

Softwareentwicklung in der Wissenschaft

Prof. Dr. Jörn Behrens,
CliSAP,

Numerische Methoden in den Geowissenschaften

Alexander Njemz
8njemz@informatik.uni-hamburg.de

8. März 2011

Inhaltsverzeichnis

1	Einführung	1
1.1	Einleitung	1
1.2	Interviewführung	2
1.3	Interviewpartner	2
1.4	CliSAP	3
1.5	AWI	3
2	Praxis der Softwareentwicklung	3
2.1	Wissenschaftliches Rechnen	3
2.2	TsunAWI	4
2.2.1	Das Sumatra-Andaman-Erdbeben und GITEWS	4
2.2.2	Das Projekt	6
2.3	Amatos	6
2.3.1	Adaptive Gitter	6
2.3.2	Das Projekt	7
2.4	Praxis	7
2.4.1	Hardware	7
2.4.2	Sprachen	8
2.4.3	Verwendete Bibliotheken und Compiler	8
2.4.4	Code Editing	8
2.4.5	Debugging	8
2.4.6	Testing & Validierung	9
2.4.7	Versionskontrolle	9
2.4.8	Projektmanagement	11
2.4.9	Codequalität & Code Review	11
2.4.10	Codedokumentation	11
2.4.11	Ausblick	12
2.4.12	Äußere Umstände der SiW	12
2.5	Kurs	12
3	Abschluss	13
3.1	Zusammenfassung	13

1 Einführung

1.1 Einleitung

In dieser Arbeit werden die Inhalte eines Interviews wiedergegeben, das im Rahmen eines Seminars an der Universität Hamburg im Wintersemester

2010/11 statt gefunden hat. Titel des Seminars ist “Softwareentwicklung in der Wissenschaft”. Der Interviewpartner war Prof. Dr. Jörn Behrens. Zunächst werden zur Einführung kurz Informationen zu der Interviewführung, dem Interviewpartner und zu Institutionen, die für seinen Werdegang von Interesse sind, gegeben. Im zweiten Teil werden dann erst zwei Softwareprojekte vorgestellt, an denen Herr Behrens maßgeblich mitgewirkt hat, und es wird auf die Praxis der Softwareentwicklung eingegangen.

1.2 Interviewführung

Zur Vorbereitung des Interviews wurden zum einen die Webseite [1] und eine wissenschaftliche Publikation [2] über *amatos* zur Verfügung gestellt. In eigenständiger Recherche wurde dann noch die Habilitationsschrift von Professor Behrens gefunden, aus der das Vorwort und die Einleitung zu Rate gezogen wurden. Des weiteren wurden die Materialien der Webseite des Seminars [3] zuhulfe genommen, vor allem der exemplarische Fragenkatalog für die Interviewführung. Ein weiteres Hilfsmittel, das zum Einsatz kam, war ein Digitales Aufnahmegerät welches am Medienkompetenzzentrum (MCC) des Regionalen Rechenzentrums der Universität Hamburg entliehen wurde.

Das Interview fand in Professor Behrens’ Büro am 09.12.2010 um 10 Uhr morgens statt und dauerte etwa eine Stunde. Der Verlauf des Interviews war nicht durchgeplant. Es war im Interesse des Interviewenden den Interviewten möglichst frei sprechen zu lassen, nur gelegentlich wurden Detailfragen zum Softwareentwicklungsprozess gestellt.

1.3 Interviewpartner

Das Interview wurde mit Jörn Behrens geführt. Herr Behrens ist seit 2009 Professor für numerische Methoden in den Geowissenschaften am KlimaCampus der Universität Hamburg. Von 2006 bis 2009 war er Leiter der Arbeitsgruppe Tsunami Modellierung am Alfred-Wegener-Institut in Bremerhaven (AWI).

Zwischen 1991 und 1996 schrieb Herr Behrens seine Dissertation am AWI über “Adaptive Semi-Lagrangian Method for Shallow Water Equations”. Danach hat er noch zwei weitere Jahre als Postdoc am AWI verbracht.

Seine Habilitation [4] mit dem Titel “Adaptive Atmospheric Modeling - Key techniques in grid generation” hat er 2005 an der Universität München vorgelegt.

1.4 CliSAP

Der Exzellenzcluster “Integrated **C**limate **S**ystem **A**nalysis and **P**rediction” (CliSAP) wurde im Oktober 2007 gestartet [5].

