

ATLAS Vista: Ausbau der Ergebnisdarstellung und Datenverwaltung

Abschlussbericht zum
Vorhaben RS 1179



Gesellschaft für Anlagen-
und Reaktorsicherheit
(GRS) mbH

Abschlussbericht/ Final Report

Reaktorsicherheitsforschung-
Vorhabens Nr.:/
Reactor Safety Research-Project No.:
RS1179

Vorhabensitel / Project Title:
ATLAS Vista: Ausbau der Er-
gebnisdarstellung und Daten-
verwaltung

ATLAS Vista: Extended Meth-
ods for Visualization and Data
Management

Autor / Author:
T. Voggenberger

Berichtszeitraum / Publication Date:
November 2010

Anmerkung:

Das diesem Bericht zugrunde lie-
gende F&E-Vorhaben wurde im
Auftrag des Bundesministeriums
für Wirtschaft und Technologie
(BMWi) unter dem Kennzeichen
RS1179 durchgeführt.

Die Verantwortung für den Inhalt
dieser Veröffentlichung liegt beim
Auftragnehmer.

GRS - A - 3577

Kurzfassung

Das grundlegende Ziel der ATLAS-Entwicklung ist es, für Rechenprogrammsysteme in der Reaktorsicherheit eine einheitliche Simulationsumgebung zur Verfügung zu stellen, in der die Ergebnisdaten anschaulich visualisiert werden und die Simulation interaktiv beeinflusst werden kann. Im Rahmen des Vorhabens RS1179 wurde der Analysesimulator ATLAS mit neuen Funktionen für die Ergebnisdarstellung und die Datenverwaltung entsprechend dem Stand der Technik erweitert.

Für ATLAS wurden neue Darstellungsmethoden für ortsabhängige Simulationsdaten entwickelt, die besonders für Containment- oder Reaktorkernmodelle benötigt werden und die Daten anschaulich im räumlichen Zusammenhang visualisieren können. Es wurde ein 3D-Modell des Reaktorgebäudes eines Siedewasserreaktors erstellt, in dem in einer Virtual Reality Darstellung zeitveränderliche Daten angezeigt werden können. Für den Reaktorkern können die lokalen Daten in axialen Schnitten und Profilen in intuitiven Grafiken analysiert werden. Das dynamische Verhalten der Brenn- und Steuerstäbe bei schweren Störfällen ist nun in automatisch erzeugten Bildern verfolgbar.

Die wachsende Komplexität und Detaillierung in der Simulation führt zu einem enormen Datenaufkommen, das sowohl zur Laufzeit der Simulationsprogramme als auch bei der Langzeitspeicherung angemessen behandelt werden muss. Die Datenaustausch und die Datenhaltung in ATLAS wurden daher auf Basis von Dateien mit normierten wissenschaftlichen Datenformaten erneuert. Diese werden einheitlich für alle Simulationsmodelle verwendet, sind leicht erweiterbar und garantieren einen sicheren und schnellen Zugriff für nahezu beliebige Rechner- und Softwareplattformen. Die Archivierung von Daten wird mit einer Kombination von Datenbank- und Archivierungssoftware sowie spezialisierten Festplatten- und Bandsystemen durchgeführt. Damit lassen sich sehr große Datenmengen zuverlässig speichern, mit der verknüpften Metainformation und Suchabfragen wiederauffinden und bei Bedarf wiederherstellen

Zusätzlich wurden viele weitere Verbesserungen in ATLAS beim Visualisierungsmodul und Bildeditor APG durchgeführt und alle Änderungen in einer neuen Programmversion integriert. Diese wird begleitet von Bedienhandbüchern, einem Benutzerforum und Systemen zur Codeverwaltung. Insgesamt ergeben damit sich für die ATLAS Anwender erhebliche Vorteile bei der Durchführung von Sicherheitsanalysen in Bezug auf Qualität und Effizienz.

Abstract

The main goal of the ATLAS development is to provide a common simulation interface for reactor safety analysis codes, which supports the visualization of results and allows the interactive control of the simulation process. Within the framework of research project RS1179 the analysis simulator ATLAS was improved by new functionality in visualization and data management taking into account the state of the technology.

For spatial simulation data, new visualization methods in ATLAS have been developed which are able to depict clearly the local dependency, which are especially useful for containment and reactor core models. A 3D-model of the reactor building of a BWR was generated, which is used in a virtual reality environment and can depict time dependent data. Spatial data of the reactor core may be analyzed by meaningful graphics showing axial cuts and profiles. The dynamic behavior of fuel and control rods during severe accidents can now be tracked in automatically generated pictures.

The increasing complexity and particularity in the simulation models lead to a huge data amount. This has to be taken into account both during the simulation runs and the long term storage of the data. Therefore the data exchange and management in ATLAS were redesigned based on a file level approach using standard scientific data formats. They are employed in a common way for all simulation models, may be easily extended and guarantee a reliable and fast access for many hard and software platforms. The archival storage of the data is managed by a combination database and archival software and specialized magnetic disc and tape systems. This approach is able to store very high data volumes reliable. Using the attached metadata and search algorithms the data may be easily identified and retrieved on demand.

In addition to the mentioned work many other enhancements of ATLAS have been performed in the visualization program and the picture editor APG. All changes are included into a new release version of the ATLAS software. New user manuals, a WEB based user forum and code management systems are also available. Overall the ATLAS users get an improved quality and efficiency in the performance of reactor safety analyses.

Inhaltsverzeichnis

1	Einleitung	1
2	Zielsetzung	2
3	Weiterentwicklung von ATLAS	5
3.1	Erweiterung des Visualisierungsmoduls.....	5
3.1.1	Darstellung von dynamischen Daten in virtueller 3D-Umgebung.....	5
3.1.1.1	Auswahl und Evaluierung von Softwarewerkzeugen	5
3.1.1.2	Erstellung eines 3D-Modells für das Reaktorgebäude eines SWR-69.....	8
3.1.1.3	Visualisierung von Simulationsdaten in Virtual Reality (VR)	15
3.1.1.4	Zusammenfassende Bewertung des Verfahrens	18
3.1.2	Erweiterte Darstellungsoptionen für Zeitkurven und bei der Kerndatenvisualisierung	19
3.1.2.1	Darstellungsoptionen für Zeitkurven	19
3.1.2.2	Kerndatenvisualisierung	32
3.1.2.3	Ortsdiagramme für ATHLET	36
3.1.3	Darstellung der Nodalisierung und der Brennstäbe in Verbindung mit dynamischen Daten für ASTEC	39
3.1.3.1	Programmbeschreibung.....	39
3.1.3.2	Programmentwicklung.....	40
3.1.3.3	Programmstruktur	41
3.1.3.4	Berechnungsteil	41
3.1.3.5	Bedienungsteil.....	42
3.1.3.6	Grafikteil	42
3.1.3.7	Programmbedienung.....	42
4	Verwaltung, Verteilung und Archivierung von Daten	54
4.1	Konzept für die Verwaltung und den Zugriff der Simulationsdaten und der Benutzereingriffe	54
4.1.1	Einsatz eines Datenbanksystems	54
4.1.2	Dateibasierter Ansatz.....	55

4.1.3	Grundsätzliche Untersuchungen zum Dateiformat NetCDF.....	57
4.2	Implementierung der in ATLAS benötigten Funktionen.....	59
4.3	Senden von Nachrichten an die Prozessmodelle	60
4.4	Anpassung der Startoberfläche und Startprozeduren	61
4.5	Interaktives Tool zur schnellen Auswertung von NetCDF-Dateien	63
4.6	Erreichte Vorteile gegenüber der Server basierten Version.....	65
4.7	Archivierung von Versuchs- und Simulationsdaten.....	66
4.7.1	Anforderungen	66
4.7.2	Realisierung	66
4.7.2.1	Datenbibliotheken	67
4.7.2.2	Definition der Metainformationen	67
4.7.2.3	Eingabe der Metainformationen	69
4.7.3	Bedienung von SharePoint-Listen.....	71
4.7.3.1	Die Speicherung der Daten.....	74
4.7.4	Test des Systems.....	76
4.7.5	Beschränkungen	77
5	Programmoptimierung und Dokumentation.....	78
5.1	Optimierung des Visualisierungsprogramms.....	78
5.1.1	Speicherung von Visualisierungsoptionen	78
5.2	Optimierung des Bildeditors APG	82
5.2.1	Farbänderungen nach Farbskalen	82
5.2.1.1	Einfache Farbskala	82
5.2.1.2	Umschaltbare Farbskala	88
5.2.2	Steuerung der Sichtbarkeit von Bildelementen	91
5.2.3	Zusatzinformation zur Dokumentation der ATLAS-Bilder.....	93
5.3	Benutzerhandbücher.....	95
5.4	Benutzerforum.....	97
5.5	Qualitätssicherungsmaßnahmen	99
5.5.1	Versionsmanagement	99
5.5.2	Fehlerverfolgung	101
5.5.2.1	Überblick	101

5.5.2.2	Anmeldung	102
5.5.2.3	Übersicht und grundsätzlicher Aufbau	103
5.6	Persönliche Einstellungen (My Account).....	112
5.6.1	Installation	114
6	Zusammenfassung und Ausblick.....	117
6.1	ATLAS Übersicht.....	117
6.2	Datenmanagement.....	117
6.3	Visualisierung.....	119
6.4	Qualitätssicherung	121
6.5	Grenzen und neue Anforderungen.....	122
7	Literaturverzeichnis	124
Anhang 1: Datenmanagement bei den Trendkurven in ATLAS.....		A-1
Anhang 2: Syntax zur Beschreibung der Trendsets in der ATLAS Bilddatei		A-3
Anhang 3: Beispiel für den Befehl „MULTISCALE“ in einer APG-Bilddatei.....		A-16
Anhang 4: Namensgebung und Parameterübergabe im Modul „NetCDF“		A-18
Anhang 5: Schnittstellenprogramme in den Prozessmodellen (z. B. ATHLET) und Monitor für Datenverwaltung und Datenaustausch		A-20

Verteiler

Abbildungsverzeichnis

Abb. 3-1	Grundriss eines SWR-69 bei 0.0 m Höhe.....	9
Abb. 3-2	Querschnitt durch das Reaktorgebäude von KKI-1	10
Abb. 3-3	Gestaltung des Hauptfensters von Google-SketchUp	11
Abb. 3-4	Abbildung wichtiger Teile des Reaktorgebäudes eines SWR-69	12
Abb. 3-5	Abbildung des unteren Stockwerkes mit zugehörigen Komponenten eines SWR-69.....	13
Abb. 3-6	Mittellinie eines Leitungsverlaufs in SketchUp.....	14
Abb. 3-7	3D-Leitung entlang einer Mittellinie in SketchUp	14
Abb. 3-8	Auszug einer 3D-Datei aus SketchUp für Datenvisualisierung in Vizard	15
Abb. 3-9	Nodalisierungsschema von COCOSYS für das Reaktorgebäude.....	16
Abb. 3-10	Visualisierung einer niedrigen Raumtemperatur im 3D- Reaktorgebäude (Viewport1, Beginn der Transiente)	17
Abb. 3-11	Visualisierung einer hohen Raumtemperatur im 3D-Reaktorgebäude (Viewport2, Ende der Transiente).....	18
Abb. 3-12	Auswahl-Baum für Elemente aller Diagramme im Trend Property Editor	21
Abb. 3-13	Header-Dialog im Trend Property Editor	22
Abb. 3-14	Dialog zur Bestimmung der Schriftart im Trend Property Editor	23
Abb. 3-15	Footer-Dialog im Trend Property Editor.....	24
Abb. 3-16	Legend-Dialog im Trend Property Editor	25
Abb. 3-17	Time-Axis-Dialog im Trend Property Editor	27
Abb. 3-18	Y-Axis-Dialog im Trend Property Editor.....	29
Abb. 3-19	Curve-Dialog im Trend Property Editor.....	31
Abb. 3-20	Beispiel für die Darstellung einer Leistungsverteilung im Reaktorkern	33
Abb. 3-21	Auswahl von Einzelkanaldaten im Reaktorkern.....	36
Abb. 3-22	Darstellungsoptionen von Ortsverteilungen.....	37
Abb. 3-23	Axialdiagramm einer Leistung im Reaktorkern	38
Abb. 3-24	Axialdiagramm einer Leistungsdichte im Reaktorkern.....	38
Abb. 3-25	Hauptfenster von ASIG	44
Abb. 3-26	Verwendung eines ASIG-Bildes für Thermohydraulikobjekte in ATLAS	45
Abb. 3-27	Verwendung eines ASIG-Bildes für Brennstabobjekte in ATLAS	46

Abb. 3-28	Globale Optionen zur Darstellung von Objekten in ASIG	47
Abb. 3-29	Schematische Darstellung der Verknüpfung von Objekten in ASIG ...	48
Abb. 3-30	Optionen bei der Darstellung von einzelnen Objekten in ASIG	51
Abb. 4-1	Startoberfläche von ATLAS – File Dialog	63
Abb. 4-2	Formular mit den Eingabefeldern für die Metainformationen für Daten	70
Abb. 4-3	Dropdown-Liste der Anlagen und der Störfallgruppen.....	71
Abb. 4-4	SharePoint-Liste mit den Metainformationen von Simulationsrechnungen	72
Abb. 4-5	Menü zur Erzeugung neuer Dokumente und Ordner.....	73
Abb. 4-6	Das Updatemenü der Dokumente	73
Abb. 4-7	Menü zum Sortieren und Filtern der Metainformationen	74
Abb. 4-8	FAS2050 GRS (Garching)	75
Abb. 4-9	Dialog zum Speichern der Formulareingabe	77
Abb. 5-1	Optionen zur Darstellung verschiedener Zeitpunkte in ATLAS	80
Abb. 5-2	Beispieldarstellung eines Trendsets in ATLAS.....	81
Abb. 5-3	Bildeditor APG - Farbskalenerzeugung	83
Abb. 5-4	Definition von einzelnen Farbskalen.....	84
Abb. 5-5	Scale Definition Details.....	86
Abb. 5-6	Dialogfenster zur Farbauswahl	86
Abb. 5-7	Zuordnung einer Farbskala zu einem Grafikelement.....	87
Abb. 5-8	Dialog zur Definition einer Multifarbskala	89
Abb. 5-9	Auswahlmenü für häufig benötigte Multifarbskalen	90
Abb. 5-10	Definition einzelner Farbskalen und Variablen für Multifarbskalen....	91
Abb. 5-11	Definition der Funktion „Visibility Script“ im „Pick-Dialog“	92
Abb. 5-12	Ansteuerung und Objektauswahl für „Visibility Script“	93
Abb. 5-13	ATLAS-Graphik mit Informationszeile.....	94
Abb. 5-14	Erstellung des APG User’s Manual in Fast-Help	96
Abb. 5-15	Hilfe mit Tooltip und Statuszeile	97
Abb. 5-16	Willkommenseite der ATLAS User‘ Area	98
Abb. 5-17	Verwaltung des Benutzer und Beiträge in der ATLAS User‘ Area.....	99
Abb. 5-18	Bedienoberfläche von RapidSVN	100
Abb. 5-19	Anmeldeseite von Mantis.....	102
Abb. 5-20	Persönliche Übersichtsseite in Mantis	103
Abb. 5-21	Standardseite des Mantis Systems	105
Abb. 5-22	Problembereicht mit „Reporter“-Status	107

Abb. 5-23	Senden einer Benachrichtigung aus Mantis	108
Abb. 5-24	Maske zur Generierung eines Problembereichs.....	109
Abb. 5-25	Persönliche Daten in Mantis verwalten.....	112
Abb. 5-26	Benachrichtigungen in Mantis verwalten	113
Abb. 5-27	Computer-Profil in Mantis verwalten.....	114
Abb. 5-28	ATLAS Setup-Projekt.....	115
Abb. 5-29	ATLAS Setup Welcome Dialog.....	116
Abb. 6-1	Kopplung der ATLAS Module	118

1 Einleitung

Rechenprogrammsysteme zur Simulation von Stör- und Unfallabläufen im Kühlkreislauf und im Sicherheitsbehälter von KKW sind nach langjähriger Entwicklung zu komplexen Systemen herangewachsen, die Modelle für alle wichtigen technischen und physikalischen Prozesse enthalten. Besondere Bedeutung haben die Simulation des Reaktor-kühlkreislaufs mit Thermohydraulik und Wärmeübertragung, die Komponenten- und Leittechnikmodelle, die Neutronenkinetik, Kernzerstörungsmodelle und die Nachbildung von Druckaufbau- und Spaltproduktverhalten im Containment. Das grundlegende Ziel der ATLAS-Entwicklung ist es, für diese Modelle eine einheitliche Simulationsumgebung zur Verfügung zu stellen, in der die Ergebnisdaten anschaulich visualisiert werden und die Simulation interaktiv beeinflusst werden kann.

