten mit geografischen Objekten. Bei *GeoAPI* handelt es sich um den Quellcode dieser Schnittstelle in der Programmiersprache Java. An *GeoAPI* haben überwiegend dieselben Leute gearbeitet wie auch an der Schnittstellenspezifikation des OGC. Dies hat den Vorteil, dass mit den Ansprechpartnern Probleme beider Seiten besprochen werden können.

Etwas problematisch ist, dass Weiterentwicklungen der Schnittstelle hauptsächlich anhand der *GeoAPI*-Implementierung gemacht werden. Die Schwächen und Unvollständigkeiten werden in *GeoAPI* beseitigt bzw. implementiert und danach erst in die Spezifikation übernommen. Dadurch kann es passieren, dass die definierten Schnittstellen von der OGC-Internet-Seite nicht vollständig mit den Interfaces aus der *GeoAPI*-Implementierung übereinstimmen. Für Programmierer und Entwickler ist es deshalb sicherer, sich an der Java-Dokumentation bzw. an den Kommentaren in den *GeoAPI*-Quelldateien zu orientieren. So können Fehler bei der Planung und beim Design frühzeitig vermieden werden.

2.5 Der Koordinatentransformationsdienst des BKG

Das Bundesamt für Kartographie und Geodäsie stellt auf seinen Webseiten des Geodatenzentrums einen CTS zur Verfügung, der in Anlehnung an die Standards von ISO und OGC einzelne Koordinaten oder ganze ASCII-Dateien im Bereich von Deutschland zwischen den gängigen Geo-

referenzsystemen umrechnen kann. Der Service bietet eine interaktive Oberfläche und eine HTTP-Schnittstelle, die eine direkte Einbindung des *Coordinate Transformation Service* (CTS) in eigene Applikationen erlaubt. Alle Transformationen erfolgen mit einheitlichen Parametersätzen für Deutschland. Die Genauigkeit der Umrechnung entspricht dabei kartographischen Anforderungen im Maßstabsbereich 1:25.000 bis 1:1.000.000 (siehe die Webseiten des Geodatenzentrums, URL des Dienstes selber: <https://upd.geodatenzentrum.de/cts/server>).

2.6 Der projektierte WCTS der GIB

Im Rahmen der GIB ist ein WCTS projektiert (GIBb 2006). Dieser soll ab 2006 oder 2007 und mindestens bis 2010 zur Verfügung stehen, um Investitionssicherheit zu bieten. Er soll primär für die Umrechnung von Koordinaten bei Beibehaltung des zugrundeliegenden Ellipsoids genutzt werden, wie zum Beispiel bei der Umrechnung von geographischen WGS84-Koordinaten in ETRS89-Meridianstreifensysteme.

2.7 Weitere Produkte und Bibliotheken für Koordinatentransformationen

Bei der amerikanischen *National Geospatial-Intelligence Agency* (NGA) stellt deren Geodäsie-Gruppe einen *Geographic Translator* (GEOTRANS) zur Verfügung (URL: <http://earth-info.nga.mil/GandG/geotrans/index.html>), mit dem Koordinatenumformungen und Koordinatentransformationen berechnet werden können. Er stellt sich wie ein Taschenrechner dar, mit dem neben den bekannten zivilen Projektionen und Transformationen auch militärische Koordinatennotationen berechnet werden können. Dieser ist jedoch auch in andere Applikationen einbindbar, so wird z.B. bei den Anwendungen der Fa. TerraGo Technologies diese Bibliothek zur Verarbeitung von GeoPDF verwendet (Bufkin 2005).

Weiterhin gibt es die *PROJ.4 Cartographic Projections library* (URL: <http://remotesensing.org/proj/index.html>), mit der ebenfalls entsprechende Berechnungen durchgeführt werden können. Diese Bibliothek ist gemäß der Information auf den Webseiten in das bekannte freie GIS GRASS integriert.

3 Entwurf und Implementierung eines Koordinatentransformationssdienstes

3.1 Anforderungen an einen CTS

Vor dem Hintergrund des Bedarfs an Koordinatenoperationen, wie sie sich im Rahmen von Auswertezentren ergeben, wurden die nachfolgenden Top-Level-Anfor-

derungen an den Koordinatentransformationssdienst gestellt:

- Berücksichtigung der OGC- bzw. ISO-Standards;
- Anbindung der EPSG-Datenbank, um einen initialen Datensatz an Transformationsparametern zur Verfügung zu haben;
- Verfügbarkeit von Parametern für globale Anwendungen;
- Autarkheit des Dienstes; das bedeutet, er muss innerhalb eines abgeschlossenen Netzes ohne Anschluss an das Internet realisiert werden können;
- Nutzung einer bestehenden Datenbankzugriffsschicht aus einem Framework für diesen Service im Rahmen einer Mehrschichtenarchitektur (z.B. Starke 2002), der komponentenhaften Entwicklung von Softwaresegmenten und deren Wiederverwendung;
- Nutzbarkeit des Service von verschiedenen Plattformen aus;
- Nutzbarkeit des Dienstes als WCTS, wie er aktuell als Diskussionsgrundlage in OGCE (2005) als Draft spezifiziert ist;
- Fähigkeit zur Massen- und Einzelpunkttransformation;
- Berücksichtigung von Notationsanforderungen, z.B. die Darstellung von Koordinaten in Form von Zeichenketten im *Military Grid Reference System* (UTMREF, z.B. DMA 1990), als GEOREF (NATO 1993) oder die Darstellung von geografischen Koordinaten in verschiedenen Zeichenketten-Formen (Dezimalgrad, z.B. GGG.gggggggg[E/W]GG.gggggggg[N/S]); oder als Grad-Minuten-Sekunden mit Nachkommastellen bei den Sekunden, z.B. GGGMMSS.ssss[E/W]GGMMSS.ssss[N/S]);
- Fähigkeit für vertikale Datumsübergänge von Höhenmodellen;
- Fähigkeit zur Bestimmung von Höhenkomponenten für 2-D-Koordinaten;
- Fähigkeit zur (horizontalen) Transformation von georeferenzierten Rasterdaten (damit allgemein von Rasterdaten wie Luft- und Satellitenbildern, Rasterkarten und gitterartig vorliegenden Höhenmodellen);
- Fähigkeit zur Transformation von Vektordaten;
- Fähigkeit zur Berechnung von Transformationsparametern (zumindest in Standardverfahren wie z.B. der überbestimmten 2-D- bzw. 3-D-Helmerttransformation);
- Fähigkeit zur Aktualisierung bzw. Ergänzung von Transformationsparameterdatenbeständen.

Die letzte Anforderung kann natürlich kontrovers diskutiert werden. Wenn ein Dienst von einer Organisation zur Verfügung gestellt wird, so muss damit nicht notwendigerweise verbunden sein, dass ein Nutzer die Datenbestände aktualisieren kann. Eine sinnvolle Einschränkung der Anforderung wäre es dann, wenn es in dem Fall zumindest die Möglichkeit gibt, dass ein Nutzer sich eigene Transformationsparameter berechnen kann, mit de-

nen dann gleichsam an den CTS übergebene Koordinaten transformiert werden würden. Dies wäre durch die vorletzte Anforderung gegeben. Ebenso wäre es ein interessanter Service, wenn die Transformationsparameter an die aufrufende Applikation zurückgeliefert werden würden. In diesem Sinne würde dann wenigstens die Algorithmik hinter den Transformationen zur Verfügung gestellt im Sinne eines *Processing Service*.

3.2 Diskussion der Anforderungen

Bei der Entscheidung, wie der Prototyp realisiert werden sollte, wurde zunächst auf die Anforderung der Nutzbarkeit des Dienstes als WCTS verzichtet. Der prototypenhafte *Server* sollte zunächst einmal ohne Webanbindung realisiert werden, um in einem ersten Schritt die Architektur des *Servers* selber zu untersuchen. Da der Service zusätzlich von weiteren Plattformen nutzbar sein soll, ohne direkt über einen Web Service zu gehen, muss die Realisierung in diesem Zusammenhang ohnehin in zwei Schichten erfolgen, einer *Server*-Komponente, die z. B. über CORBA (*Common Object Request Broker Architecture*) angesprochen werden kann, und dann einen *Server* für den *Web Service*, der dann mit dem ersten *Server* kommunizieren kann. Eine Erweiterung des eigentlichen *Server Service* zu einem WCTS ist daher prinzipiell in einer weiteren Stufe möglich und kann bei den sich anschließenden Arbeiten berücksichtigt werden. Ebenso wurde festgelegt, zunächst einen Durchstich für reine 2-D-Punkttransformationen zu machen ohne die Berücksichtigung von Schwerefeldmodellen oder das Berücksichtigen von anderen Geoinformations-Formaten. Die Berücksichtigung von Geoidmodellen wird von den relevanten Standards und der EPSG-Datenbank durch die Verfügbarkeit von Klassen für zusammengesetzte CRS sowie 3-D-CRS prinzipiell unterstützt (Abs. 2.2; OGCa 2001; OGCc 2004; OGCf 2005; OGPb, OGPc 2006). Es bedarf jedoch der Einbindung zusätzlicher Datenquellen über die EPSG-Datenbank hinaus, so dass hiervon Abstand genommen wurde.

Die Nutzung eines existierenden Dienstes wie dem des BKG oder eines projektierten Dienstes wie der von GIB (GIBb 2006) kann unter der Vorgabe, dass der Dienst unabhängig von einem Zugang zum Internet sein muss, nicht erfolgen, zudem ist dann die Verwendung eigener Parameter nicht möglich und der Service des BKG beschränkt sich hinsichtlich seiner Parameter auf das Gebiet der Bundesrepublik. Dies widerspricht der Anforderung nach verfügbaren Transformationsparametern für globale Anwendungen, soweit dies nicht durch die EPSG-Datenbank abgedeckt ist. Es muss also mindestens eine eigene Quelle für die Transformationsparameter wie die EPSG-Datenbank genutzt werden, und aus kommerziellen und lizenzrechtlichen Gründen muss eine eigene Implementierung erfolgen.

3.3 Auswahl einer Spezifikation für den CTS

Auf der Suche nach einer geeigneten verbreiteten Spezifikation für die Implementierung eines Koordinatentransformationsservice fällt die erste Wahl sofort auf die Spezifikation Nummer 01-009 des OGC aus dem Jahre 2001 (OGCa 2001). Diese Schnittstellenspezifikation ist eine rein technische Anleitung für Programmierer, deren Ziel eine Implementierung eines CTS ist.