Gefördert wird CliSAP im Rahmen der Exzellenzinitiative der Deutschen Forschungsgemeinschaft mit 32 Millionen Euro über fünf Jahre. Die Sicherung der Fördergelder gab den Anstoß den so genannten KlimaCampus an der Universität Hamburg zu gründen. Zum KlimaCampus gehören achtzehn Institute der Universität Hamburg sowie folgende außeruniversitäre Partner: das Max-Planck-Institut für Meteorologie, das Helmholtz-Zentrum Geesthacht und das Deutsche Klimarechenzentrum.

Die gesetzten Ziele des KlimaCampus sind die Untersuchung des Klimas, sowohl die Vorhersage des künftigen als auch die Erforschung der klimatischen Geschichte, nicht nur unter naturwissenschaftlichen Gesichtspunkten sondern auch unter gesellschaftswissenschaftlichen.

1.5 AWI

Das Alfred-Wegener-Institut für Polar- und Meeresforschung (AWI) wurde 1980 in Bremerhaven gegründet und gehört inzwischen der Helmholtz-Gemeinschaft an. “Das Alfred-Wegener-Institut forscht” laut der Startseite der Internetpräsenz [6] “in der Arktis, Antarktis und den Ozeanen der mittleren und hohen Breiten. Es koordiniert die Polarforschung in Deutschland und stellt wichtige Infrastruktur wie den Forschungseisbrecher Polarstern und Stationen in der Arktis und Antarktis für die nationale und internationale Wissenschaft zur Verfügung.” Die am AWI stattfindende Forschung umfasst dabei vor allem die Fachbereiche Geowissenschaften, Biowissenschaften und Klimawissenschaften.

2 Praxis der Softwareentwicklung

2.1 Wissenschaftliches Rechnen

Auf die Frage, was sein eigentliches Aufgabengebiet sei, antwortete Professor Behrens “Wissenschaftliches Rechnen” und erklärte weiter “wir rechnen, damit Ergebnisse herauskommen, und wir tun das wissenschaftlich, weil es keine Verfahren von der Stange gibt”. Insbesondere beschäftigt sich Professor Behrens mit so genannten adaptiven Gittern. Auf diesen aufsetzend werden dann Partielle Differentialgleichungen gelöst. Die Anwendungen in denen diese Verfahren in Herrn Behrens’ Gruppe zum Einsatz kommen sind zumal geowissenschaftliche Anwendungen und hier besonders Ozeananwendungen,

wie z.B. Tsunamis und auch Atmosphärenanwendungen, wie etwa der Spurenstofftransport. In seiner Laufbahn hat er sich zudem noch mit der Lösung großer Gleichungssysteme und der Computeroptimierung, sowohl Parallelisierung als auch die Optimierung auf Einzelcore-Maschinen, beschäftigt.

2.2 TsunAWI

2.2.1 Das Sumatra-Andaman-Erdbeben und GITEWS

Am 26. Dezember 2004 ereignete sich im Sundagraben ein schweres Erdbeben, das so genannte Sumatra-Andaman-Erdbeben. Das Erdbeben erreichte eine Stärke von 9,2 auf der Richterskala. Laut dem United States Geological Survey (USGS) entsprach die freigesetzte Energie etwa 23000 Hiroshima-Bomben [7]. Mehr als 225000 Menschen wurden durch den Tsunami getötet, über eine Million wurden durch ihn obdachlos [8].

Das Erdbeben wurde ausgelöst durch die Verschiebung tektonischer Platten und zwar der Indischen Platte unter die Burma-Platte. Der USGS schätzt die Länge des Bruchs auf etwa 1000 Kilometer. Die horizontale Verschiebung des Meeresbodens beläuft sich auf etwa 10 Meter, die Vertikale auf ein paar Meter [7].

Auf offenem Meer erreicht ein Tsunami nur etwa eine Höhe von 30 Centimetern, was dazu führt, dass er von Seeleuten kaum wahrzunehmen ist. Laut Zeitungsberichten erreichten die Tsunami-Wellen eine Höhe von bis zu 15 Metern [7].