Zu diesem Zweck wurden in verschiedenen vorangehenden ATLAS Entwicklungsvorhaben grundlegende Methoden für ein effizientes Pre- und Postprozessing der Reaktorsicherheitscodes bereitgestellt. Diese umfassen eine grafische Bedienoberfläche und eine Ergebnisdarstellung in vielfältigen Diagrammen und Prozessbildern. Für die Erstellung der Bilder sind spezielle Bildeditoren verfügbar. Zusätzlich sind weitere Hilfsmittel für Reaktorsicherheitsuntersuchungen vorhanden, beispielsweise können teilweise Modelldaten erstellt und geprüft werden, Bedienprozeduren simuliert werden und Unsicherheitsanalysen in ATLAS durchgeführt werden.

Zur Abdeckung des weiten Anwendungsspektrums sind in ATLAS das Reaktoranlagenmodell ATHLET /ATH 06/, das Containmentmodell COCOSYS sowie die Integralcodes ASTEC /ALL 05/ und MELCOR /GAU 00/ in einer einheitlichen Bedienumgebung integriert.

ATLAS wird gegenwärtig von Forschungs- und Aufsichtsorganisationen national und international für einen großen Anwendungsbereich eingesetzt. Besonders erwähnenswert sind dabei die im Auftrag des BMU erstellten detaillierten Störfallsimulatoren für deutsche KKW, die Durchführung von „Human Factor Analysen“, das Training von Operateuren im Bereich schwerer Störfälle und die Anwendung als Standardvisualisierungswerkzeug für ATHLET.

2 Zielsetzung

Mit der fortschreitenden Entwicklung der Simulationscodes in Richtung immer detaillierter Modelle und höherer örtlicher Auflösungen, z. B. mehrdimensionale Simulation des Reaktorkerns, ergaben sich auch für ATLAS erhöhte Anforderungen in der Visualisierung und bei der Handhabung der Daten. Insbesondere die Darstellung ortsbezogener Information entsprach zu Beginn des Vorhabens teilweise nicht mehr den Stand der Technik. Dadurch entsprachen in einigen Fällen die Anschaulichkeit und die Interpretierbarkeit der Daten sowie die Qualität bei Dokumentation der Ergebnisse nicht den Anforderungen. Weiterhin gab es Probleme bei der Verteilung, dem Zugriff und der Verwaltung der immer stärker wachsenden Simulationsdatensmengen. In Einzelfällen musste die Zahl der darstellbaren Größen eingeschränkt werden oder der Zugriff auf ältere Daten war nicht oder nur mit erheblichem Aufwand möglich.

Das Ziel der Weiterentwicklung von ATLAS im Vorhaben RS1179 bestand daher darin, dem Stand der Technik auf dem Gebiet der Ergebnisdarstellung und des Datenmanagements mit der Einführung neuer Methoden Rechnung zu tragen. Damit konnten Qualität und die Effizienz der Analysen weiter erhöht werden.

Bei den aktuellen Anwendungen von ATLAS werden neben den zeitlichen Verläufen der Ergebnisdaten immer mehr ortsabhängige Darstellungen der Daten benötigt, wie sie z. B. von Containment- oder Reaktorkernmodellen berechnet werden. Generell wurden Methoden entwickelt, die statische oder dynamische Daten, wie z. B. Brandlasten, Temperaturen oder Stoffkonzentrationen, in einem anschaulichen räumlichen Zusammenhang visualisieren können. Mit realitätsnahen Darstellungen in 2D- oder 3D-Geometrie wurde erreicht, dass eine Zuordnung der Nodalisierung und der Simulationsergebnisse zur realen Umgebung erfolgen kann.

Auch bei der Darstellung der Zeitverläufe bestand noch Verbesserungsbedarf, weil im Vergleich zu modernen kommerziellen Produkten, eine Reihe von Anpassungsmöglichkeiten durch den Benutzer fehlten, wie beispielsweise Skalierungs- und Beschriftungsoptionen. Diese sind aber für eine verständliche Darstellung und Ergebnisdokumentation unverzichtbar.

Das Verfahren zur Darstellung von dynamischen Daten im Zusammenhang mit Bildern zur Ortsdiskretisierung bei ATHLET hat sich zur Analyse der Rechenergebnisse als sehr nützlich erwiesen und sollte daher auch für den Integralcode ASTEC in ähnlicher

Form verfügbar gemacht werden. Zur besseren Kontrolle der Eingabedaten und einer komfortablen Visualisierung von dynamischen Daten sollte deshalb ein Werkzeug entwickelt werden, das automatisch dynamisierte Bilddarstellungen der Nodalisierung der Thermohydraulik und eine Darstellung des Brennstabverhaltens erzeugen kann.

Die wachsende Komplexität und Detaillierung in der Simulation führt zu einem enormen Datenaufkommen, das sowohl zur Laufzeit der Simulationsprogramme als auch bei der Archivierung angemessen behandelt werden muss. Hinzu kommen eine Vielzahl von Simulationen, die in Rahmen spezieller Sicherheitsanalysen (z. B. Level-2 PSA) oder für Code-Validierung durchgeführt werden und auch Daten aus Messungen in Versuchsanlagen oder aus aufgetreten Ereignissen in Reaktoranlagen. Die Daten sind mit vielfältigen Zusatzinformationen (Eingabedaten, Programmversion, Randbedingungen etc.) verknüpft, die bisher aber in geringem Maß rechnergestützt verfügbar sind.

Für ein zeitgemäßes Datenmanagement war eine Reorganisation der Prozesskommunikation und des Datenaustauschs in ATLAS und ein Redesign der Datenhaltung im ATLAS Datenserver notwendig. Diese konnte nur durch eine Neuentwicklung des Datenservers realisiert werden, da die vorhandene Lösung wegen ihrer grundlegenden Konzeption und der der historisch gewachsenen Programmierung nicht mehr auf aktuelle Anforderungen erweiterbar war. Diese bestehen in der Handhabung großer Simulationsdatenmengen (mehrere Hunderttausend pro Simulationszeitschritt), der Erweiterung bei den möglichen Datenformaten, dem gleichzeitigen Zugriff auf mehrere Datenquellen.

Ein weiteres Ziel war eine verbesserte Archivierung vorhandener Simulationsdaten, die einen langfristigen Zugriff auf die Ergebnisdaten sicherstellt, Zusatzinformationen verwaltet und Such- und Zugriffsfunktionen bereitstellt. Die Metainformationen zu den Simulationsrechnungen und Verweise auf zugehörige Unterlagen und Dateien sollten in einer Datenbank gespeichert werden, damit diese besser verwaltet werden können und einfach wieder auffindbar sind. Die Speicherung der sehr hohen Datenmengen, resultierend aus großen Problemzeiten, der Berechnung vieler Varianten und sehr detaillierter Modellierung, sollte ebenfalls anforderungsgerecht gelöst werden. Dies konnte durch den Einsatz eines Speichersystems in Verbindung mit spezialisierter Hard- und Software realisiert werden. Zum Zugriff auf die Daten wird ein globaler Namensraum für Verzeichnisse genutzt, der unabhängig vom physikalischen Speicherort ist. Tatsächlich sind die Daten auf Festplatten und auf Magnetbandsystemen verteilt.

Zuletzt wurde eine Reihe von Qualitätssicherungsmaßnahmen in diesem Vorhaben geplant und durchgeführt. Die Programmänderungen sollten in einem zeitgemäßen Versionsmanagementsystem dokumentiert werden und dort zurückverfolgt werden können. Die erkannten oder gemeldeten Programmfehler werden in einer Fehlerverwaltungssoftware („Bugtrackingsystem“) erfasst und die Bearbeitung und die durchgeführten Maßnahmen werden dort dokumentiert. Für die Nutzer von ATLAS sollte ein WEB-basiertes Userforum geschaffen werden, um den Austausch von aktuellen Informationen besser unterstützen zu können. Die Benutzerhandbücher sollten den aktuellen Entwicklungsstand berücksichtigen und in interaktiver Version und als Printmedium in konsistenter Fassung vorliegen.

3 Weiterentwicklung von ATLAS

3.1 Erweiterung des Visualisierungsmoduls

3.1.1 Darstellung von dynamischen Daten in virtueller 3D-Umgebung

Die Ursprünge von Anwendungen der virtuellen Realität liegen überwiegend im Bereich der Fahrzeugsimulation bzw. Flugsimulation und der Unterhaltung. Heute ist der Einsatzbereich jedoch wesentlich breiter gefächert. Nach Brooks [Br99] sind die allgemeinen Vorteile eines *Virtual Reality*-Systems eine Steigerung der Produktivität und somit eine Reduzierung der Kosten. Zusätzlich hilft das System aber auch, die Kommunikation im Team zu verbessern. Für den Bereich der numerischen Simulation bietet eine VR-Umgebung darüber hinaus noch weitere Vorteile. Hier sind insbesondere die Zuordnung des Diskretisierungsgitters im Rechenprogramm zur realen Umgebung und die Möglichkeit zur Visualisierung großer Datenmengen von Interesse. Die Visualisierung stellt in diesem Fall eine Interpretationshilfe dar, die es ermöglicht, in einer virtuellen Umgebung mit den Daten zu arbeiten und zu experimentieren. Die echte Tiefenwirkung und die Möglichkeit, sich in der virtuellen Welt zu bewegen, stellen dabei eine intuitive und natürliche Form der Betrachtung dar, so dass der Anwender sich auf die wesentlichen Aspekte seiner Arbeit konzentrieren kann /LIN 05/.

3.1.1.1 Auswahl und Evaluierung von Softwarewerkzeugen

Das in Halden entwickelte VR-Werkzeug „Virtual Planner“ wurde auf Eignung für die geplante Anwendung eingehender untersucht. Es basiert auf den offenen Softwarestandards VRML, X3D und Java3D. Auf Basis dieser Techniken sind verschiedene Anwendungen entwickelt worden wie die Verfolgung der Dosisrate bei Wartungsarbeiten im Kernkraftwerk oder das Training zur Bedienung einer Brennelementwechselmaschine. Die Einspielung externer Daten ist möglich, allerdings bestehen Einschränkungen hinsichtlich der Geschwindigkeit bei großen Datenmengen oder Datenraten, da die Daten im ASCII-Dateiformat (XML) vorliegen müssen. Die Erstellung der statischen 3D-Objekte erfolgt in Halden mit dem kommerziellen Werkzeug „CREATE“. Prinzipiell ist aber jedes 3D-Tool geeignet, das die Daten als VRML exportieren kann. Immersive Techniken, wie Head Mounted Displays (HMD), Data Gloves, Stereo etc. werden unterstützt. Die Nutzung dieses VR-Werkzeugs ist prinzipiell mög-

lich. Die notwendige Anpassung der Software benötigt die Unterstützung durch die Entwickler in Halden, da eine schriftliche Dokumentation der Programme und Schnittstellen nur in geringem Umfang verfügbar ist. Offen bleibt die Erzeugung einer realistischen virtuellen Umgebung, wenn keine CAD-Pläne vorhanden sind. Fotogrammetrische Verfahren wurden in Halden bisher nicht eingesetzt.

Auf der anderen Seite existiert eine Vielzahl kommerzieller Werkzeuge zur Erstellung von VR-Anwendungen. Man unterscheidet die Erstellung der virtuellen Objekte (Avatare, Gegenstände, Räume) mit Werkzeugen wie 3D-StudioMax, Maya oder Google SketchUp und die das Zusammenfügen dieser Objekte zu einer virtuellen Welt in sogenannten Szenegrafen mit Software wie Vizard oder Virtools, wobei Überschneidungen der Funktionen existieren. Für diese Werkzeuge sind ein großer Kreis von Anwendern und entsprechende Erfahrungen vorhanden. Programmschnittstellen existieren meist in Form von Skriptsprachen (z. B. Python), sodass eine Integration externer Daten möglich ist. Die Kosten der Software liegen relativ hoch, wobei allerdings teilweise eine freie Weitergabe der Runtime-Applikationen möglich ist. Eine Aussage zur praktischen Anwendung hinsichtlich Aufwand, Eignung und Einschränkungen erfordert genauere Tests an Beispielen.

Das kommerzielle VR-Werkzeug „Vizard“ von WorldViz (WOR 10) wurde einer intensiven Evaluierungsphase unterzogen. Die Nutzung dieser Software benötigt umfangreiche Kenntnisse zu den Anwendungs- und Realisierungsmöglichkeiten (z. B. Bewegung im 3D-Raum, Blickwinkleinstellungen, Lichtquellen, Texturen etc.). Deshalb war eine intensive Einarbeitung mit Hilfe von mitgelieferten Beispielapplikationen notwendig. Im Anschluss daran wurde mit dieser Software eine Vielzahl von Einzeltests durchgeführt, in denen die besonderen Anforderungen für die geplante Applikation in ATLAS näher untersucht wurden. Ebenso wie der VR-Software von Halden waren Programmierschnittstellen, der benötigte Aufwand zur Erstellung der 3D-Objekte, die Darstellungsgeschwindigkeit, die Hardwareanforderungen und die Kosten von Interesse.