Charakteristisch für die Schnittstelle ist die Aufteilung der einzelnen *Interfaces* in die drei Pakete Positionsbestimmung, Koordinatensysteme und Koordinatentransformationen. Zu beachten ist, dass in dieser Spezifikation unter einem Koordinatensystem eigentlich ein Koordinatenreferenzsystem verstanden wird. Weil die Flexibilität der Schnittstellendefinition ein wichtiges Kriterium ist, können bei dieser Spezifikation beispielsweise Koordinaten beliebiger Dimensionen verwendet werden. Für die Verwendung dieser Spezifikation spricht, dass die Programmiersprache für die Implementierung relativ problemlos frei wählbar ist. Bei der Spezifikation sind schon die Schnittstellendefinitionen für COM und CORBA mitgeliefert. Für die Sprache Java ist auch der Code der Schnittstelle vorhanden. Ein weiterer Vorteil ist, dass diese Spezifikation speziell für Koordinatentransformationen erstellt wurde (Desruissaux 2005).

Von der anderen Seite betrachtet, bringt die CTS-Spezifikation auch einige Probleme mit sich. Einerseits basiert sie nicht auf dem aktuellen Modell der EPSG-Datenbank. Von OGC-Mitgliedern selbst wird sie als veraltet eingestuft, weil sie keinen ISO-Standard unterstützt und mittlerweile durch den ISO-Standard 19111 abgelöst wurde (Desruissaux 2005).

Dieser ISO-Standard wird über die abstrakte Spezifikation »OGC Abstract Specification Topic 2, Spatial Referencing by coordinates« (OGCc 2004) definiert und hat sich heutzutage durchgesetzt. In dieser ist der Umgang mit Objekten von Referenzsystemen, Koordinatensystemen, Transformationen etc. festgeschrieben. Auch die enge Verknüpfung des Objektmodells des ISO-Standards mit dem aktuellen Modell der EPSG-Datenbank spricht für eine Verwendung dieses Standards.

Daraus ergibt sich die Notwendigkeit, den CTS auf der Basis einer aktuelleren Spezifikation zu entwickeln, die den neueren ISO-Standard 19111 unterstützt, und dies ist für die GO-1 der Fall.

Im Vergleich zur CTS-Spezifikation bietet die GO-1-Spezifikation (OGCf 2005) einige Vorteile. Zur Basis dieses Dokuments gehört u. a. der ISO-Standard 19111 und die Spezifikation des CTS (OGCa 2001). Dadurch fließen die speziellen Aspekte für Koordinatentransformationen, unter anderem auch die Belange von vertikalen Datumsübergängen und 3-D-Koordinatensystemen, ein und gleichzeitig ist die Unterstützung eines heute gängigen ISO-Standards vorhanden. Die Spezifikation besitzt zudem eine hohe Konformität mit der Datenbankstruktur der EPSG (Desruissaux 2005).

Von Vorteil sind ebenso die bereits bestehenden *Open-Source*-Entwicklungen bezüglich der GO-1-Spezifikation mit der freien Bibliothek GeoAPI (siehe Abs. 2.4) in der Programmiersprache Java, durch die der Entwicklungsaufwand verringert wird. Außerdem ist ebenfalls die *Open-Source*-Komplettimplementierung in GeoTools (siehe Abs. 2.4) in der Sprache Java verfügbar.

Aus den genannten Gründen fiel daher die Entscheidung, die GO-1-Spezifikation und GeoAPI für die Implementierung des zu entwickelnden CTS zur verwenden. Bei der Implementierung der Schnittstelle werden die für Koordinatentransformationen irrelevanten *Interfaces* nicht berücksichtigt. Der Fokus der Entwicklung liegt dadurch klar auf dem *referencing*-Paket der Schnittstellendefinition.

3.4 Aspekte des Designs und der Implementierung des CTS

Die unter 3.4 nachfolgenden Abschnitte befassen sich mit der technischen Realisierung des CTS. Insbesondere aus Sicht des Hauptautors ist es notwendig, dass sich Geodäten noch viel mehr mit der Realisierung von Diensten befassen, mit denen das Fachwissen der Geodäsie den Anwendern und anderen Disziplinen in einfacher Form von Diensten (*Services*, siehe *Coordinate Transformation SERVICE*) zur Verfügung gestellt wird. Daher müssen sich Geodäten noch weit mehr mit Themen aus der Informatik befassen, um die Spezifikation und Entwicklung von Diensten mitgestalten zu können. Aus subjektiver Sicht des Hauptautors gibt es etliche Beispiele, bei denen die Initiative zur Implementierung von typisch geodätischen Aufgabenstellungen für die Praxis von Anwendern ausging und nicht von Seiten der Geodäsie.

3.4.1 Einbettung des CTS in das Framework

Ein erster Aspekt, der beim Entwurf des CTS berücksichtigt werden muss, ist die Frage, an welcher Stelle im *Framework* der Basisdienst implementiert wird. Durch die Schichtenstruktur des *Frameworks* ergeben sich zwei Möglichkeiten für die *Server*-seitige Einbindung des CTS in das *Framework*.