Nach den tragischen Ereignissen des 26.12.2004 begannen weltweit Hilfgelder zu fließen und es wurden Hilfsprojekte ins Leben gerufen. Ein solches Hilfsprojekt ist das German-Indonesian Tsunami Early Warning System (GITEWS). Das Bundesministerium für Bildung und Forschung beauftragte die Helmholtz-Gemeinschaft mit der Entwicklung eines Tsunami-Früwarnsystems für den Indischen Ozean und hieraus entstand GITEWS. Deutschland kooperiert bei diesem Projekt mit Indonesien. Basis für diese Kooperation ist die am 14. März 2005 unterzeichnete Joint Declaration zwischen dem BMBF und dem indonesischen Forschungsministerium [9]. Folgende Institutionen sind an GITEWS beteiligt: Deutsches GeoForschungszentrum (GFZ) Konsortialführer, Alfred-Wegener-Institut für Polar- und Meeresforschung (AWI), Bundesanstalt für Geowissenschaften und Rohstoffe (BGR), Deutsches Zentrum für Luft- und Raumfahrt (DLR), GKSS Forschungszentrum Geesthacht, Gesellschaft für Technische Zusammenarbeit (GTZ), Konsortium Deutsche Meeresforschung (KDM), Leibniz-Institut für Meereswissenschaften (IFM-GEOMAR), United Nations University - Institute for Environment and Human Security (UNU-EHS)[10].

Für die Region ist ein Frühwarnsystem besonders wichtig, da von der Entstehung eines Tsunamis bis zum ersten Aufrollen auf die indonesische Küste nur etwa zwanzig Minuten vergehen [10]. Im Falle des Tsunamis vom 26.12.2004 vergingen 41 Minuten bis zum ersten Auflaufen von Wellen an der indonesischen Küste, aber man geht davon aus, dass künftige Erdbeben sich noch näher an der indonesischen Küste ereignen werden [11].

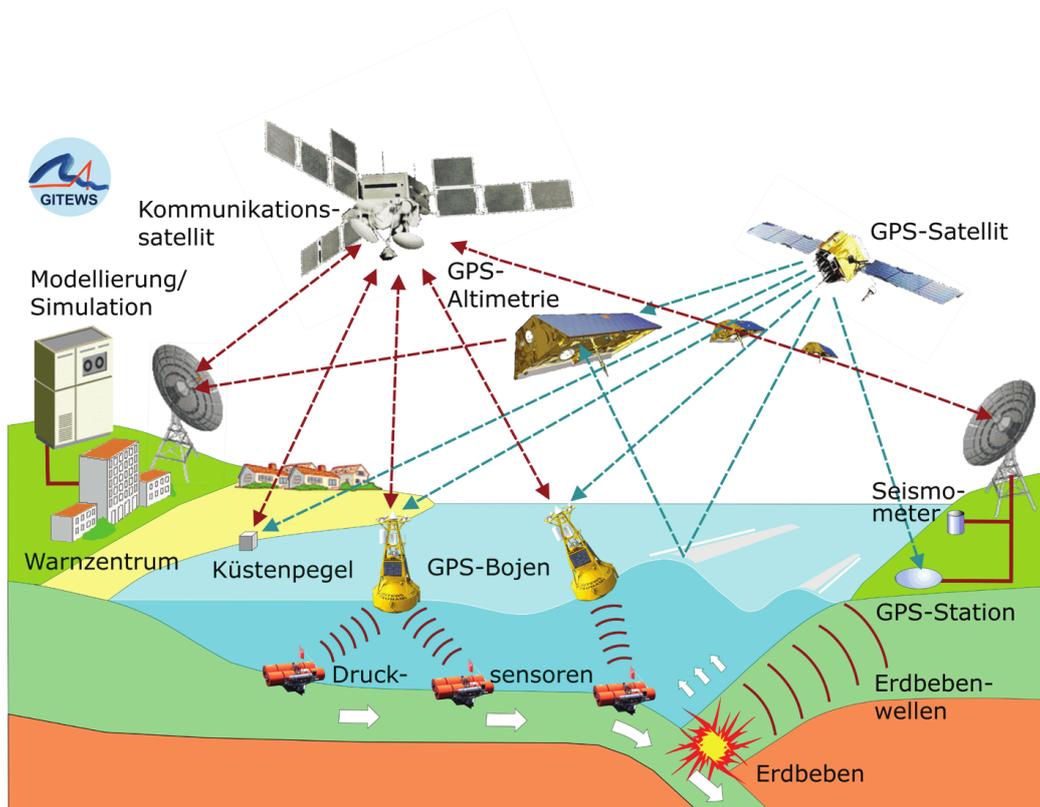


Abbildung 1: GITEWS Komponenten

Auf der Abbildung 1 sind die Komponenten des GITEWS schematisch dargestellt. Man sieht die gesamte Sensor- und Beobachtungstechnik, die vonnöten ist, um einen Tsunami möglichst früh zu erfassen. Für uns von besonderem Interesse ist jedoch das Warnzentrum am linken Bildrand. In diesem Warnzentrum läuft unter anderem die Simulationssoftware *TsunAWI*. Diese läuft offline und simuliert kontinuierlich unterschiedlichste Tsunami-Szenarien durch. Im Falle von aussergewöhnlichen Messdaten wird nun mithilfe des Entscheidungs-Unterstützungssystems ein vorausberechnetes Szenario ausgewählt, das den aktuell vorliegenden Messdaten am ehesten entspricht, um hieraus zu ermitteln, ob evakuiert werden muss, oder man Ent-

warnung geben kann.