Vizard nutzt als Programmschnittstelle „Python“, eine interpretierte Programmiersprache, in der effizient objektorientiert programmiert werden kann. Für Python existieren Interfaces zu C und FORTRAN, so dass sich insbesondere die Zugriffsroutinen auf das neu entwickelte Datenformat von ATLAS nutzen lassen und daher der Zugriff auf die Simulationsergebnisse und eine Steuerung prinzipiell möglich sind.

Vizard enthält keine Werkzeuge zur Erstellung grundlegender geometrischer 3D-Objekte (wie z. B. Raumgeometrien). Diese können aber aus anderen Anwendungen wie CAD-Programmen oder Fotometriewerkzeugen oder generellen 3D-Entwurfswerkzeugen (z. B. 3D-MAX, Blender) importiert werden. Diese Grundobjekte können dann in Vizard dynamisch verändert werden, beispielsweise durch andere Texturen, Lichtquellen oder Transformationen. Ebenso stehen eine Reihe von vordefinierten Objekten (z. B. Avatare) zur Verfügung, die in den 3D-Szenen genutzt werden können. Ein Ausschnitt des Reaktorcontainments des Halden-Versuchsreaktors lag im VRML Format und konnte problemlos in Vizard importiert und visualisiert werden.

Die Test wurden auf einem leistungsfähigen PC (Mehrkernprozessor, spezielle Grafikkarte) unter WINDOWS-XP durchgeführt und ergaben keine Darstellungsprobleme auch bei komplexem Rendering und schneller Bewegung durch die 3D-Szene.

Besonders positiv ist zu bemerken, dass aus einer entwickelten Visualisierungslösung ein unabhängiges Programm (Executable) erzeugt werden kann, das ohne Lizenzkosten und ohne Installationsaufwand beliebig verteilt werden kann. Zur Änderung von Visualisierungsoptionen (z. B. Benutzermenüs, dynamische Effekte) ist aber eine kostenpflichtige Vizard-Entwicklerlizenz notwendig.

Nach dem positiven Abschluss der Evaluierung wurde beschlossen, das Produkt als Einzelplatzlizenz zu erwerben und für die Arbeiten im Vorhaben einzusetzen. Ausschlaggebend war letztlich der von WorldViz angebotene professionelle Support, die umfangreiche Dokumentation, der große Anwenderkreis und die gute Funktionalität mit programmatischer Erweiterungsmöglichkeiten.

Aus der Datenlage von computerbasierten Geometriedaten ergab sich, dass die notwendigen 3D-Darstellungen eines Reaktorgebäudes und den Einbauten vollständig im Projekt generiert werden mussten, da CAD-Daten von Hersteller oder Betreiber nicht zugänglich waren. Aus diesem Grund wurde auch Software zur Erstellung von 3D-Modellen evaluiert. In diesem Bereich wurde besonders auf eine kostengünstige Lizenz und eine einfache Anwendbarkeit geachtet, da der Aufwand für diese Modellierung nicht in der ursprünglichen Planung des Vorhabens berücksichtigt war.

Zur Generierung der statischen 3D-Modelle wurden zunächst verschiedene fotometrische Softwarewerkzeuge auf Eignung untersucht. Alle Werkzeuge dieser Art benötigen eine Vielzahl von geeigneten Fotos der Originalgeometrien (z. B. Komponenten) aus

unterschiedlichen Blickwinkeln. Als Beispiele wurden Leitungen und Kessel eines Anlagenraums gewählt. Die untersuchten fotometrischen Verfahren, die Shareware Tools „Photo To 3D“ und „Canoma“ liefern bei komplexen Objekten und semiautomatischer Verarbeitung der Bilder keine zufriedenstellenden Ergebnisse. Das gilt besonders dann, wenn Objekte nicht komplett fotografiert werden können, z. B. bei teilweiser Verdeckung oder wegen ihrer Größe im Raum. Kommerzielle Werkzeuge, wie „PhotoModeler“ von Eos Systems /EOS 10/ konnten aus Zeit- und Kostengründen nicht näher untersucht werden. Es ist aber festzuhalten, dass die Darstellung der Einbauten zwar den Realitätseindruck verbessert, aber für die 3D-Dynamikdaten-Visualisierung nicht unbedingt benötigt wird.

Des Weiteren ist es auch möglich, spezielle 3D-Laser-Scanner einzusetzen, eine spezialisierte und teure Hardware, die eine Vielzahl von Punktkoordinaten der realen Objekte erfassen können. Allerdings ist es heute noch nicht möglich, diese Punktwolken automatisch in ein 3D-CAD-Modell zu übertragen. Dennoch erleichtern diese Daten den Prozess zur Erstellung eines parametrischen Modells erheblich. Insgesamt ergibt sich als Erkenntnis, dass die Modellierung realitätsnaher 3D-Welten mit erheblichem manuellen Zeichenaufwand verbunden ist.

Als Werkzeug zur manuellen Erstellung von 3D-Modellen für Gebäude wurde die Software "Google SketchUp" eingesetzt, mit der sehr effizient aus einfachen 2D-Grundformen 3D-Modelle generiert werden können. SketchUp ist eine Designsoftware von Google, die das Erstellen von komplexen 3D-Objekten ermöglicht. Im Gegensatz zu anderen CAD-Programmen ist SketchUp auch für Einsteiger leicht zu bedienen, was den Ausschlag zur Verwendung dieser Software gegeben hat. Besonders wichtig für die weitere Arbeit war, dass die Software diverse CAD-Formate importieren und exportieren kann, also die erstellten 3D-Modelle an anderer Stelle, z. B. in anderen CAD-Programmen, Vizard etc. weiterverwendet werden können.

3.1.1.2 Erstellung eines 3D-Modells für das Reaktorgebäude eines SWR-69

Es wurde geprüft, ob in der GRS CAD-Daten für ein Reaktorgebäude bzw. einzelner Räume und eingebauter Komponenten verfügbar sind. Originale Herstellerdaten sind in der GRS nicht oder nur als 2D-Grundrisse vorhanden. Lediglich Baupläne auf Papierbasis als Grundrisse und Schnitte sind für einige deutsche Anlagen verfügbar. CAD-Daten für Komponenten, wie Reaktordruckgefäß, Pumpen usw. sind ebenfalls nicht

vorhanden Die Papierdokumentation ist häufig nur schematisch, ohne Bemaßungen und andere maschinenbautechnische Details, vorhanden und wird teilweise durch Fotos unterschiedlicher Qualität ergänzt.

Als Testbeispiel wurde das Reaktorgebäude eines SWR der Baulinie 69 ausgewählt, da dafür zum Teil 2D-CAD-Daten und detaillierte Pläne vorlagen, ebenso ein Simulationsdatensatz für das Containmentmodell COCOSYS.

Nach entsprechender Einarbeitung wurde ein vereinfachtes 3D-Modell des Reaktorgebäudes mit Türen und Wänden erstellt. Entsprechend vorliegender Schnitt- und Grundriss-Darstellungen in Papierplänen wurden 3D-Darstellungen für verschiedene Höhenebenen des Reaktorgebäudes erstellt. Die Darstellungen in den Abb. 3-1 und Abb. 3-2 sind aus urheberrechtlichen Gründen sehr stark abstrahiert gezeigt. Wichtige Großkomponenten wie Behälter werden teilweise in den einzelnen Betriebsräumen als vereinfachte 3D-Körper angezeigt. Aufgrund der sicherheitstechnischen Bedeutung ist das Reaktorcontainment mit dem Sicherheitsbehälter und Kondensationskammer und Einbauten wie Reaktordruckgefäß mit den anschließenden Leitungen detaillierter ausgeführt. Die 3D-Modelle sind sowohl in einer Gesamtsicht des Reaktorgebäudes als auch in Einzelmodellen der verschiedenen Stockwerke und des Sicherheitsbehälters verfügbar.

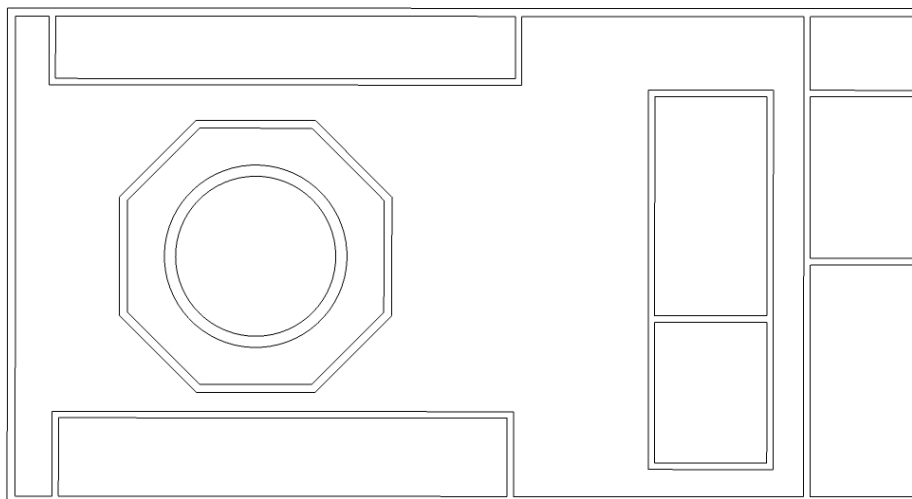


Abb. 3-1 Grundriss eines SWR-69 bei 0.0 m Höhe

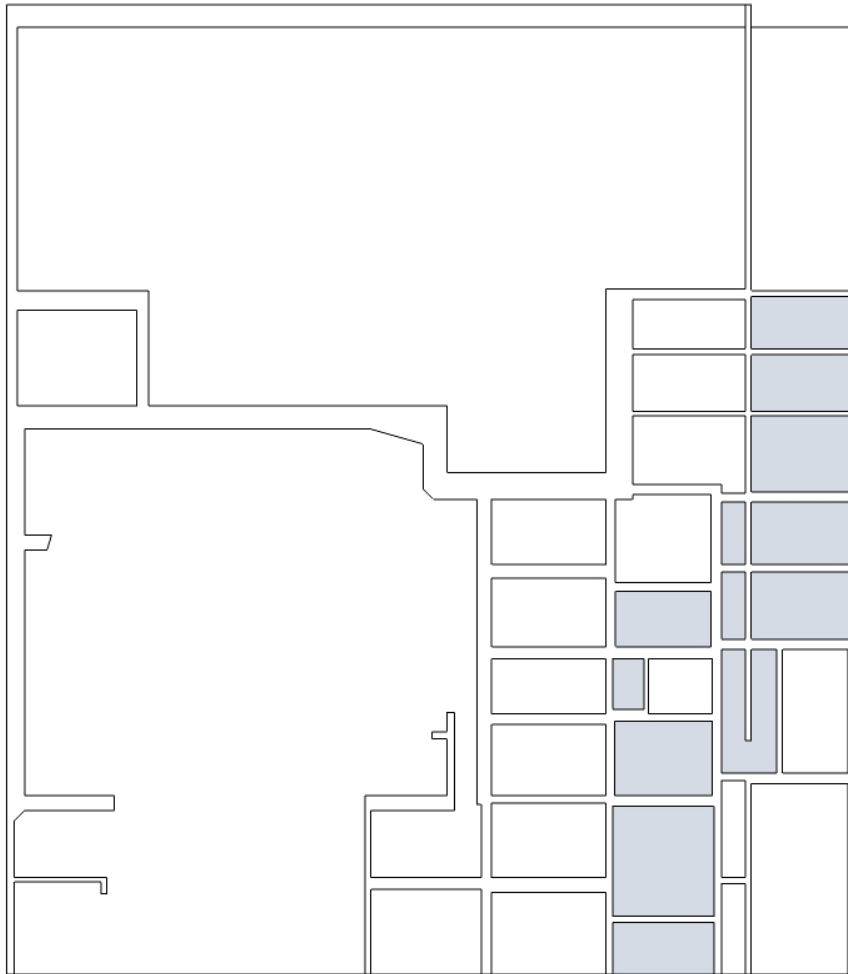


Abb. 3-2 Querschnitt durch das Reaktorgebäude von KKI-1

Zur Erzeugung des 3D-Modells in SketchUp ist folgende Vorgehensweise angewandt worden. Zuerst ist die Zeichnung der in Papierplänen verfügbaren Grundrisse und Querschnitte in separate SketchUp -Dateien erfolgt. Zu diesem Zeitpunkt lag schon die Erfahrung aus früheren Versuchen vor, dass die in den Grundrissen vorhandenen geometrischen Angaben allein zur Darstellung dreidimensionaler Abbildungen nicht ausreichen.