Die erste Möglichkeit ist, eine Transformation vom *Client* aus über bestimmte Kriterien (z. B. Quell- und Zielreferenzsystem) anzufordern. Diese wird auf dem *Server* erstellt und über CORBA an den *Client* übertragen. Von Vorteil ist hinsichtlich der Abhörsicherheit von CORBA, dass lediglich die Transformation übertragen wird und nicht die zu transformierenden Daten. Nachteilig ist der sehr hohe Entwicklungsaufwand bei der CORBA-Übertragung, da nahezu alle Klassen und *Interfaces* übertragbar sein müssen. Dieser Nachteil wiegt schwer, wenn andere Plattformen angebunden werden sollen. Für diese muss dann ebenfalls eine Komplettimplementierung durchgeführt werden.

Auch aus diesem Grund gelangt man zu einer zweiten Möglichkeit (Abb. 3). Der Basisdienst des CTS wird lokal kompakt gehalten. Dies bedeutet, dass der *Client* eine Anfrage über eine Zwischenschicht startet. Dabei werden in speziell entworfenen Klassen die gewünschten Kriterien und die zu transformierenden Koordinaten übertragen. Auf der *Server*-Seite wird die entsprechende Transformation erstellt und auf die Daten angewendet. Die Ergebnisse werden an den *Client* zurückgesendet. Dadurch be-

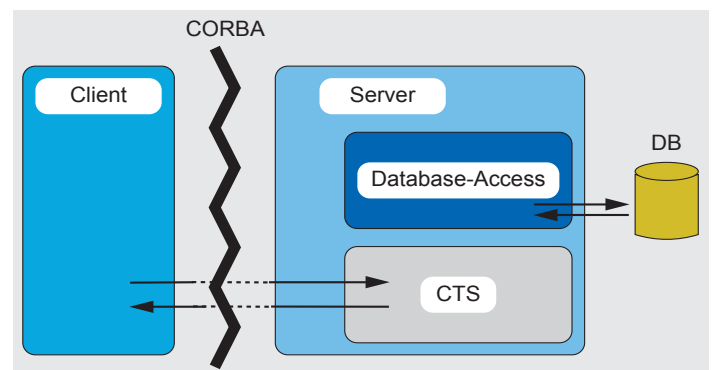


Abb. 3: Platzierung des CTS innerhalb des vorgegebenen Frameworks

stehen Klassen und *Interfaces* lediglich auf dem *Server* und sind lokal. Der Nachteil, dass Daten nicht ausreichend gesichert übertragen werden, relativiert sich, da die übertragenen Daten im Allgemeinen nicht von höchster Sensibilität sind. Von Vorteil ist, dass eine schlanke Schnittstelle mit geringem Entwicklungsaufwand zum *Client* besteht.

Aufgrund der kompakteren Lösung und des geringeren Entwicklungsaufwandes fällt die Entscheidung zugunsten der zweiten Lösung. Bezüglich der Performance schneidet sie allerdings bei einer großen Menge zu transformierender Daten (z. B. bei Bildern) schlechter ab, da diese über das Netzwerk übertragen werden müssen. In der ersten Lösung dagegen wird nur das Transformationsobjekt übertragen, dessen Datenmenge immer einen kleinen Wert hat. Dies gilt jedoch nur, solange Transformationen berechnet werden, die nicht auf Datenfeldern oder umfangreichen Parameterdatensätzen beruhen wie Geoid- oder Höhenmodelle. Wenn derartige vertikale Datumsübergänge berechnet werden, gilt dieses Argument nicht mehr, da für die Berechnung von Geoidmodellen deutlich größere Datenmengen an den *Client* übertragen werden müssten. Die erste Lösung führt aber auch zu grundlegenden wirtschaftlichen Bedenken. Wenn man ein Transformationsobjekt von einem CTS mit entsprechenden Parametern erhält, so benötigt es keinen großen Aufwand, die Parameter auszulesen. Vor dem Hintergrund, dass die Bestimmung von derartigen Parametern teilweise mit erheblichem Aufwand und Kosten verbunden ist, so ist dieser Aspekt zumindest kritisch zu bedenken oder vertraglich zu regeln.

3.4.2 Einheiten und Koordinatennotationen

Bei der Prüfung der Spezifikationen stellt sich heraus, dass eine Darstellung von Koordinaten als Zeichenketten, wie es bei UTMREF oder GEOREF der Fall ist, von