2.2.2 Das Projekt

Das Projekt wurde 2005 am AWI begonnen. Herr Behrens war Leiter der Arbeitsgruppe und hatte ein Team von zehn Leuten unter sich. Von diesen haben einige programmiert, einige haben andere Sachen gemacht. Zu den Tätigkeiten von Professor Behrens im Rahmen der Arbeitsgruppe gehörte die Programmierung nicht. Wie er selbst sagt, bestand eine wesentliche Leistung von ihm darin, ein Konzept zu entwickeln, wie man die Unsicherheit des Vorhersageprozesses quantifizieren kann.

Zusätzlich wurde noch ein Softwareentwickler eingestellt, der aber nicht an der Simulationssoftware mitgeschrieben hat, sondern die Software entwickelt hat, die im Warnzentrum in Indonesien den Auswahlprozess des geeigneten Tsunami-Szenarios aus der Datenbank steuert.

2.3 Amatos

2.3.1 Adaptive Gitter

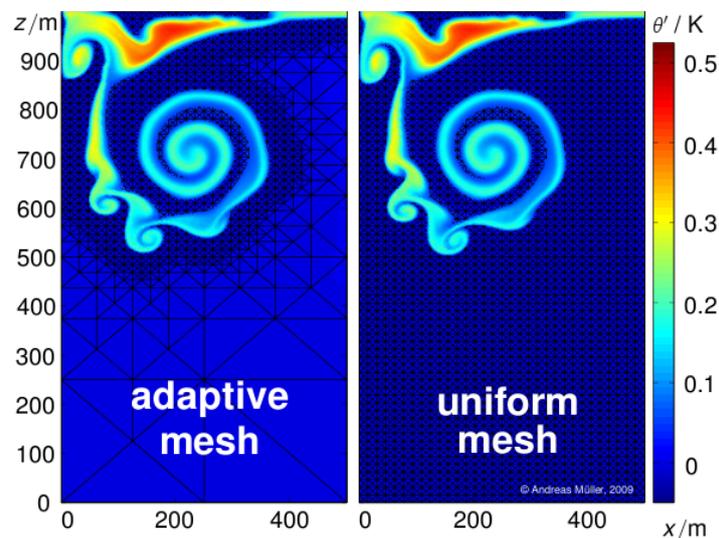


Abbildung 2: adaptive vs. uniform aus:[12]

Es ist schon erwähnt worden, dass adaptive Gitter eines der Hauptforschungsthemen von Professor Behrens sind. Hier soll nun kurz erklärt werden, was es damit auf sich hat. Es ist ein Standardverfahren in der Parallelisierung von Programmen Gitter über ein zu berechnendes Gebiet zu legen, da

die einzelnen Gitterzellen unabhängig voneinander parallel berechnet werden können, und nur an den Rändern der Gitterzellen Abhängigkeiten zwischen benachbarten Gitterzellen bestehen. Üblicherweise werden die Abstände der Gitterpunkte vor der Ausführung gleichmäßig gewählt und zur Laufzeit nicht mehr verändert. Dies ist für viele Probleme, wie z.B. die Modellierung von Wolken, nicht effizient, weil man es bei diesen Problemen mit Prozessen verschiedener Skalen zu tun hat, man spricht von vielskaligen Prozessen. Für manche dieser Prozesse ist eine besonders hohe Auflösung notwendig, bei anderen reicht eine niedrige Auflösung. Adaptive Gitter bieten nun die Möglichkeit den Abstand der Gitterpunkte den jeweils benötigten Auflösung anzupassen. Auf Abbildung 2 kann man sehr gut den Vergleich zwischen einem adaptiven und einem gleichförmigen Gitter sehen. Hier wird eine Blase warmer Luft umgeben von kalter Luft simuliert. Auf der linken Seite sind die Abstände der Gitterpunkte am Rand der Wolke besonders klein, da sich hier die eigentlich wichtigen Vorgänge abspielen.

In der Publikation [12] war die Simulation mit adaptiven Gittern sechsmal so effizient wie der herkömmliche Ansatz.