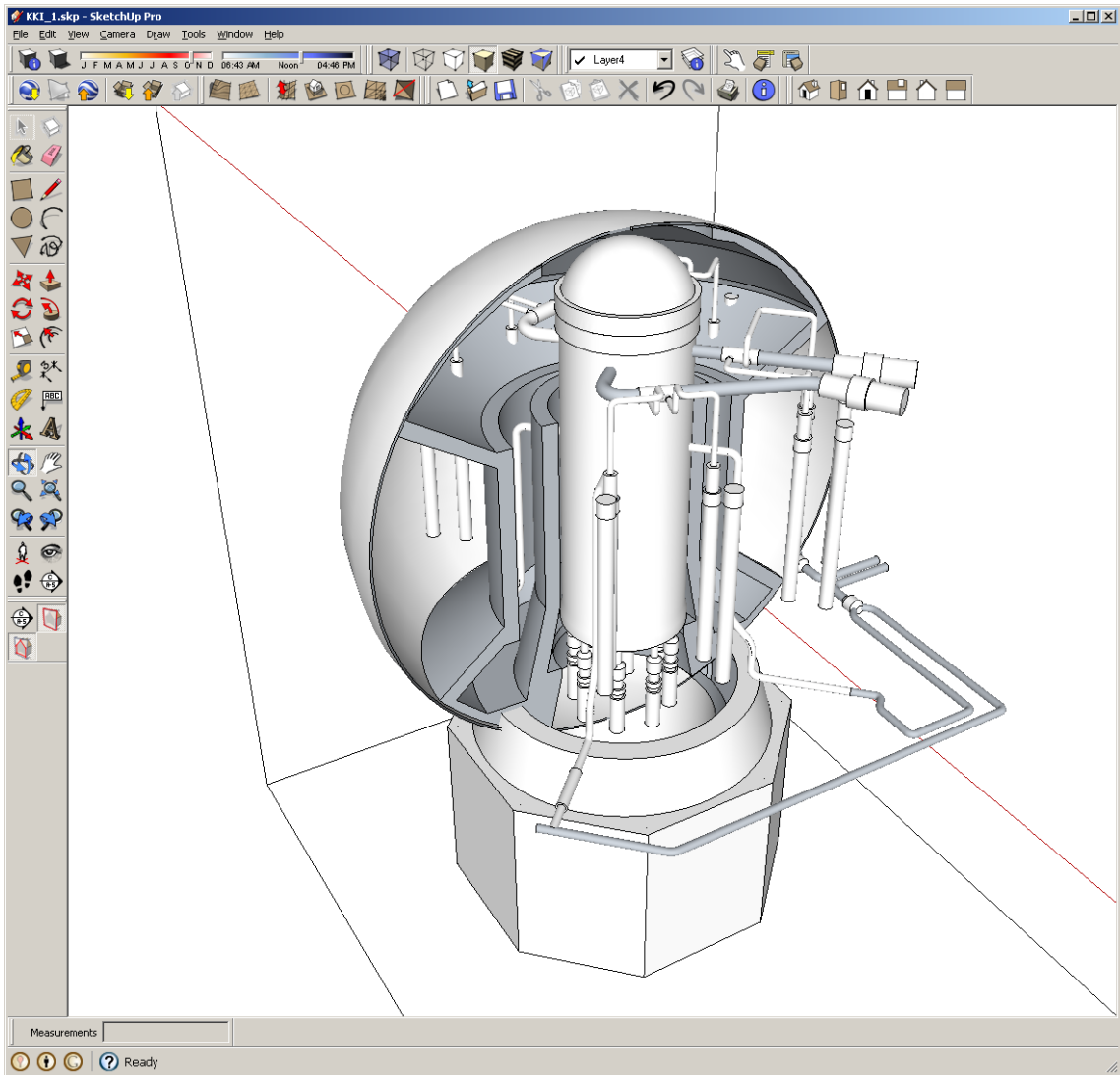


Abb. 3-3 Gestaltung des Hauptfensters von Google-SketchUp

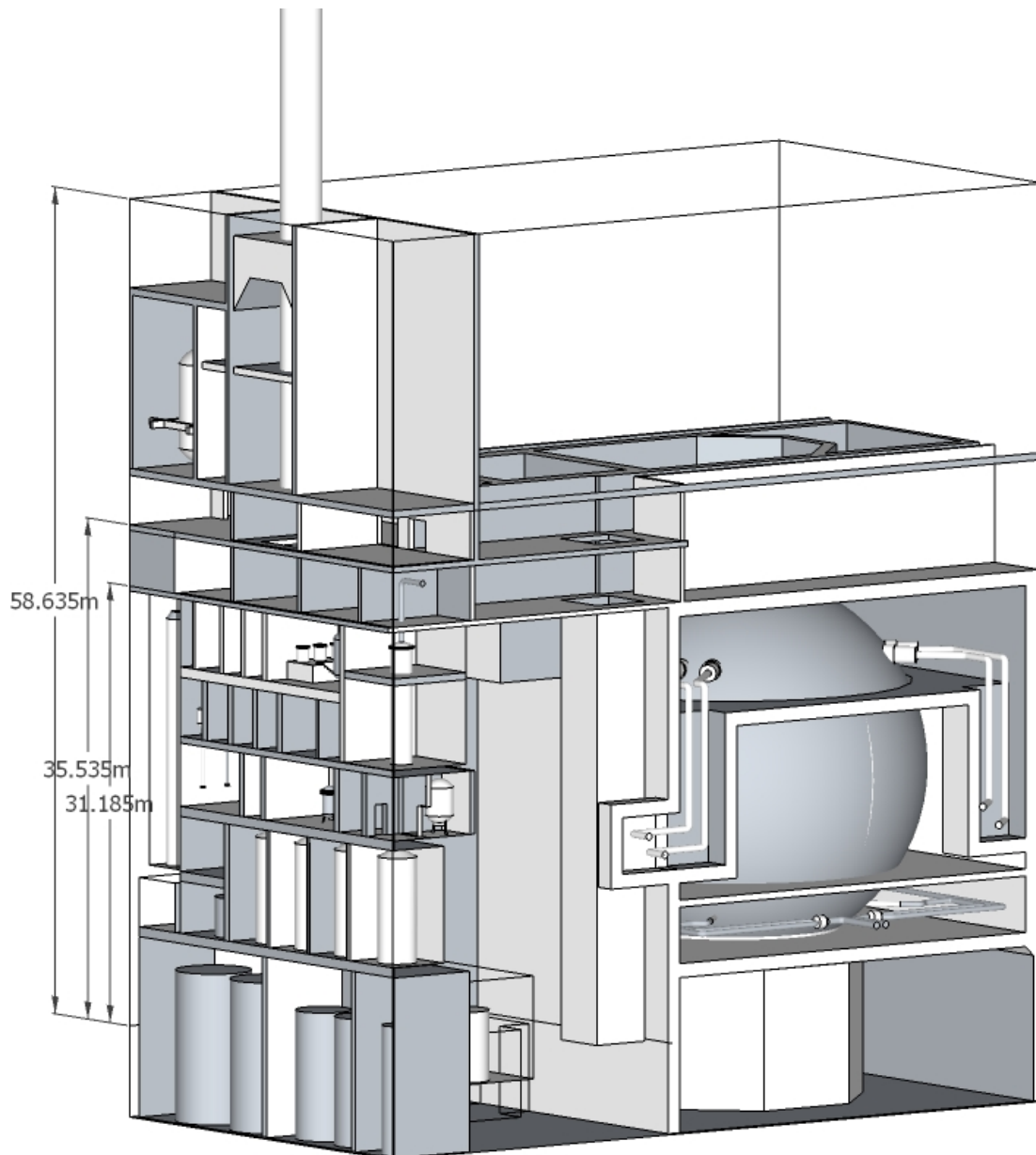


Abb. 3-4 Abbildung wichtiger Teile des Reaktorgebäudes eines SWR-69

Es wurde klar, dass zur Abbildung der Räume und Komponenten eines gewissen Stockwerkes, die geometrische Intersektion des entsprechenden Grundrisses mit allen vorhandenen Schnitten notwendig war. Nur so konnte unter anderem, z. B., die Höhe der Räume und die Gestalt und Anordnung der Komponenten ermittelt werden.

Zum Aufbau der Etagen in 3D (Abb. 3-4) ist deswegen die „Technik des Imports“ verwendet worden, d. h. in die betroffenen Grundriss-Dateien wurden die in separat gespeicherten Schnitte, importiert. Bei dieser Arbeitsweise ist eine exakte

Übereinstimmung der Positionen der Objekte im Grundriss und Schnitt erforderlich. Diese Übereinstimmung kann nur durch die genaue Übertragung der Maße und ein einheitliches Koordinatensystem erreicht werden.

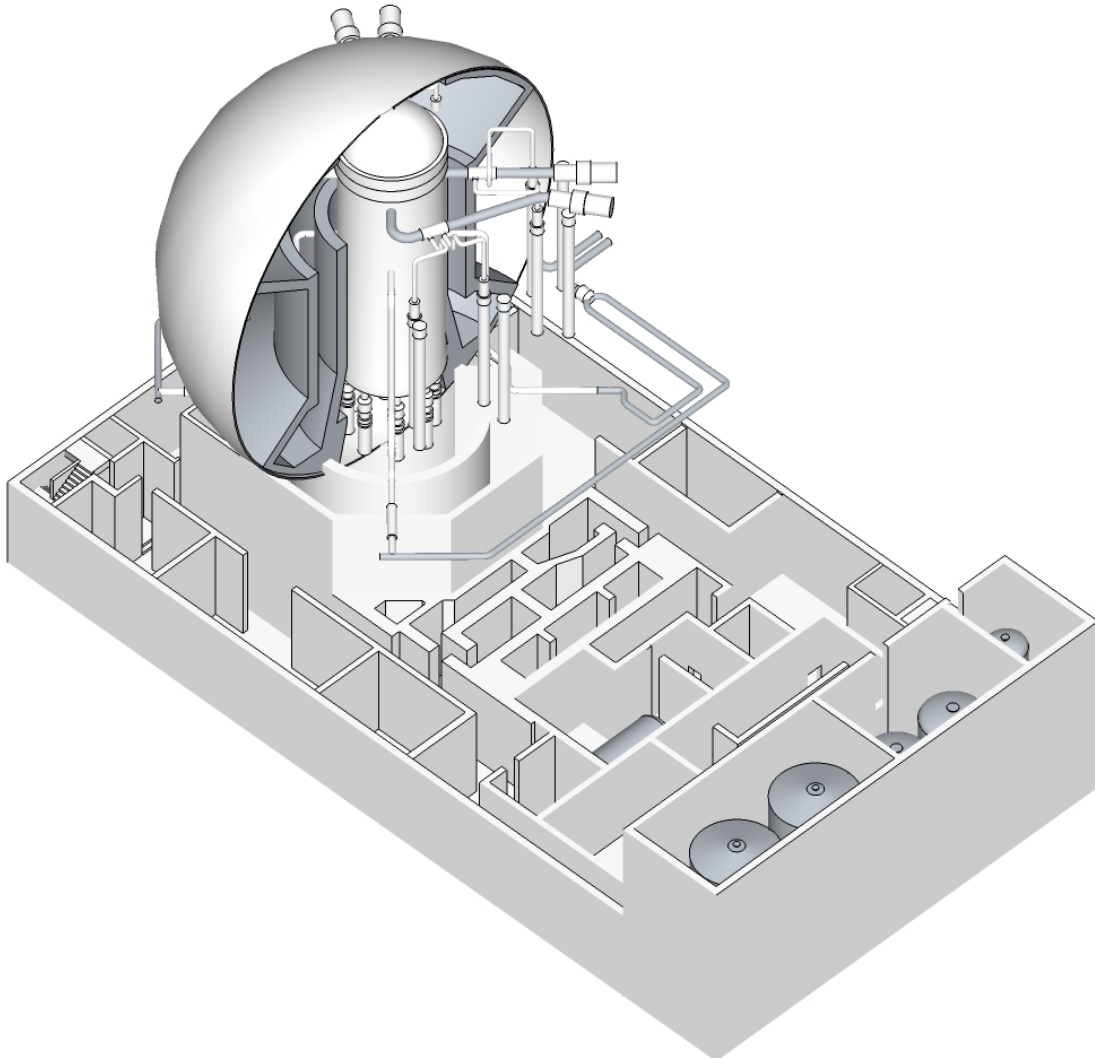


Abb. 3-5 Abbildung des unteren Stockwerkes mit zugehörigen Komponenten eines SWR-69

Es ergibt sich die Erkenntnis, dass die Erstellung der dreidimensionalen Geometrie umso leichter durchführbar ist je genauer die zweidimensionalen Pläne sind.

Für die Aufbau der Wände und der Anlagenkomponente aus den zweidimensionalen Grundrissen und Schnitten sind vor allem die Tools „Push/Pull“ und „Follow me“ vorteilhaft einsetzbar. Mit Hilfe des ersten lässt sich aus einem beliebigen Grundriss ein entsprechendes orthogonales Objekt schnell aufbauen. Mit dem zweiten kann man die komplexeren Komponenten der Anlage, unter anderem den Reaktordruckbehälter, den Sicherheitsbehälter und die Rohrleitungen, darstellen. Zum Zeichnen einer Rohrleitung

kann man beispielsweise folgendermaßen vorgehen. Zuerst muss man die Mittellinie der Rohrleitungen so genau wie möglich zu zeichnen (Abb. 3-6).

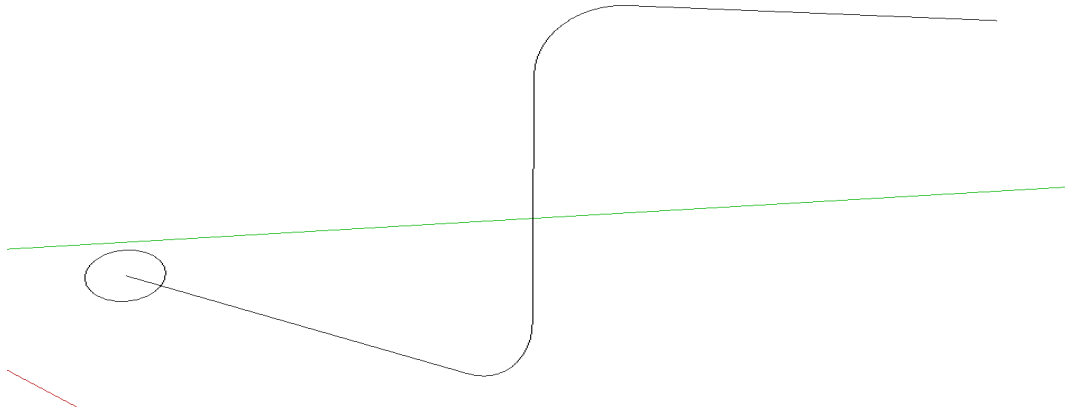


Abb. 3-6 Mittellinie eines Leitungsverlaufs in SketchUp

Diese Linien dienen dann als Pfad um mit dem Tool „Follow me“ eine Rohrleitung mit konstantem Durchmesser entlang des Pfades zu generieren (Abb. 3-7).