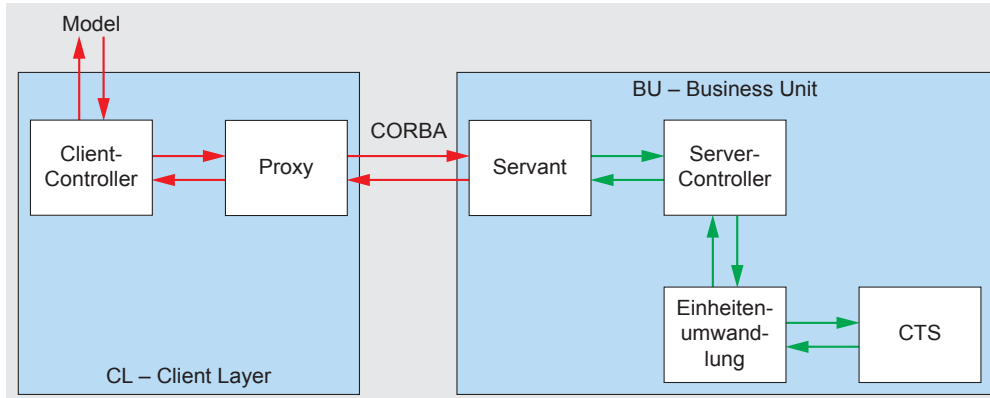


Abb. 4: Teilschritte einer Koordinatenoperation

den OGC/ISO-Standards in Ein- und Ausgabe nicht unterstützt wird. Die Einzeltransformationen werden in Basiseinheiten (Meter und Bogenmaß) berechnet. Für nicht unterstützte Koordinatenformen muss daher vor (und nach) der Transformation eine Koordinatenumformung (Einheitenwandlung) in die Basiseinheiten (und umgekehrt) in einer Zwischenschicht im Server erfolgen (Abb. 4). Für Winkelangaben und nicht-metrische Längeneinheiten ist dies trivial. Es gibt aber Koordinatennotationen, bei denen die Koordinaten in einer speziellen Zeichenkettenform angegeben werden, z.B. 28° 30' 00.00" N, oder UTMREF oder GEOREF. Diese Umwandlung muss speziell implementiert werden.

3.4.3 Implementierung des Prototypen

Die Implementierung des Prototypen erfolgte in der Programmiersprache Java. Für die Einheitenumwandlung wurde das frei erhältliche Java-Paket *javax.units* von der Apache-Organisation benutzt. Es wird auch von GeoTools verwendet.

Es wurde zunächst nur eine Beispieltransformation realisiert, um anhand dieser die prinzipielle Realisierbarkeit des Dienstes mit seinen Anforderungen zu demonstrieren.

Damit der auf einem Server integrierte CTS von Projekten oder anderen Programmen aus angesprochen werden kann, ist eine Entwicklung einer Anbindung über eine definierte Schnittstelle notwendig. Auf die standardisierte Schnittstelle des CTS wird also bei dessen Nutzung nicht direkt zugegriffen, sondern lediglich auf eine eigens implementierte Schnittstelle zum Server.

Eine Anfrage an den Service verläuft wie in Abb. 4 veranschaulicht. Dabei werden CORBA-fähige Modelle erzeugt und diese über den Client-Controller sowie weitere Komponenten des Frameworks (Proxy und Servant) an den Server-Controller übergeben. Dieser beinhaltet die

Logik, führt – falls notwendig – eine Einheitenumwandlung durch und ruft den CTS für das CORBA-Modell auf. Nachdem die Transformation durchgeführt ist, wird das CORBA-Modell durch alle Komponenten wieder zurückgegeben.

Auf die weiteren sehr technischen Details der Implementierung soll an dieser Stelle aus Platzgründen nicht eingegangen werden. Erwähnt werden soll nur noch, dass auch die Ansätze von Rösch und Zimmermann (2005) berücksichtigt wurden bezüglich des Einsatzes von Entwurfsmustern; so wurden u. a. *Singletons* für die Fabriken verwendet, mit denen die Objekte erzeugt werden.

4 Tests und Ergebnisse

4.1 Performance des CTS

Obwohl die Performance des Services nicht ausdrücklich als Anforderung besteht, ist sie dennoch in den Fokus der Entwicklung geraten, weil die bislang erreichten Ergebnisse in keiner Weise zufriedenstellend sind.

Während die eigentliche Berechnung der Zielkoordinaten noch in einem vertretbaren Rahmen liegt, dauert die Erstellung einer Koordinatentransformation deutlich zu lang. Sie liegt im Bereich von über fünf Sekunden für die Beispieltransformation zwischen den beiden verwendeten kalifornischen *ProjectedCRS*.

Die auf die Erstellung folgende Durchführung von tausend Transformationen verbraucht relativ wenig Zeit. Mit der programmierten Messmethode ist ein Zeitaufwand von 65 Millisekunden gemessen worden. Deutlich mehr Transformationen erhöhen selbstverständlich die Berechnungsdauer. Hunderttausend Koordinatenberechnungen benötigen über drei Sekunden. Da vor allem die Erstellungszeit für den praktischen Einsatz inakzeptabel ist, sollte bei zukünftigen Arbeiten hier ein Schwerpunkt liegen.

Bei der Analyse des jetzigen Entwicklungsstands mit *Rational Quantify* von *Rational Software* fällt auf, dass auf den ersten Plätzen in der Rechenzeitrangliste keine Klassen des CTS auftauchen. Naheliegender ist, dass die Datenbankschicht die meiste Zeit benötigt, weil viele Datenbankklassen in den vorderen Rängen auftauchen und auch die Gründe für diese Verzögerung zum Teil bekannt sind.

Lösungsmöglichkeiten für die Rechenzeitverminderung der Datenbankschicht (z. B. Starke 2002) sind:

- Eigene Datenbankschicht ohne Einsatz der im Framework eingesetzten *Java Reflection API*. Dies wird immer dann benötigt, wenn die zu erzeugende Klasse und

deren Komponenten zur Übersetzungszeit noch nicht bekannt sind. In einer eigenen Datenbankschicht ist es nicht notwendig, auf diese API zurückzugreifen, da die zu ladenden Klassen bekannt sind.