2.3.2 Das Projekt

Das Projekt *amatos* begann 1997. *Amatos* ist eine Funktionenbibliothek zur Generierung adaptiver Gitter. Als das Projekt begann, hatte Herr Behrens noch keinerlei Kenntnis von Methoden und Praxis der Softwaretechnik. Er hat geschaut, was die anderen so machen, und hat es nachgeahmt.

Das Projekt wurde nur begonnen, weil es zu der Zeit nichts Vergleichbares gab, das Professor Behrens nutzen hätte können. Andernfalls hätte er das Projekt nicht begonnen, das es “natürlich eine Mordsarbeit war”.

2.4 Praxis

2.4.1 Hardware

TsunAWI läuft im operationellen Einsatz auf IBM Power6 und Power7 Architekturen mit 4-8 Cores und etwa 16 GB Hauptspeicher pro Knoten sowie Intel Architekturen (SGI Altix) mit Intel Nehalem Prozessoren. Am CliSAP läuft die Software auf Sun-Clustter mit 8 Cores pro Knoten sowie einer IBM, bei der Professor Behrens sich nicht sicher war, ob sie 8 oder 16 Prozessoren pro Knoten. Ursprünglich wurde *amatos* auf einer SGI entwickelt, die laut Herrn Behrens sehr schönes Virtual Shared Memory hatte.

2.4.2 Sprachen

Die Software von Herrn Behrens Gruppe wird in Fortran90/95 geschrieben. Auch *amatos* und *TsunAWI* sind in Fortran90/95 geschrieben. *amatos* ist mit OpenMP parallelisiert, jedoch bisher nicht mit MPI. Professor Behrens ist persönlich daran gescheitert *amatos* mit MPI zu parallelisieren. Heute würde er dieses Unternehmen anders angehen und wäre vermutlich auch erfolgreicher. Allerdings lief es bisher hinreichend performant auf Shared-Memory-Architekturen.

Ein Ziel von Professor ist es, bei Gelegenheit die Programmiersprache Python zu erlernen, da dies ihm die Möglichkeit gäbe, einfacher mit dem Fortran-Quelltext zu interagieren, indem er "Frontend-Code" in Python schreiben könnte.

2.4.3 Verwendete Bibliotheken und Compiler

Die Software ist meist generisch in Fortran programmiert und weist nur wenige Abhängigkeiten zu anderen Bibliotheken auf. Ausnahmen sind vor allem die NetCDF-Bibliothek und Bibliotheken für lineare Algebra. NetCDF wird in den Geowissenschaften standardmässig als Datenformat verwendet. Was Bibliotheken für lineare Algebra angeht, so versucht man stets proprietäre Bibliotheken des Compiler-Herstellers zu benutzen, da diese in einer Weise optimiert sind, wie man es selbst gar nicht hinbekäme. Als Compiler kommen vor allem der Intel Fortran Compiler als auch der GNU Fortran Compiler zum Einsatz. Aufgrund der wenigen Abhängigkeiten ist es nicht besonders schwierig die Software auf andere Maschinen zu portieren, nur bei Linux ist das Einbinden der Bibliotheken manchmal ein bisschen blöd.

2.4.4 Code Editing

Die Wahl des Text Editors ist einem jeden selbst überlassen. Professor Behrens persönlich benutzt aktuell den XCode Editor auf dem Mac. Integrierte Entwicklungsumgebungen finden augenblicklich keine Verwendung, obwohl Herr Behrens gehört hat, dass sowohl Eclipse als auch XCode inzwischen ganz brauchbar sein sollen für die Fortran-Entwicklung. Er würde sich wünschen endlich zu verstehen, wie integrierte Entwicklungsumgebungen konfiguriert werden.

2.4.5 Debugging

Es wird eine kommandozeilenorientierter Debugger verwendet. Zum Debuggen paralleler Programme benutzt Professor Behrens den graphischen De-

bugger Totalview. Herr Behrens hat sich aufgrund des hohen Preises von Totalview die verfügbaren Debugger angeschaut, aber nach seiner Einschätzung ist Totalview der einzig brauchbare. “Ohne den hätte ich viele Sachen gar nicht machen können.”

2.4.6 Testing & Validierung

Professor Behrens ist Anhänger der testgetriebenen Entwicklung. Das meint nicht, dass tatsächlich für jede einzelne Subroutine zunächst ein Unittest implementiert wird und erst dann die eigentliche Subroutine selbst, sondern dass vor allem Herr Behrens selbst immer mit Hinblick auf bestehende Testcases entwickelt. Dies ist eine Vorgehensweise, die er auch stets seinen Studenten zu vermitteln sucht.