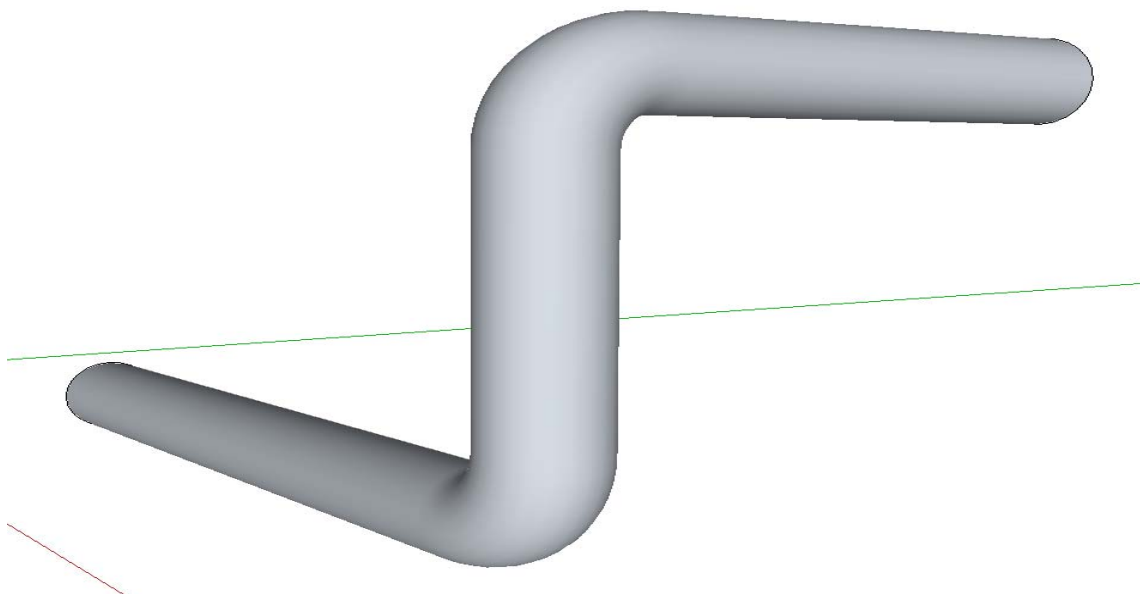
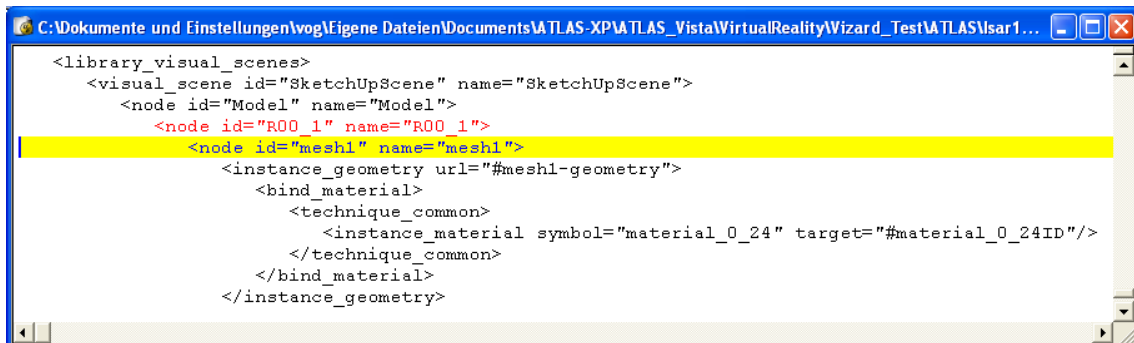


Abb. 3-7 3D-Leitung entlang einer Mittellinie in SketchUp

3.1.1.3 Visualisierung von Simulationsdaten in Virtual Reality (VR)

Für den Datenaustausch zwischen SketchUp und der VR-Software Vizard wurden verschiedene Formate untersucht. Gute Ergebnisse liefert das „COLLADA“-Format /BAR 08/, das als ein XML-basiertes offenes Austauschformat von verschiedenen Herstellern entwickelt wird. Mit diesem Format konnten die in SketchUp generierten Modelle in das VR-Werkzeugs Vizard eingespielt werden. Es ist wichtig, die einzelnen Räume, die später mit Daten aus der Containmentsimulation mit COCOSYS dynamisiert werden sollen, in SketchUp als Komponente zu speichern und mit einem Namen zu versehen. Besonders geeignet ist eine Bezeichnung wie im COCOSYS Eingabedatensatz, um die Zuordnung der Ergebnisdaten zu erleichtern. In Abb. 3-8 wird ein Auszug aus der COLLADA-Datei für die 3D-Darstellung des Reaktorgebäudes in der untersten Höhenebene gezeigt. Die rot markierte Zeile (Abb. 3-8) bezeichnet den Raum unterhalb des Sicherheitsbehälters, in diesem Fall den Zylinder „R00_1“ bzw. R00 in der Simulation, wie im entsprechenden Nodalisierungsdigramm von COCOSYS (Abb. 3-8) angegeben.



```
<library_visual_scenes>
  <visual_scene id="SketchUpScene" name="SketchUpScene">
    <node id="Model" name="Model">
      <node id="R00_1" name="R00_1">
        <node id="mesh1" name="mesh1">
          <instance_geometry url="#mesh1-geometry">
            <bind_material>
              <technique_common>
                <instance_material symbol="material_0_24" target="#material_0_24ID"/>
              </technique_common>
            </bind_material>
          </instance_geometry>
        </node>
      </node>
    </node>
  </visual_scene>
</library_visual_scenes>
```

Abb. 3-8 Auszug einer 3D-Datei aus SketchUp für Datenvisualisierung in Vizard

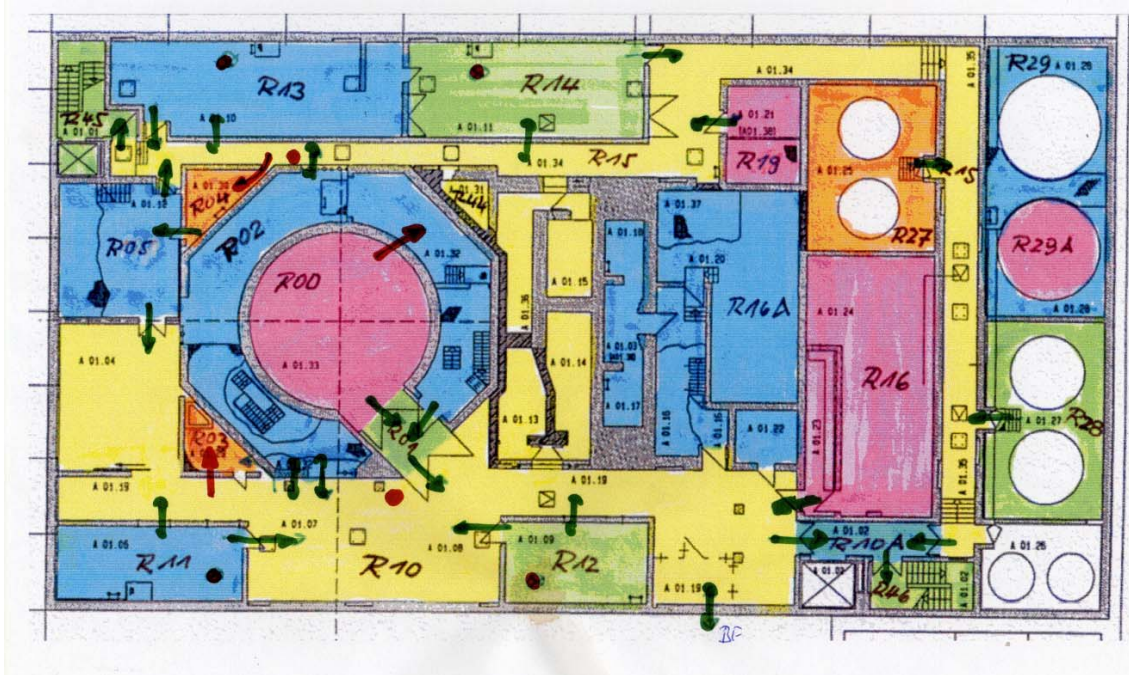


Abb. 3-9 Nodalisierungschema von COCOSYS für das Reaktorgebäude

In Vizard kann dann mit dem Befehl „viz.add(filename)“, wobei „filename“ der Name der Collada-Datei für das Reaktorgebäude ist, in die virtuelle Welt geladen werden.

In Vizard ist der Node „R00_1“ dann ein Teilelement des Reaktorgebäudes. Für dieses Element können dann in Vizard grafische Eigenschaften gesetzt werden, z. B. Farbe oder Transparenz. Diese können dann zu Laufzeit von Vizard mit einem Skript Programm in Python entsprechend der zeitlichen Änderung von physikalischen Größen in der Simulation geändert werden.

Zu diesem Zweck war eine Möglichkeit zur Einspielung von Simulationsergebnissen nach Vizard notwendig. Für den Datenzugriff in ATLAS auf Simulationsdaten wird die neue NetCDF-Dateischnittstelle (siehe Abschnitt 4.2) verwendet. Diese sollte daher auch für Vizard genutzt werden. Dazu musste eine spezielle Programmbibliothek entwickelt werden, mit der von Vizard Programmierschnittstelle "Python" auf die Daten zugegriffen wird. Diese Programmbibliothek wird für Vizard in Form einer WINDOWS-Dynamic-Link-Library (DLL) zur Verfügung gestellt und umfasst alle Funktionen, die auch in ATLAS zur Verfügung stehen, also z. B. Lesen von Daten zu einem Zeitpunkt oder Lesen einer Wertereihe in einem Zeitfenster (Trend). Das Lesen der Simulationsdaten in Vizard wird momentan über einstellbaren „Timer“ gesteuert, also beispielsweise jede Sekunde ein neuer Simulationszeitpunkt. Eine Synchronisation mit ATLAS

kann, bei Bedarf, über eine zusätzliche Steuerdatei abgewickelt werden. Die Daten werden dann ähnlich wie in ATLAS, farbig kodiert. In den Beispielbilder Abb. 3-10 und Abb. 3-11 wird die Raumtemperatur angezeigt, wobei Dunkelblau einer niedrigen Temperatur (30 Grad Celsius) und Rot einer hohen Temperatur (200 Grad Celsius) entspricht. Die Simulationsdaten wurden einer vorliegenden COCOSYS-Rechnung für eine SWR-69 mit einem RDB-Deckelleck entnommen. Auf einem leistungsfähigen Quadcore-PC mit einer Midrange-Grafikkarte, NVIDIA Quadro FX 1700 /NVI 10/, ergeben sich gute Reaktionszeiten beim Bildaufbau bei Änderung der dynamischen Daten oder des Viewports.

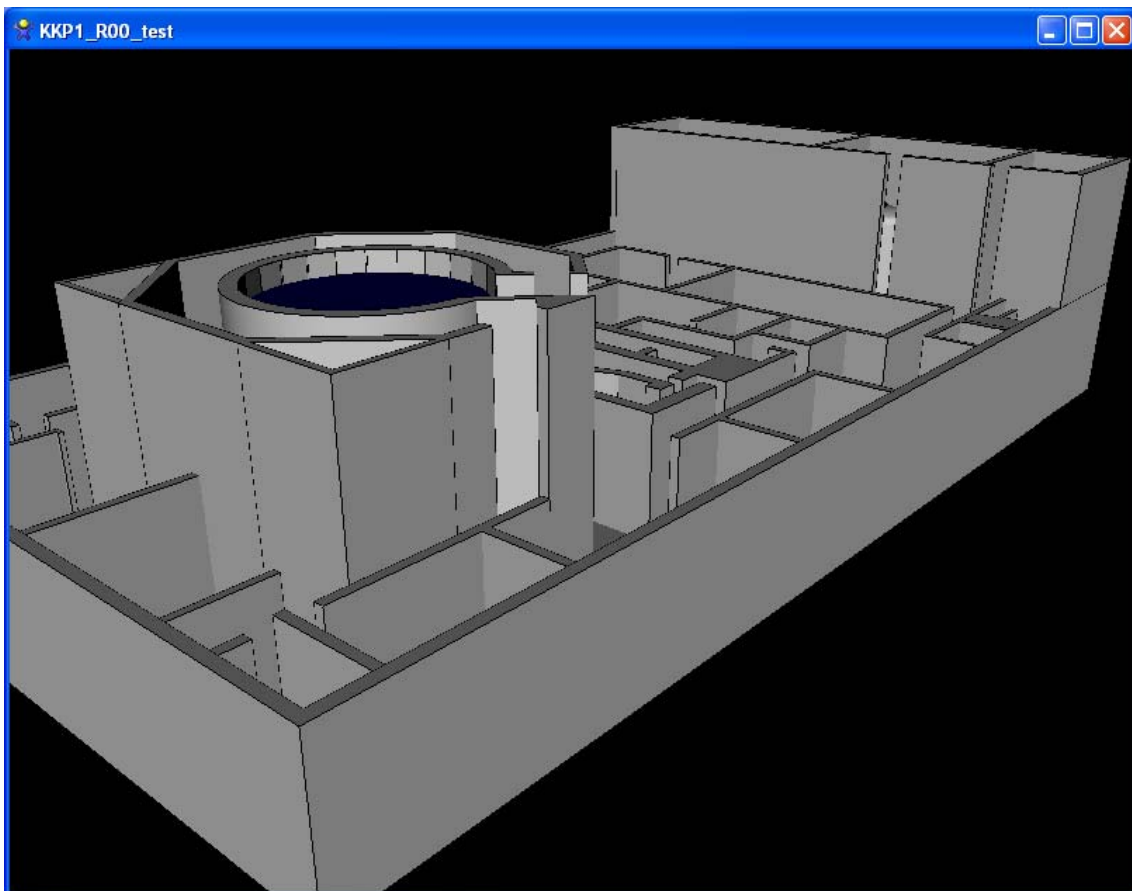


Abb. 3-10 Visualisierung einer niedrigen Raumtemperatur im 3D-Reaktorgebäude (Viewport1, Beginn der Transiente)

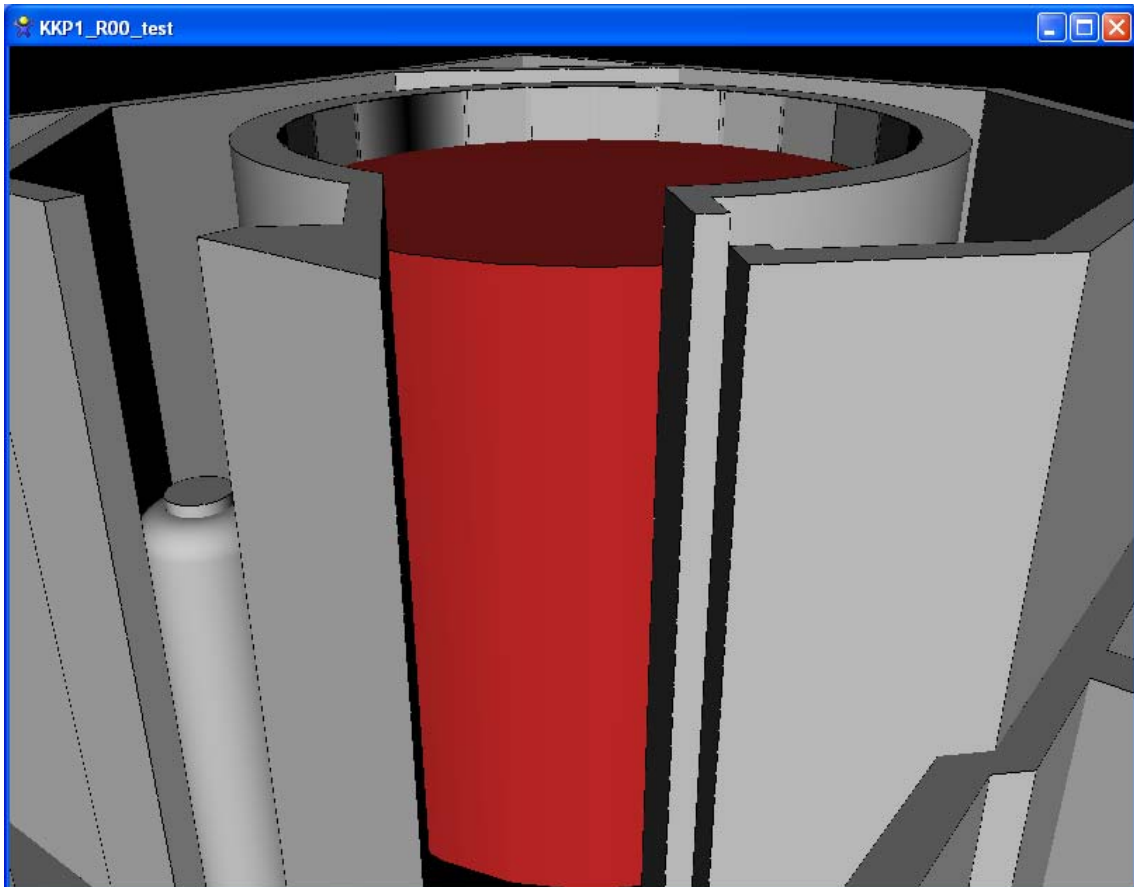


Abb. 3-11 Visualisierung einer hohen Raumtemperatur im 3D-Reaktorgebäude (Viewport2, Ende der Transiente)

Der Benutzer kann sich beliebig in der virtuellen Umgebung bewegen. Dazu wurde eine kombinierte Steuerung durch Tastatur und Maus entwickelt. Mit dieser Steuerung kann er seinen Standpunkt und seine Blickrichtung verändern (Viewport), sich also einem Objekt nähern oder davon entfernen und sich in alle Richtungen umschaun. Die Steuerung erfordert vom Benutzer etwas Übung und wäre durch Nutzung spezieller 3D-Eingabegeräte, z. B. Joystick, einfacher und intuitiver.