- Optimierung der vorhandenen Datenbankschicht: Für diesen Ansatz ist es notwendig, sich tiefer in die *Database-Access*-Schicht des *Frameworks* einzuarbeiten.
- Optimierung der Abfragen: Mit Bezug auf die Datenbankschicht ist es möglich, am Design des CTS und an einzelnen Funktionen selber Optimierungen hinsichtlich Performancesteigerung vorzunehmen.
- Eigener *Cache*: Vor dem Laden aus der Datenbank kann in einem eigenen *Cache* nachgeschaut werden, ob das Objekt bereits zuvor geladen und in diesem *Cache* gespeichert wurde. Ist dies der Fall, so ist kein weiterer Datenbankzugriff notwendig. Dieser *Cache* kann z. B. beim *Server*-Start befüllt werden und bei Bedarf aktualisiert werden.

4.2 Genauigkeit des CTS bei Verwendung der EPSG-Daten für das Beispiel

Bezüglich der Genauigkeit der Berechnungen sieht das Ergebnis deutlich besser aus als beim Kriterium Performance. Wegen der in der Datenbank beschriebenen Ungenauigkeit der Algorithmen unterscheidet sich das Ergebnis, wenn für die gleiche Transformation ein anderer Transformationsweg gewählt wird. Beispielsweise unterscheidet sich das Ergebnis für die Beispieltransformation dieser Entwicklung des CTS vom GeoTools-Ergebnis um bis zu fünf Meter! Das liegt daran, dass GeoTools für diese Transformation die NADCON-Methode verwendet und der selbst entwickelte Service den Umweg über das WGS84-System beschreitet.

Einen gut einzuordnenden Wert für die Genauigkeit der Operationsalgorithmen erhält man, wenn mehrfach hintereinander mit den jeweiligen Ergebniswerten die Operation und deren Inverse ausgeführt werden, da sich dann der Ausgangswert wieder ergeben muss. Der relative Fehler von 0.034 (Koordinaten: 1.0, 1.0) als Ergebnis für die Beispieltransformation nach hundert solchen Schritten im CTS ist ein akzeptabler Wert, zumal GeoTools hierbei erheblich ungenauer ist.

5 Schlussfolgerungen und Ausblick

Die Ergebnisse des Prototyps sind in Anbetracht der kurzen Entwicklungszeit zufriedenstellend, wenngleich sich bereits jetzt einige Ansätze für die weitere Entwicklung zeigen. Zum einen muss der Service hinsichtlich der Algorithmen weiter ausgebaut werden. Ein Anschluss an einen entsprechenden WCTS wäre weiterhin einer der nächsten Schritte. Auf die potentiellen Verbesserungsmöglichkeiten hinsichtlich Performance wurde im Abschnitt 4.1 hingewiesen.

Bezüglich der diskutierten Spezifikationen ist zu sagen, dass die Möglichkeit für die Berechnung von Transformationsparametern vollständig fehlt. Es ist verständlich, dass ein CTS nicht unbedingt eine Möglichkeit haben sollte, dass Nutzer eigene Parameter einspielen, da hier dann keine Kontrolle über die Transformationsparameter und insbesondere deren Qualität mehr gewährleistet werden kann. Dies ist nicht zuletzt aus Haftungsgründen keine tragbare Lösung. Jedoch wäre es zu überdenken, ob die Schnittstellen nicht dahingehend erweitert werden, dass sich Nutzer mit dem CTS eigene Transformationsparameter aus einem Satz von identischen Punkten berechnen können, um diesen Satz von Parametern dann auf weitere Punkte anwenden zu können.

Ebenso fehlen bei den existierenden Spezifikationen die Möglichkeiten, etablierte Zeichenkettennotationen für Koordinaten wie UTMREF und GEOREF zu verwenden. Die Möglichkeit, einen Übergang von 2-D-Koordinaten nach 3-D-Koordinaten zu machen, ist durch die verwendeten Spezifikationen abgedeckt. Allerdings bedarf es hier zusätzlich einzubindender Daten, die durch den verfügbaren EPSG-Datensatz nicht direkt zur Verfügung stehen. Zusätzlich muss beachtet werden, dass dieser Ansatz nur für Einzelpunkte ohne geometrischen und semantischen Kontext korrekt ist. Bei der Berechnung von Höhenkomponenten zu 2-D-Vektordatenbeständen stellt sich der Sachverhalt anders dar. Hier muss eine geometrisch und semantisch korrekte Datenintegration von 2-D-Situationsmodellen und Höhenmodellen erfolgen. Dies ist Gegenstand aktueller Forschung (vgl. Koch und Heipke 2004; Lenk und Heipke 2002; 2006).

Vertikale Datumsübergänge wurden bei diesen Untersuchungen nicht weiterführend betrachtet. Allerdings ist festzustellen, dass die aktuellen Spezifikationen einfache additive Übergänge berücksichtigen, die nach Heck (2003) für geodätische Genauigkeitsanforderungen in der Nähe der Erdoberfläche ausreichend sind. Hier kommt es auf die Qualität der verwendeten Daten und der Interpolationsansätze an, um zufriedenstellende Ergebnisse zu erzielen. Dies sollte bei weiteren Arbeiten detaillierter untersucht werden.