Wäre das geeignete Instrumentarium vorhanden, ist nicht ausgeschlossen, dass man tatsächlich klassisch testgetrieben entwickeln würde. Das heißt, wenn es die Möglichkeit gäbe in integrierten Entwicklungsumgebungen Fortran testgetrieben zu entwickeln, im Augenblick ist der Overhead durch fehlende Tool-Unterstützung jedoch zu groß.

Bisher wird Validierung so betrieben, dass man sich eine handvoll Testcases baut, von denen man Lösungen kennt, oder vereinfacht sie so sehr, dass es glaubhafte Referenzlösungen gibt. Dies fängt mit ganz einfachen Sachen an. Man schaut z.B., ob der Code einfache Erhaltungseigenschaften erfüllt, wie dass etwa die Masseerhaltung gewährleistet ist. Dann gibt es vereinfachte Tests, für die es eine analytische Lösung gibt. Man kann z.B. die lineare Wellenausbreitung auf dem Papier berechnen. Außerdem gibt es Tests mit Überflutungsrandbedingungen, wo z.B. eine Welle den Strand hinaufläuft, wofür man im linearen Regime analytische Lösungen kennt. Ferner läßt sich noch mit Messwerten aus der Realität und dem Labor validieren. Hieran validiert man dann den Code. Nun ändert man den Parameterbereich und man kennt keine Lösungen für diese veränderten Gleichungen. Es gibt keine Verifikation. “Man muss immer skeptisch sein” und Plausibilitätstests machen, sich fragen: “Kann das plausibel möglich sein?”

Besonders intensiv hat Herr Behrens sich mit diesen Fragen im Tsunami-projekt in Bremerhaven auseinandergesetzt, da es bei diesem Projekt darum ging Menschenleben zu retten.

2.4.7 Versionskontrolle

Amatos wurde zunächst “manuell” versioniert, wie man an Abbildung 3 sehen kann. Inzwischen unterliegt der *amatos* der Versionskontrolle mit Subversion (SVN) [13]. Auch *TsunAWI* wird mit SVN versioniert.

```

!*****
! USES:
!   MISC_globalparam, MISC_error, FEM_define, FEM_handle, FEM_create,
!   FEM_gridmanag
! LIBRARIES:
!
! REFERENCES:
!   the grid refinement strategy has been taken from
!   M.C. Rivara: "Algorithms for Refining triangular grids suitable
!   for adaptive and multigrid techniques", INT. JOU. NUM. METH. ENG.,
!   Vol. 20, pp. 745-756 (1984)
!   E. Baensch: "Local mesh refinement in 2 and 3 dimensions",
!   IMPACT COMP. SCI. ENG., Vol. 3, pp. 181-191 (1991)
! VERSION(S):
!   1. original version                j. behrens      8/96
!   2. now with edges                  j. behrens     10/96
!   3. now with time and handles, clearup_grid and
!   grid_timeduplicate added, renamed  j. behrens     10/96
!   4. grid_clean added, clearup_grid
!   modified                           j. behrens     5/97
!   5. hashing tables                  j. behrens     7/97
!   6. many changes and bug-fixes      j. behrens     8/97
!   7. grid_timetoggle moved to GRID_api j. behrens     9/97
!   8. extracted FEM_create            j. behrens    10/97
!   9. parallel version                j. behrens     1/98]
!  10. find_resolvpatch added          j. behrens     1/98
!  11. find_resolvpatch alg. changed   j. behrens     2/98
!  12. changed edge data struct.       j. behrens     2/98
!  13. updated for amatos version 1.0   j. behrens    11/2000
!
!*****
MODULE FEM_gridgen
USE MISC_globalparam
USE MISC_error
USE FEM_define
USE FEM_handle
USE FEM_create
USE FEM_dataretrieve
USE FEM_signature
USE GRID_utils
!--daFlo--
USE IO_emit
USE FEM_boundary

```

Abbildung 3: Auszug aus der Datei FEM_Gridgen.F90

2.4.8 Projektmanagement

Amatos wird mit trac [14] verwaltet. Am AWI wurde GForge [15] benutzt. Am CliSAP wird für das Projektmanagement redmine [16] genutzt. Der Umfang in dem das Ticketing, das diese Systeme bereitstellen, genutzt wird, ist abhängig von der Zahl der Entwickler, die gleichzeitig an einem Modul arbeiten. Es gab Zeiten in denen bis zu fünf Entwickler gleichzeitig an *amatos* gearbeitet haben. In diesen Zeiten wurde das Ticketing intensiv genutzt. Auch hat eine Zeit lang ein Consultant an der Parallelisierung von *amatos* mitgearbeitet. Im Augenblick wird das Ticketing bei *amatos* nicht genutzt.