3.1.1.4 Zusammenfassende Bewertung des Verfahrens

Für das Reaktorgebäude deutscher SWR-69 Anlagen liegt ein weitgehend komplettes, maßstabgetreues 3D-Geometriemodell vor, das in anderen Anwendungen (z. B. CAD-Programme, Gittergeneratoren) nutzbar ist. Die Einbauten in den Räumen, abgesehen vom Sicherheitsbehälter, sind nur exemplarisch berücksichtigt und ein realitätsnahes Rendering (Texturen, Licht, etc.) sind noch nicht vorhanden. Aufwendig und schwierig

ist die Darstellung komplexer Einbauten in einem Raum, insbesondere bei sehr unregelmäßiger Geometrie.

Es konnte aber gezeigt werden, dass sich Virtual Reality Werkzeuge dazu eignen, auch dynamische Simulationsdaten in Verbindung mit der Darstellung von Räumen und Komponenten zu visualisieren. Im Vergleich den bisher verwendeten schematischen Darstellungen (z. B. Schnitte) kann der Benutzer die Daten wesentlich leichter im Raum zuordnen und erhält durch die flexiblen Ansichtsmöglichkeiten der VR-Software in auch eine bessere Gesamtsicht auf die lokalen Daten in den verschiedenen Räumen.

3.1.2 Erweiterte Darstellungsoptionen für Zeitkurven und bei der Kerndatenvisualisierung

Die Darstellung der Zeitkurven und der ortsabhängigen Daten im Reaktorkern wird mit einer kommerziellen Softwarekomponente (ComponentOne Chart Active-X Control) ausgeführt. Diese Software verfügt über eine Vielfalt von Auswahlmöglichkeiten in der Darstellung, wie Achsenskalierung, Legenden, Farbskalen usw., die über die Bedienoberfläche nicht oder nur ungenügend angesprochen werden konnten. Mit einer verbesserten Auswahlmöglichkeit einiger dieser Optionen konnte die Darstellungsqualität und die Verständlichkeit verbessert werden. Auch eine Verwendbarkeit der Diagramme in der Dokumentation von Analysen ist nun weitere Nachbearbeitung möglich.

3.1.2.1 Darstellungsoptionen für Zeitkurven

Zur Modifizierung von Diagrammen konnte bisher nur der „Property Editor“ von „**ComponentOne Chart**“ /COM 09/ zur Verfügung gestellt werden, dessen Benutzeroberfläche wenig übersichtlich und daher nicht sehr leicht zu bedienen ist. Außerdem bietet er lediglich die Möglichkeit, die Darstellung einzelner Diagramme anzupassen und nicht ganze Gruppen gemeinsam zu ändern. Aus diesem Grund wurde ein eigener, für ATLAS angepasster, Trend Property Editor entwickelt. Die programmtechnische Umsetzung wird detailliert in einem eigenen Bericht /DEI 08/ erläutert, der auch als Referenz für eventuelle Weiterentwicklungen genutzt werden kann.

Für die Vorgabe wichtiger und häufig modifizierter Eigenschaften der Darstellung der Zeitkurven und der Koordinatensysteme, wie beispielsweise Linientyp, Achsenmaß-

stab, Legende, enthält dieser eine Benutzeroberfläche mit optimierten Bediendialogen. In diesen Dialogen können auf einfache Weise das Erscheinungsbild einzelner Kurven, von Kurvengruppen, von Trendsets oder global von allen Diagramme verändert werden. Die Änderungen können bei Bedarf wieder zurückgenommen werden.

Die Eigenschaften der einzelnen Elemente eines Trendkurvendiagramms, wie Beschriftungen, Kurvenfarben, Achsen, werden mit speziellen Bezeichnungen in einer Datei gespeichert (siehe Abs. 5.1.1) und können so beim Neustart von ATLAS wieder hergestellt werden. Auch die Speicherung verschiedener Darstellungsvarianten ist möglich. Ebenfalls ist es möglich, bei Bedarf direkt in der Speicherdatei Änderungen mit einem Standardeditor vorzunehmen. Mit diesem Konzept sind bedarfsbezogene Erweiterungen der Einstelloptionen leicht realisierbar.

Der Aufwand zur Anpassung und Optimierung bei der Darstellung von Zeitdiagrammen wurde spürbar verringert. Es ist nun möglich optimierte Darstellungsformen, z. B. für schriftliche Berichte, vorzuhalten und ohne Zusatzaufwand jederzeit zu erzeugen. Die Qualität der erzeugbaren Diagramme entspricht den anderen Post-processingwerkzeugen, die gegenwärtig für ATHLET eingesetzt werden wie beispielsweise JSPLLOT /SIE 09/. Mit der in ATLAS vorhandenen Option zum Import von dort verwendeten Definitionsdateien zur Beschreibung der Diagramme, ist eine einfache Migration möglich.

Im Folgenden wird die Nutzung und Bedienung des Trend Property Editors beschrieben. Zwei Möglichkeiten gibt es, diesen aufzurufen:

- Mit einem Klick der rechten Maus-Taste auf den Diagramm-Hintergrund öffnet sich ein Popup-Menü, aus dem der Property Editor ausgewählt werden kann.
- Nach der Auswahl von „Options“ aus der Haupt-Menü-Leiste von ATLAS öffnet sich ein Popup-Menü, aus dem der Property Editor ausgewählt werden kann.

Nach dem Start des Property Editors wird ein Benutzerdialog (Abb. 3-12) angezeigt, der im linken Teil in einer Baumstruktur die Elemente aller Diagramme enthält, die vom Anwender modifiziert werden können:

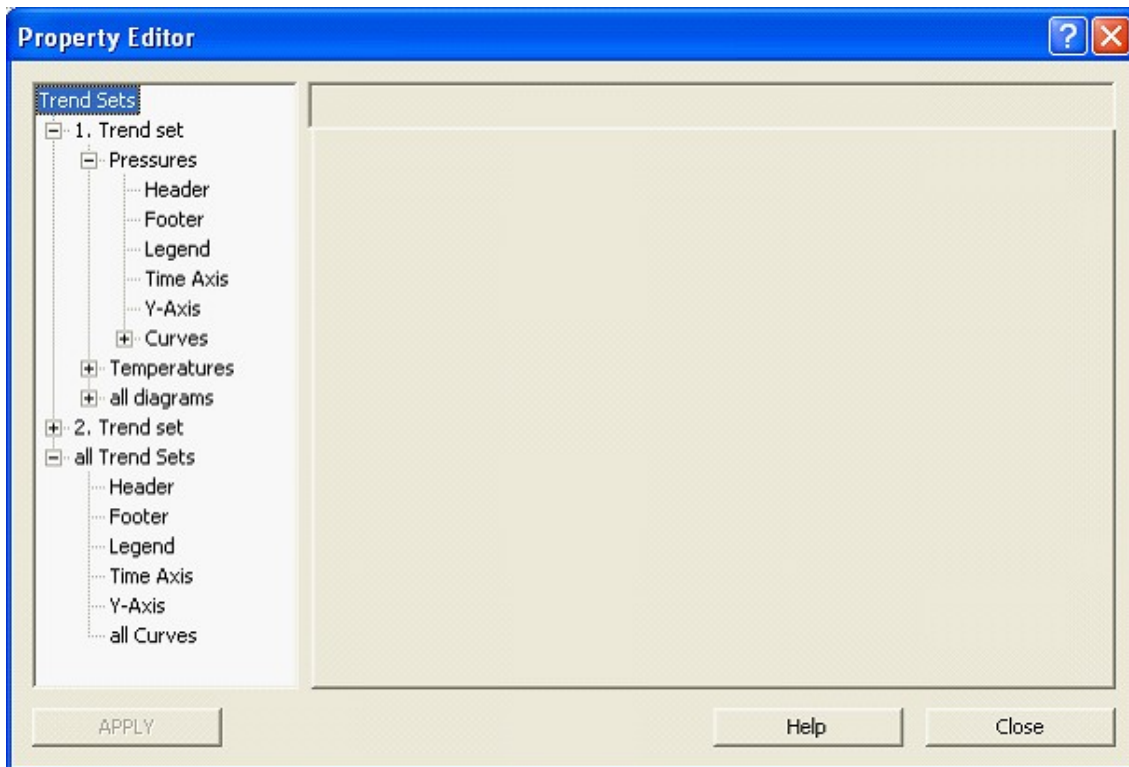


Abb. 3-12 Auswahl-Baum für Elemente aller Diagramme im Trend Property Editor

Durch einen Mausklick auf eine Zeile des Baums wird ein Element zur Modifizierung ausgewählt. Unter „**all Trend Sets**“, „**all diagrams**“ und „**all Curves**“ können einige Parameter aller Trendsets bzw. aller Diagramme oder aller Kurven gemeinsam gleichzeitig modifiziert werden, z. B.:

- Sollen Parameter der Y-Achse des Diagramms “Pressures” im 1. Trendsets modifiziert werden, dann ist zunächst “1. Trend Set” auszuwählen, danach “Pressures” und zum Schluss “Y-Axis”.
- Sollen Parameter der Zeit-Achse für alle Trend sets modifiziert werden, dann ist zunächst “all Trend Sets” und danach “Time Axis” auszuwählen.

Nach einem Mausklick auf "**APPLY**" werden alle Änderungen an einem ausgewählten Element übernommen. Ein Mausklick auf "**Close**" beendet den Property Editor ohne Speicherung der letzten Modifikationen.

3.1.2.1.1 Header

Mit dem Dialog für „Header“ (Abb. 3-13) ist es möglich, verschiedene Optionen für die Diagrammüberschriften einzustellen.

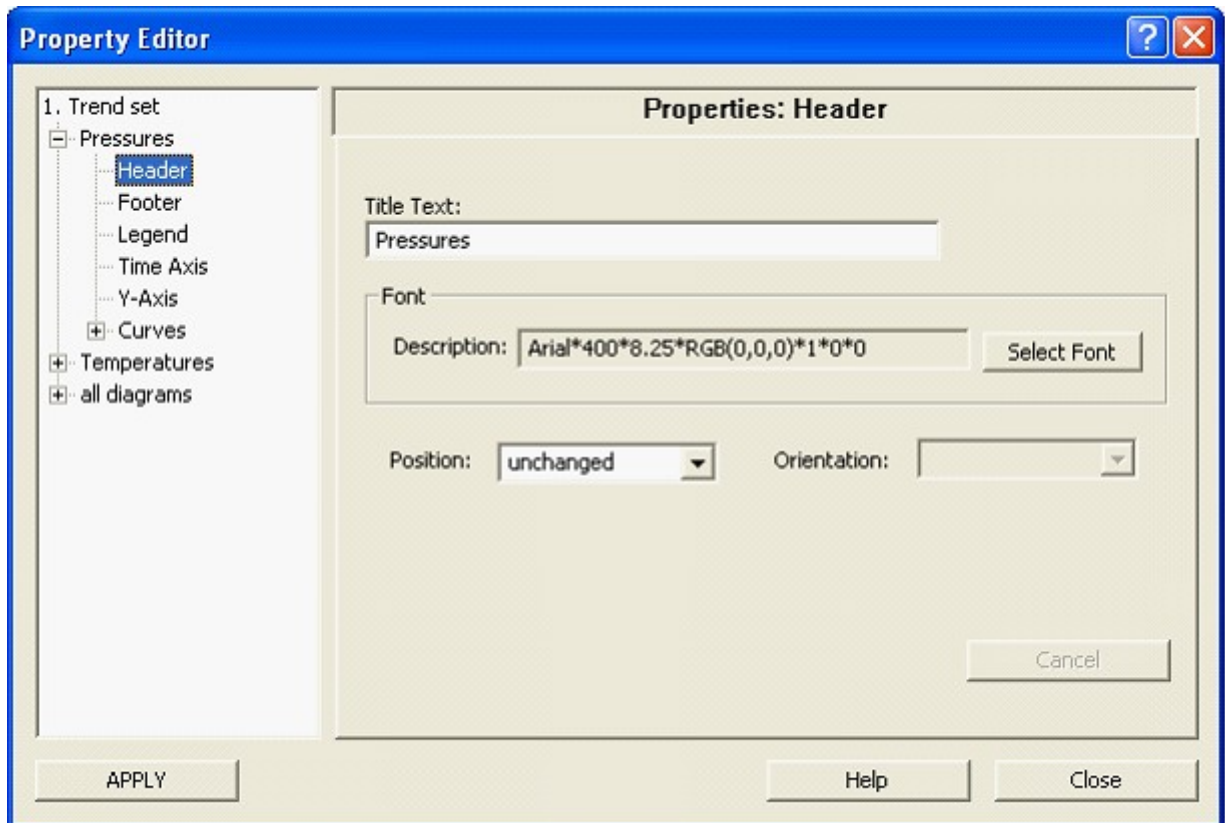


Abb. 3-13 Header-Dialog im Trend Property Editor

Title Text:

Es kann ein beliebiger Text mit maximal 255 Zeichen eingegeben werden, der als Kopfzeile über dem Diagramm ausgegeben wird.

Select Font:

Nach einem Mausklick auf "**Select Font**" öffnet sich das Menü „Schriftart“ (Abb. 3-14) in dem die Font-Parameter für die Kopfzeile ausgewählt werden können.