Angesichts der Vielfalt der in einem CTS zu implementierenden Koordinatenoperationen wäre es interessant, einen generischen Ansatz für die Berechnungen einzusetzen. Würde man die Formeln für die Koordinatenoperationen in einer interpretierbaren Form in der Datenbank (vergleiche Abs. 2.2) ablegen, wie es zum Beispiel bei der Formelsetzung mit Latex geschieht, so könnte Entwicklungsaufwand vermieden werden. Mit der Aktualisierung der Datenbasis für eine neue Koordinatenoperation wäre diese bereits verfügbar.

Die Verwendung von *Singletons* für die Fabriken, wie sie auch von Rösch und Zimmermann (2005) beschrieben ist, wird inzwischen von den Autoren kritisch hinterfragt. Beim vorgestellten CTS sind die Fabrikklassen alle als *Singletons* realisiert. Probleme können dann auftreten, wenn mehrere *Threads* dieselbe Fabrik gleichzeitig

benutzen wollen. Gegebenenfalls ist hier bei der weiteren Entwicklung eine Synchronisationsuntersuchung durchzuführen.

Dank

Die Autoren bedanken sich bei den Herren Dipl.-Inform. Th. Stoyke und Dipl.-Inform. (BA) Kenneth Schlör, beide EADS Deutschland GmbH (Friedrichshafen), für die Unterstützung dieser Diplomarbeit durch ihre Hilfe im Bereich Datenbanken und Serverentwicklung. Weiterer Dank gilt Martin Desruisseaux für seine Hinweise zu CTS und zur GO-1 Spezifikation (OGCf 2005).

Literatur

Aumann, G., Kunkel, T., Donaubaue, A., Schilcher, M.: OGC Web Services zur interoperablen Nutzung verteilter Geodatenbanken für mobile Anwendungen. GIS 6/2003, 17–23 (2003).

Bernard, L., Fitzke, J., Wagner, R.M.: Geodateninfrastruktur – Grundlagen und Anwendungen. Wichmann, Heidelberg (2005).

Bufkin, M.: GeoPDF: Description, Use and Chronology. Whitepaper von Laython Graphics/TerraGo Technologies (2005): Online-Literatur: http://www.laytongraphics.com/geopdf_white_paper.pdf, zuletzt besucht am 01.04.06.

Desruisseaux, M.: Persönliche E-Mail-Kommunikation mit einem der Entwickler von GeoAPI und GeoTools, Oktober 2005.

DMA (Defence Mapping Agency): Datums, Ellipsoids, Grids, and Grid Reference Systems. Defence Mapping Agency Technical Manual 8358.1 (1990). Online-Literatur: http://earth-info.nga.mil/GandG/coordsys/csac_pubs.html, zuletzt besucht am 27.04.06.

Donaubaue, A., Kunkel, T., Schilcher, M., Teege, G.: OGC Web Services zur interoperablen Nutzung verteilter Geodatenbanken für die Immobilienwirtschaft. In: Bernard et al. (2005), S. 202–210.

Fitzke, J., Greve, K., Müller, M., Poth, A.: Deegree – ein Open-Source-Projekt zum Aufbau von Geodateninfrastrukturen auf der Basis aktueller OGC- und ISO-Standards. GIS 9/2003, 10–16, 2003.

Fitzke, J., Greve, K., Müller, M., Poth, A.: Building SDIs with Free Software – the deegree project. In: Proc. 7th Conf. Global Spatial Data Infrastructure (GSDI 7) Bangalore, India, 2.–6. Feb. 2004, S. 97–103.

GIBa, 2006: Geodateninfrastruktur im Land Brandenburg – Sollkonzept GIB (2006). Online-Literatur: http://www.gib-portal.de/papers/Sollkonzept_GIB_komplett.pdf, zuletzt besucht: 12.04.2006.

GIBb, 2006: Geodateninfrastruktur Berlin/Brandenburg – Geoservice Application Profile (GAP) (2006). Online-Literatur: <http://www.gib-portal.de/papers/gap-v100.pdf>, zuletzt besucht: 26.07.2006.

GeoTools: Offizielle Homepage. <http://www.geotools.org>. Internetseite, zuletzt besucht: 08.02.2006.

Heck, B.: Rechenverfahren und Auswertemodelle der Landesvermessung. 3. Auflage, Wichmann, Heidelberg (2003).

Koch, A., Heipke, C.: Semantically Correct 2.5D GIS Data – the Integration of a DTM and Topographic Vector Data. In: Fisher, P. (Hrsg.): Developments in Spatial Data Handling, 11. Symposium on Spatial Data Handling. Berlin-Heidelberg-New York: Springer-Verlag, 2004, S. 509–526.

Kresse, W., Fadaie, K.: ISO Standards for Geographic Information. Springer, Berlin, Heidelberg, New York (2004).

Kuntz, E.: Kartennetzentwurfslehre. 2. Auflage, Wichmann, Heidelberg (1990).