2.4.9 Codequalität & Code Review

Als Professor Behrens begann *amatos* zu entwickeln, war ihm schon bewusst, dass er es über lange Zeit hinweg nutzen würde, und er hat demnach Wert auf Modularisierung gelegt, so dass etwa auch mal Studenten oder Doktoranden einzelne Module weiterentwickeln können, ohne immer den Kern von *amatos* anfassen zu müssen.

Bei *TsunAWI* waren die Anforderungen ganz andere, da es sich hier um Software handelt, die operationell eingesetzt wird, und somit eine gewisse Nachweispflicht besteht. Die Arbeitsgruppe am AWI sah sich zu Beginn des Projekts vor die Aufgabe gestellt, sich selbst Standards zu setzen. So wurden Coding Guidelines erstellt, die sich an denen des UK Met Office [17] orientieren. Jedoch wurde nicht streng darauf geachtet, dass diese eingehalten werden. Es gab keine institutionalisierte Code Review. Dennoch hat Herr Behrens über den Code geschaut und musste feststellen, dass dieser anfänglich “Kraut und Rüben” war, so dass er sich gerne an einer Refaktorisierung versucht hätte, wozu er aber keine Zeit hatte.

2.4.10 Codedokumentation

Bei *amatos* hat jedes Modul einen ausführlichen Header, in dem beschrieben wird, was es tut. Auch innerhalb des Quelltextes finden sich zahlreichen Kommentare. Das Handbuch zu *amatos* ist allerdings auf dem Stand von 2003 und wird unabhängig vom Quelltext gepflegt, da Professor Behrens, als er begann das Programm zu entwickeln, nicht die Softwareanwendungen bekannt waren, die es ermöglichen den Quelltext und die dazugehörige Dokumentation in ein und derselben Datei zu verwalten. Für das *TsunAWI* Projekt wurde dann Protex [18] genutzt.

2.4.11 Ausblick

Im Zuge einer aktuellen Kooperation mit der Universität Mainz, bei der es um die Modellierung von Wolken geht, kann es sich als notwendig erweisen, dass *amatos* auf das Message Passing Interface (MPI) portiert wird. Das CliSAP hat sich kürzlich eine CUDA Maschine angeschafft. Der Aufwand die bestehende Fortran-Simulationssoftware umzuschreiben, um die Grafikprozessoren zu nutzen, ist indiskutabel. Man wird den Fortran Compiler der Portland Group evaluieren, der es ermöglicht die Grafikprozessoren ausschließlich über Compilerdirektiven anzusteuern, so dass die notwendigen Änderungen am Quelltext minimal sind. Ein weiteres Vorhaben, welches jedoch nicht ganz oben auf der Prioritätenliste steht, ist das Testing eventuell zu automatisieren. So dass Testläufe maschinell ausgeführt werden sobald Änderungen an der Software in das Versionierungssystem eingecheckt werden.

2.4.12 Äußere Umstände der SiW

Generell ist der Zeitdruck bei der Softwareentwicklung in der Wissenschaft nicht so ein großes Problem. Es kommt allerdings vor, dass man schnell ein Paper fertig machen will, und dann wird auch schon mal “quick’n’dirty” entwickelt. Man nimmt sich vor die Software dann später noch zu verbessern oder gibt es einem Studenten und sagt: “Mach mal schön”. Aber das passiert dann natürlich nicht.

Nach Aussage von Professor Behrens ist Software zu schreiben in Anbetracht der Karriereoptimierung nicht sinnvoll. Ein Traum von Herrn Behrens, seitdem er angefangen hat sich mit Simulationssoftware zu beschäftigen, wäre es, eine graphische Benutzeroberfläche zu haben, die es einem ermöglicht, nicht jedes mal vor einem Simulationslauf Änderungen am Fortran-Code vornehmen zu müssen, sondern sich die Simulationsläufe mit der Maus zusammenzuklicken. Diese Software zu schreiben ist jedoch nicht karriereförderlich, da dies nichts grundlegend Neues wäre und es in Herrn Behrens’ Arbeitsgebiet “niemanden interessiert, ob man irgendwelche Software geschrieben hat”.