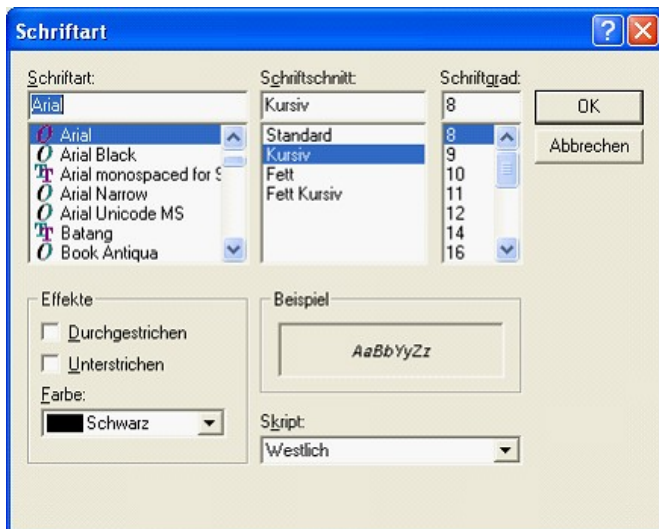


Abb. 3-14 Dialog zur Bestimmung der Schriftart im Trend Property Editor

Position:

Abhängig von der Eingabe "left" oder "center" wird die Kopfzeile entweder am linken Rand des Diagramms oder über der Diagrammitte positioniert.

3.1.2.1.2 Footer

Mit dem Dialog für „Footer“ (Abb. 3-15) ist es möglich, verschiedene Optionen für eine Diagrammunterschrift (Fußzeile) einzustellen.

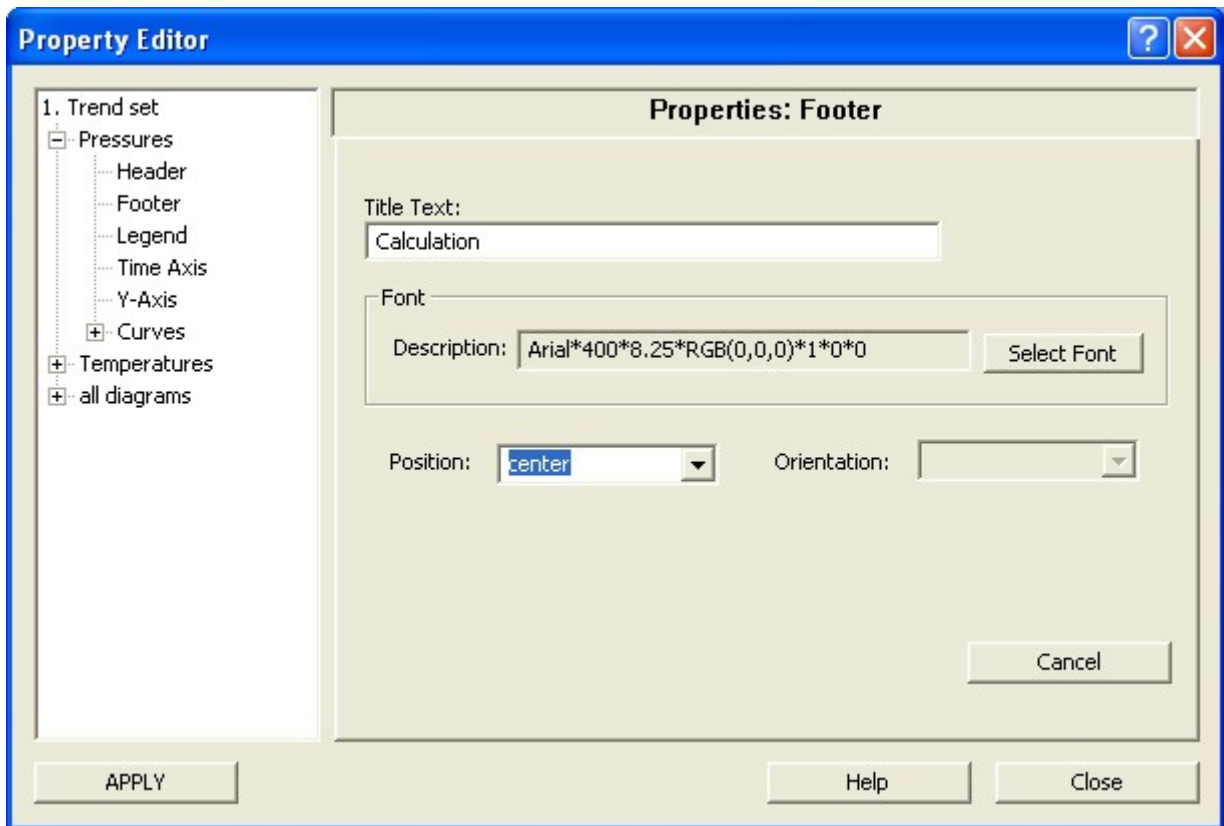


Abb. 3-15 Footer-Dialog im Trend Property Editor

Title Text:

Es kann ein beliebiger Text mit maximal 255 Zeichen eingegeben werden, der als Fußzeile unter dem Diagramm ausgegeben wird.

Select Font:

Nach einem Mausklick auf **“Select Font“** öffnet sich das Menü „Schriftart“ in dem die Font-Parameter für die Fußzeile ausgewählt werden können.

Position:

Abhängig von der Eingabe “left“ oder “center“ wird die Fußzeile entweder am linken Rand des Diagramms oder unter der Diagrammitte positioniert.

3.1.2.1.3 Legend

Mit dem Dialog für „Legend“ (Abb. 3-16) ist es möglich, verschiedene Optionen für eine Diagrammlegende einzustellen.

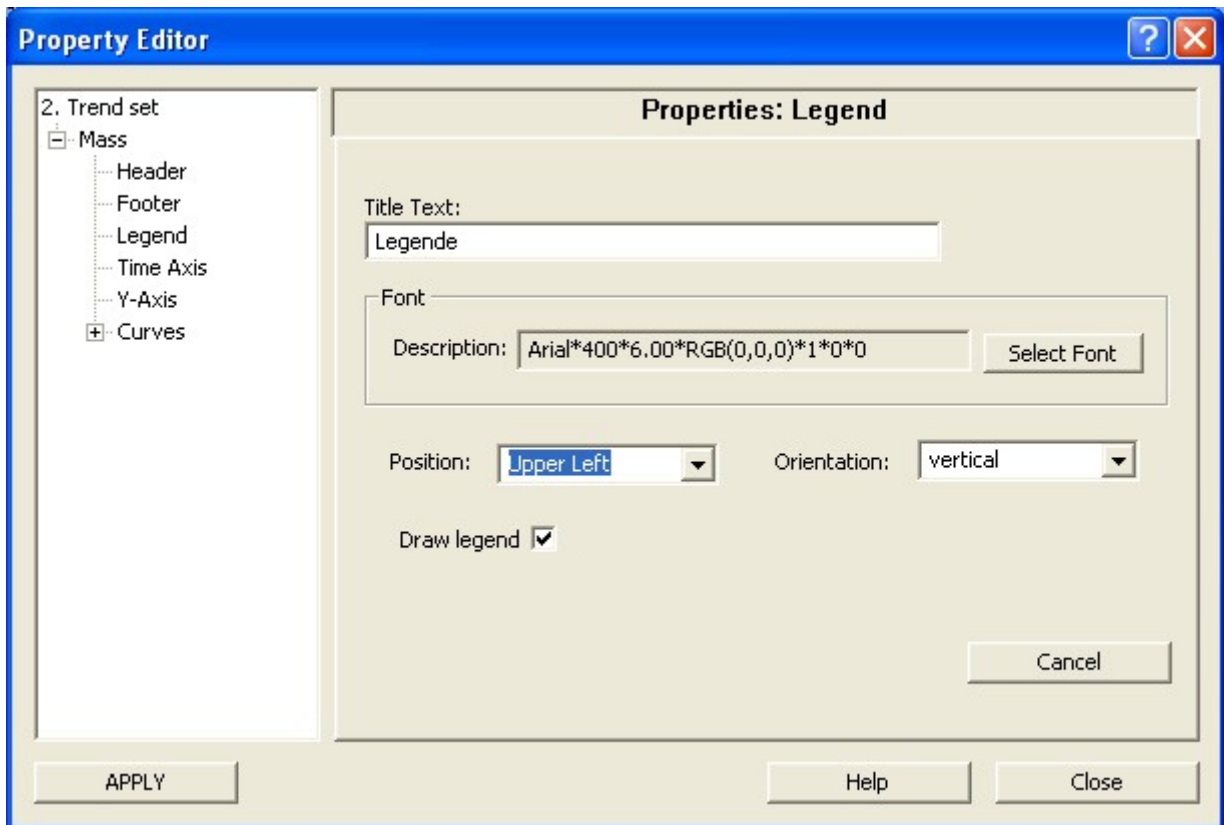
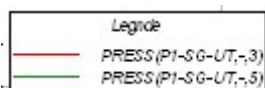


Abb. 3-16 Legend-Dialog im Trend Property Editor

Title Text:

Es kann ein beliebiger Text mit maximal 64 Zeichen eingegeben werden, der als Überschrift über den Kurvenbeschreibungen ausgegeben wird. Der eigentliche Text für die Legendenbeschriftung wird, wie bisher, im Dialog zur Definition der Trendsets festgelegt.



Select Font:

Nach einem Mausklick auf "**Select Font**" öffnet sich das Menü „Schriftart“, in dem die Font-Parameter für die Legenden-Überschrift ausgewählt werden können.

Position:

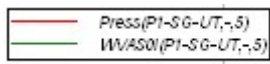
Vier Spezifikationen zur Positionierung der Legende stehen zur Verfügung:

- „Upper Left“: Die Legende wird in der oberen linken Ecke der Plot Area positioniert.
- „Upper right“: Die Legende wird in der oberen rechten Ecke der Plot Area positioniert.
- „Bottom Outside“: Die Legende wird unterhalb der Plot Area positioniert.
- „Position“: Die Positionierung der Legende bleibt unverändert erhalten.

Orientation:

Zwei Spezifikationen zur Anordnung der Kurvenbeschreibungen in der Legende stehen zur Verfügung:

- „**Vertical**“: Die Kurvenbeschreibungen werden untereinander angeordnet.



- „**Horizontal**“: Die Kurvenbeschreibungen werden nebeneinander angeordnet.



Draw Legend:

Wird "Draw Legend" deaktiviert, dann wird keine Legende ausgegeben.

3.1.2.1.4 Time Axis

Mit dem Dialog für „Time Axis“ (Abb. 3-17) ist es möglich, verschiedene Optionen für eine für die Zeitachse des Diagramms (Abszisse) einzustellen.

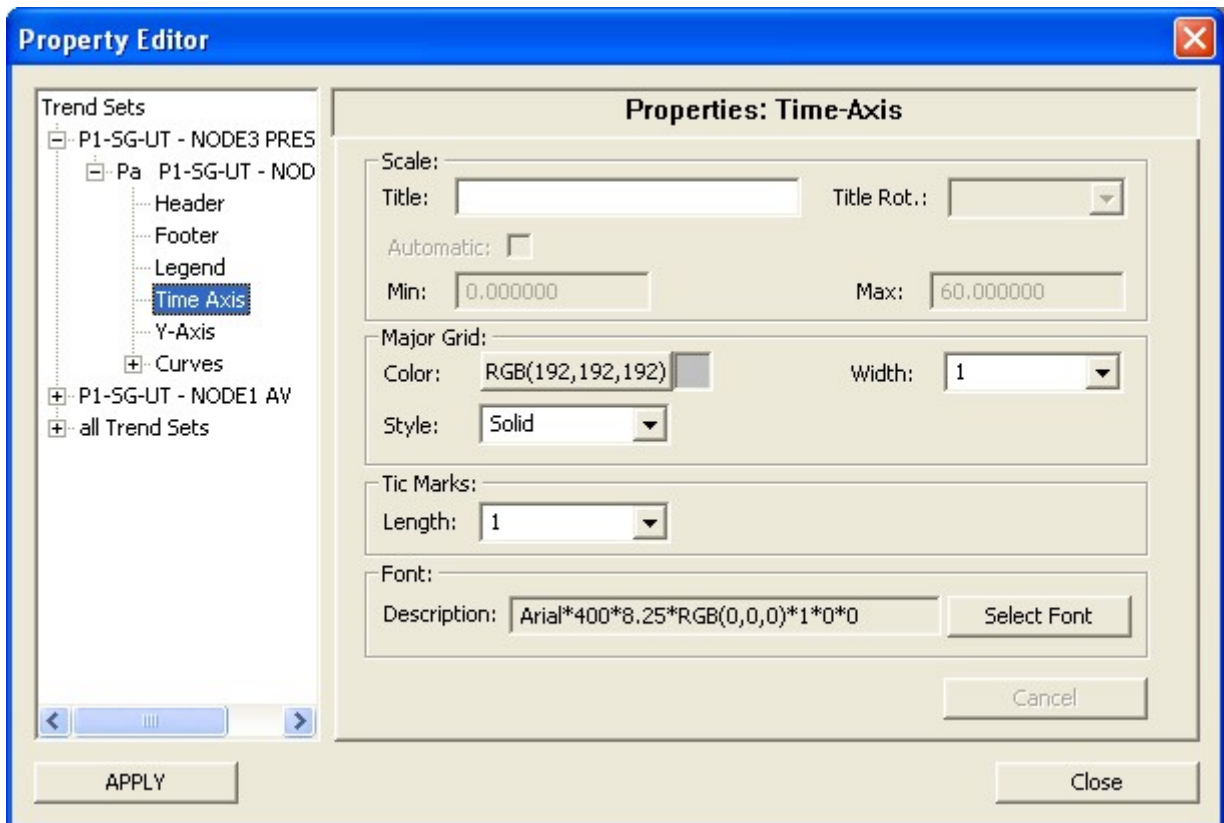


Abb. 3-17 Time-Axis-Dialog im Trend Property Editor

Title:

Eingabe eines beliebigen Texts von maximal 255 Zeichen, der unter der X-Achse angezeigt wird.

Anmerkung: Die übrigen Eingabefelder in der Gruppe „**Scale**“ sind bei der „Time-Axis“ nicht verfügbar.

Major Grid:

Festlegung von Farbe, Linienbreite und Linientyp von Gitterlinien senkrecht zur Zeit-Achse.

- Color: Nach einem Click mit der linken Maustaste auf "RGB(...)" öffnet sich das Farbauswahlmenü.
- Width: Die Linienbreite der Gitterlinien kann zwischen 1 (schmal) und 20 (breit) variiert werden.
- Style: Aus einer Liste von Linientypen (solid, dotted, dashdot etc.) kann ein Typ für die Darstellung der Gitterlinien ausgewählt werden.

Tic Marks:

Die Länge der Tic Marks kann zwischen 1 mm und 20 mm variiert werden.

Font:

Nach einem Mausklick auf "Select Font" öffnet sich das Menü „Schriftart“ in dem die Font-Parameter für die X-Achsen-Beschriftung ausgewählt werden können.

3.1.2.1.5 Y-Axis

Mit dem Dialog für „Y Axis“ (Abb. 3-18) ist es möglich, verschiedene Optionen für die Werteachse des Diagramms (Ordinate) einzustellen.