Lenk, U., Heipke, C.: Ein 2.5D-GIS-Datenmodell durch Integration von DGM und DSM mittels Triangulationen – theoretischer und praktischer Vergleich von Algorithmen und ihrer Ergebnisse. Photogrammetrie Fernerkundung Geoinformation 3/2002: 205–216.

Lenk, U., Heipke, C.: The radial topology algorithm – a new approach for deriving 2.5D GIS data models. GeoInformatica, im Druck (2006).

Müller, M., Portele, C.: GDI-Architekturmodelle. In: Bernard et al. (2005), S. 82–92.

NATO (North Atlantic Treaty Organization): STANAG No. 3408 (Edition 3) Position Reference System for Aeronautical Charts (1993).

Nebert, D.D.: Developing Spatial Data Infrastructures: The SDI Cookbook Version 2.0. Global Spatial Data Infrastructure (2004). Online-Literatur: <http://www.gsdi.org/docs2004/Cookbook/cookbookV2.0.pdf>, zuletzt besucht 22.03.2006.

OGCa: OpenGIS® Implementation Specification: Coordinate Transformation Services Revision 1.00. Open Geospatial Consortium, Inc., Dokumenten-Nummer 01-009 (2001). Online-Literatur: http://portal.opengeospatial.org/files/?artifact_id=999, zuletzt besucht 22.03.2006.

OGCb: Recommended XML encoding of coordinate reference system definitions 2.1.0. Open Geospatial Consortium, Inc., Dokumenten-Nummer 03-010r9 (2003). Online-Literatur: http://portal.opengeospatial.org/files/?artifact_id=11517, zuletzt besucht 22.03.2006.

OGCc: OGC Abstract Specification Topic 2, Spatial referencing by coordinates. Open Geospatial Consortium, Inc., Dokumenten-Nummer 04-046r3 (2004). Online-Literatur: https://portal.opengeospatial.org/files/?artifact_id=6716, zuletzt besucht 22.03.2006.

OGCd: Recommended XML/GML 3.1.1 encoding of common CRS definitions 1.0.0. Open Geospatial Consortium, Inc., Dokumenten-Nummer 05-011 (2005). Online-Literatur: https://portal.opengeospatial.org/files/?artifact_id=8837, zuletzt besucht 22.03.2006.

OGCe: Web Coordinate Transformation Service (WCTS) draft Implementation Specification 0.3.0. Open Geospatial Consortium, Inc., Dokumenten-Nummer 05-013 (2005). Online-Literatur: https://portal.opengeospatial.org/files/?artifact_id=8847, zuletzt besucht 22.03.2006.

OGCf: GO-1 Application Objects 1.0. Open Geospatial Consortium, Inc., Dokumenten-Nummer 03-064r10 (2005). Online-Literatur: http://portal.opengeospatial.org/files/?artifact_id=999, zuletzt besucht 22.03.2006.

OGPa: <http://www.epsg.org/>, zuletzt besucht 31.07.2006.

OGPb: Guidance Note Number 7, part 1 – Using the EPSG Geodetic Parameter Dataset, Version 2.0 (2006). Online-Literatur: <http://www.epsg.org/guides/docs/G7-1.pdf>, zuletzt besucht 27.07.2006.

OGPc: Guidance Note Number 7, part 2 – Coordinate Conversions and Transformations including Formulas, Version 29 (2006). Online-Literatur: <http://www.epsg.org/guides/docs/G7-2.pdf>, zuletzt besucht 03.02.2006.

Poth, A.: Koordinatentransformation mit dem Web Coordinate Transformation Service. In: Bernard et al. (2005), S. 145–150.

Rösch, N., Zimmermann, J.: Der Einsatz objektorientierter Entwurfsmuster am Beispiel der Koordinatentransformation. AVN 8-9: 282–289 (2005).

Scholl, S.: Entwurf und prototypenartige Implementierung eines OGC-konformen Coordinate Transformation Services. Unveröffentlichte Diplomarbeit an der Fachhochschule Furtwangen, Fachbereich Allgemeine Informatik (2006).

Starke, G.: Effektive Software-Architekturen. Carl Hanser, München, Wien (2002).

Torge, W.: Geodesy. 3rd Edition. Walter de Gruyter, Berlin, New York (2001).

Relevante Internetadressen

OGP Surveying and Positioning Committee bzw. EPSG: <http://www.epsg.org/>

OGC: <http://www.opengeospatial.org/>

GeoAPI: <http://geoapi.sourceforge.net/>

GeoTools: <http://www.geotools.org/>

deegree: <http://deegree.sourceforge.net/>

GDI-NRW: <http://www.gdi-nrw.org/>

Geodatenzentrum des BKG: <http://www.geodatenzentrum.de/>

Anschrift der Autoren

Dr.-Ing. Ulrich Lenk, PgDipHS
c/o EADS Deutschland GmbH, Dept. MSIC21
88039 Friedrichshafen
Tel.: +49 (0)7545 8-8075, Fax: +49 (0)7545 8-9496

Dipl.-Inform. (FH) Sebastian Scholl
Unterländer Straße 24, 70435 Stuttgart

Prof. Dr. Peter Fleischer
Hochschule Furtwangen, Fakultät Informatik
Robert-Gerwig-Platz 1, 78120 Furtwangen
Tel.: +49 (0)7723 920-2409, Fax: +49 (0)7723 920-2610