2.5 Kurs

Wichtig zu erwähnen ist noch, dass Professor Behrens selbst einen einwöchigen Blockkurs zur Softwaretechnik angeboten hat. Inhalt des Kurses waren unter anderem das Testing, die Versionierung und das Debugging. Anfänglich waren etwa fünfzehn Personen angemeldet, von denen ungefähr zehn dann tatsächlich teilgenommen haben, zumeist Doktoranden.

3 Abschluss

3.1 Zusammenfassung

Es scheint außer Frage zu stehen, dass Professor Behrens sich des Nutzens der Softwaretechnik bewusst ist, wenn es auch sicher noch Raum für Verbesserung gibt.

Allein die Tatsache, dass er, als er an *amatos* zu arbeiten begann, keinerlei Kenntnis der Softwaretechnik hatte und inzwischen selbst bereit ist anderen solche Kenntnis zu vermitteln, zeigt nicht nur eine bemerkenswerte Entwicklung sondern auch welchen Stellenwert Herr Behrens der Softwaretechnik einräumt.

Darüberhinaus lässt sich hieraus ablesen, dass Herr Behrens sowohl sieht, dass die Softwaretechnik für Naturwissenschaftler von Nutzen ist, als auch, dass es unter Naturwissenschaftlern durchaus Bedarf für softwaretechnischen Unterricht gibt.

Literatur

- [1] <http://www.amatos.info> (Stand: 05.02.2011).
- [2] J. Behrens, N. Rakowsky, W. Hiller, D. Handorf, M. Läuter, J. Pöpke, and K. Dethloff, “*amatos: Parallel adaptive mesh generator for atmospheric and oceanic simulation*,” *Ocean Modelling*, vol. 10, no. 1-2, pp. 171 – 183, 2005. The Second International Workshop on Unstructured Mesh Numerical Modelling of Coastal, Shelf and Ocean Flows.
- [3] http://wr.informatik.uni-hamburg.de/teaching/wintersemester_2010_2011/softwareentwicklung_in_der_wissenschaft.
- [4] J. Behrens, “Adaptive atmospheric modeling - Key techniques in grid generation, data structures, and numerical operations with applications.” Thesis of Habilitation, 2005.
- [5] http://www.klimacampus.de/wer_wir_sind.html (Stand: 08.03.2011).
- [6] <http://www.awi.de/de/startseite/> (Stand: 08.03.2011).
- [7] N. Geographic. http://news.nationalgeographic.com/news/2004/12/1227_041226_tsunami.html (Stand:), 2004.

- [8] E. Geist, V. Titov, and C. Synolakis, “Tsunami: Wave of change.” <http://www.scientificamerican.com/article.cfm?id=tsunami-wave-of-change&ref=sciam> (Stand: 07.03.2011), 2006.
- [9] http://www.gitews.de/fileadmin/documents/content/press/GITEWS_Flyer_DE2008.pdf (Stand: 05.02.2011).
- [10] http://www.gitews.de/index.php?id=22&L=1%2Fphpwcms%2Fincl...alendar%2Fid.txt%3F%3F%3F%2F%2Ftools%2Fsend_reminders.php%3FnoSet%3D0%2F%2Ftools%2Fsend_reminders.php%3FnoSet%3D0 (Stand: 08:03.2011).
- [11] <http://www.gitews.de/index.php?id=35&L=1%2Fphpwcms%2Finclu...ert%2Fmvcw.php%3Fstep%3D1%20%20%2Fvwar%2Fconvert%2Fmvcw.php%3Fstep%3D1> (Stand: 08.03.2011).
- [12] A. Mueller, J. Behrens, F. Giraldo, and V. Wirth, “An adaptive discontinuous galerkin method for modeling cumulus clouds.,” in *Fifth European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2010*, June 2010. (Submitted).
- [13] Apache Software Foundation, “Apache Subversion.” <http://subversion.apache.org> (Stand: 08.03.2011), 2005.
- [14] The Trac Project, “trac Integrated SCM & Project Management.” <http://trac.edgewall.org> (Stand: 08.03.2011), ?
- [15] <http://gforge.org/gf/>.
- [16] <http://www.redmine.org/>.
- [17] http://research.metoffice.gov.uk/research/nwp/numerical/fortran90/f90_standards.html (Stand 08.03.2011).
- [18] A. da Silva and W. Sawyer, “Protex Distribution (Version 2.0).” <http://gmao.gsfc.nasa.gov/software/protex/> (Stand: 08.03.2011), 2005.