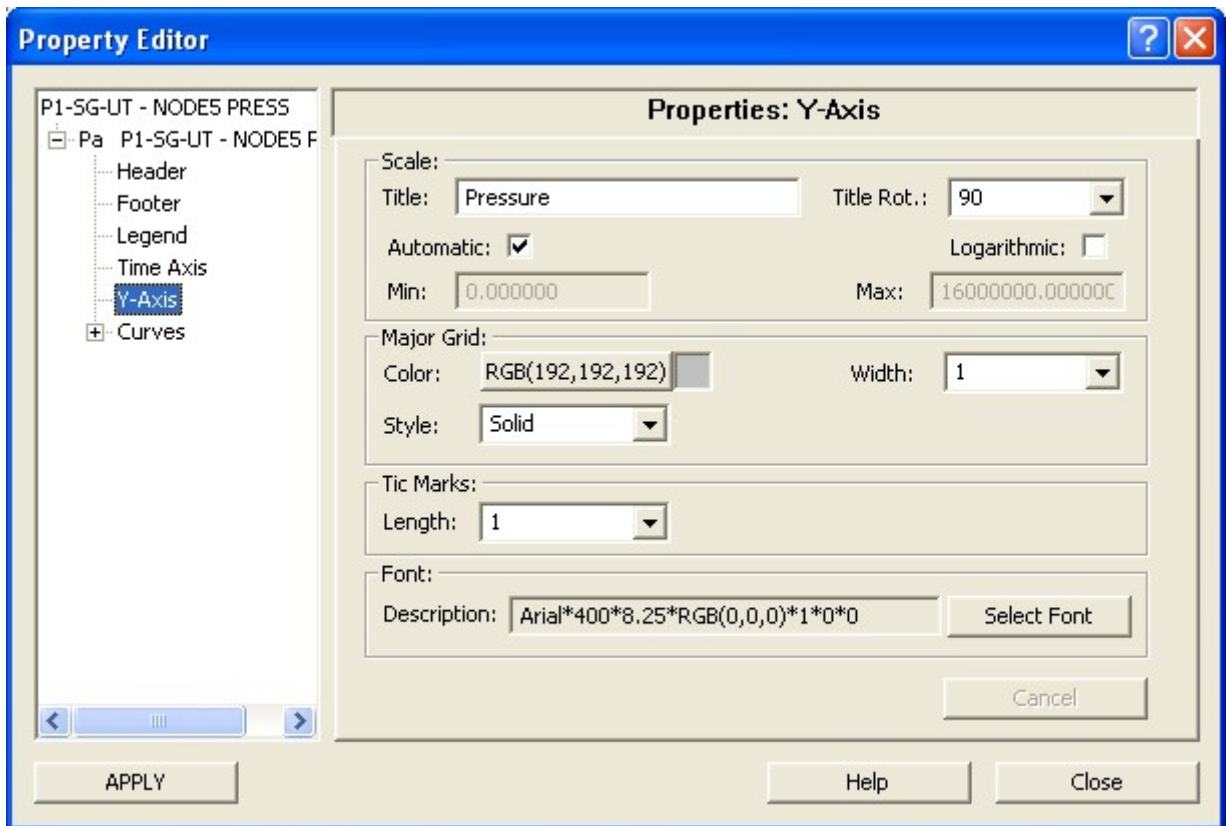


Abb. 3-18 Y-Axis-Dialog im Trend Property Editor

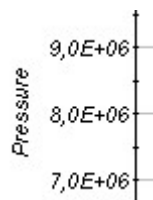
Title Text:

Es kann ein beliebiger Text mit maximal 255 Zeichen eingegeben werden, der links neben der Y-Achse oder über dem Diagramm ausgegeben wird.

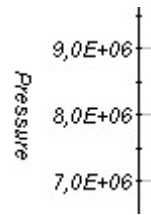
Title Rotation:

Die folgenden Beispiele zeigen die Beschriftung der Y-Achse abhängig von der Angabe unter „Title Rotation“:

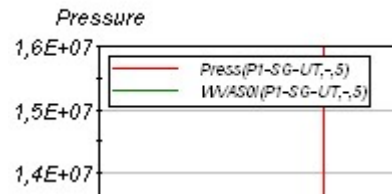
“90”:



"270"



"none"



"unchanged"

Die Position der Y-Achsen-Beschriftung bleibt unverändert erhalten.

Skalierung:

Die Skalierung der Y-Achse kann über die Eingabeparameter „**Automatic**“, „**Logarithmic**“, „**Min**“ und „**Max**“ bestimmt werden. Wird „Automatic“ aktiviert, dann werden die untere und obere Begrenzung der Achse vom Programm bestimmt. Wird „Logarithmic“ aktiviert, dann wird die Y-Achse logarithmisch skaliert.

Major Grid:

Festlegung von Farbe, Linienbreite und Linientyp von Gitterlinien senkrecht zur Y-Achse.

- Color: Nach einem Click mit der linken Maustaste auf „RGB(...)“ öffnet sich das Farbauswahlmenü.
- Width: Die Linienbreite der Gitterlinien kann zwischen 1 (schmal) und 20 (breit) variiert werden.
- Style: Aus einer Liste von Linientypen (solid, dotted, dashdot etc.) kann ein Typ für die Darstellung der Gitterlinien ausgewählt werden.

Tic Marks:

Die Länge der Tic Marks kann zwischen 1 mm und 20 mm variiert werden.

Select Font:

Nach einem Mausklick auf "**Select Font**" öffnet sich das Menü „Schriftart“ in dem die Font-Parameter für die Y-Achsen-Beschriftung ausgewählt werden können.

3.1.2.1.6 Curves

Mit dem Dialog für „Curves“ (Abb. 3-19) ist es möglich, verschiedene Optionen für die Darstellung der Kurvenlinien des Diagramms einzustellen.

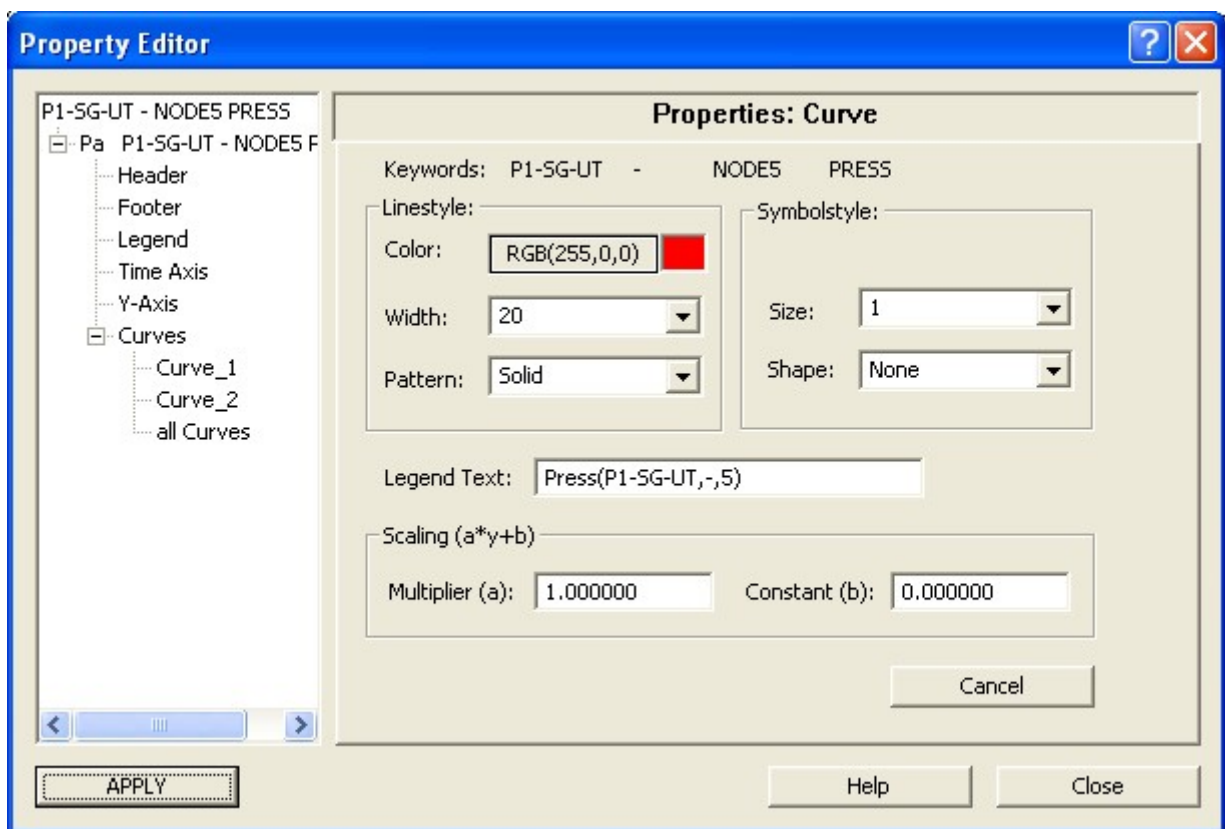


Abb. 3-19 Curve-Dialog im Trend Property Editor

Keywords:

Die vier Keywords, die die dargestellte Kurve definieren, werden hier zur Information angezeigt.

Linestyle:

Festlegung von Farbe, Linienbreite und Linientyp.

- Color: Nach einem Click mit der linken Maustaste auf „RGB(...)“ öffnet sich das Farbauswahlmenü.
- Width: Die Linienbreite kann zwischen 1 (schmal) und 20 (breit) variiert werden.
- Pattern: Aus einer Liste von Linientypen (solid, dotted, dashdot etc.) kann ein Typ für die Darstellung der Linie ausgewählt werden.

Symbolstyle:

- Size: Die Größe der Markersymbole kann zwischen 1 (klein) und 20 (groß) variiert werden.
- Shape: 16 verschiedenen Formen für Markersymbole stehen zur Auswahl. Wird „none“ ausgewählt, dann wird eine Kurve ohne Markersymbole gezeichnet. Im anderen Fall wird jeder Kurvenpunkt mit einem Markersymbol versehen.

Legend Text:

Es kann ein beliebiger Text mit maximal 64 Zeichen eingegeben werden, der in der Legende ausgegeben wird.

Scaling:

Die Funktion dient zum skalieren der Kurvenwerte, wie z. B. Umrechnung von Grad C in Grad F. Zunächst werden die Y-Werte mit a multipliziert und danach wird b addiert.

3.1.2.2 Kerndatenvisualisierung

Die für ATHLET-Simulationen mit 3-D-Neutronendynamik implementierte Funktion /VOG 06/ wurde mit neuen Darstellungsmöglichkeiten ergänzt. Die 3D-Funktion dient zur räumlichen Visualisierung wichtiger physikalischer Größen im Reaktorkernbereich. Sie ermöglicht, für eine ausgewählte axiale Schnittebene, die Auswertung der radialen Kerndaten mit verschiedenen Darstellungsmöglichkeiten und aus beliebigen Blickwin-

keln. Die Daten können optional durch Balken oder durch gekrümmte Flächen mit Farbschattierung dargestellt werden (Abb. 3-20).

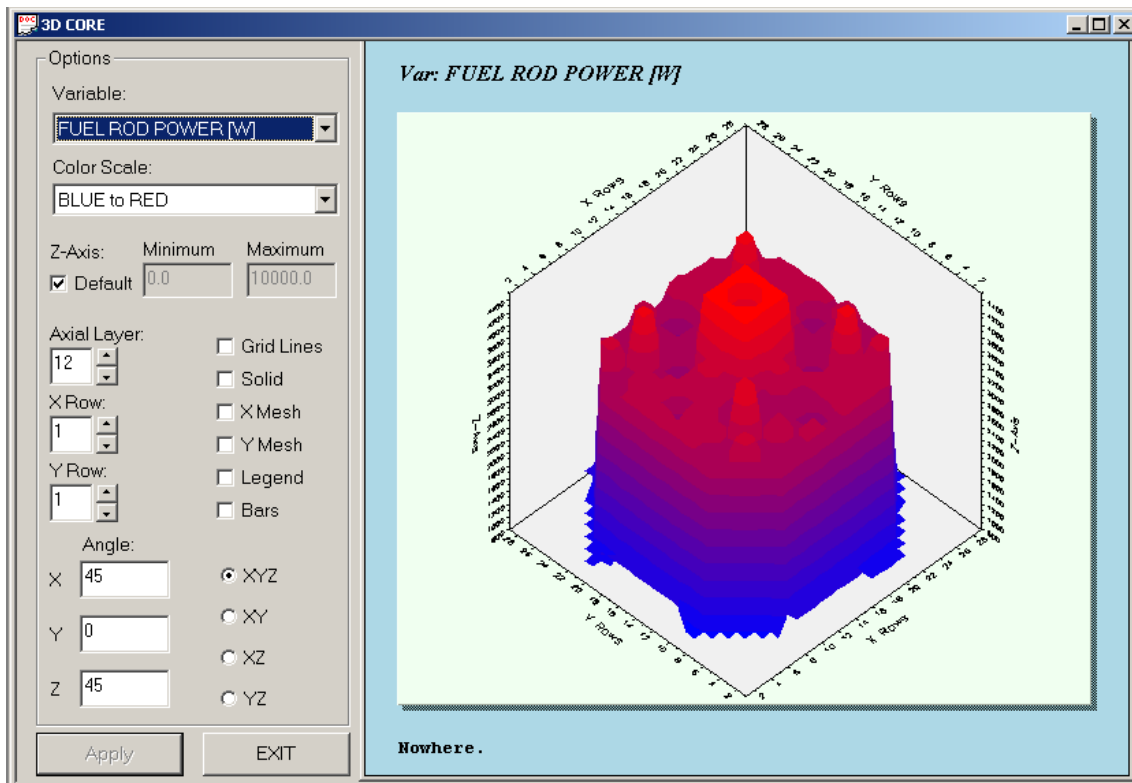


Abb. 3-20 Beispiel für die Darstellung einer Leistungsverteilung im Reaktorkern

Die radialen Daten des Reaktorkerns werden mit farbcodierten Querschnitten für die Kühlkanäle angezeigt. Die zugrunde liegenden Farbskalen können nun im Bediendialog „Options“ ausgewählt und in den Wertebereichen angepasst werden.

Es folgt die Beschreibung der einzelnen Optionen:

- **Variable**

Durch die „Combo Box“ im oberen Bereich der Bedienungs Oberfläche lässt sich die physikalische Größe auswählen, die dargestellt werden soll. Die Liste besteht aus den folgenden Variablen:

- POWER [W]
- COOLANT DENSITY [Kg/m³]
- COOLANT TEMPERATURE [°C]
- FUEL TEMPERATURE [°C]
- VOID [-]

BORON CONCENTRATION [ppm]

PRESSURE [MPa]

MASS FLOW [Kg/s]

- **Color Scale**

Ermöglicht die Auswahl der gewünschten Minimum- und Maximumfarben für die Farbskala. Die Auswahl beschränkt sich auf folgende Farbpaare:

BLUE to RED

BLUE to WHITE

RED to BLUE

RED to WHITE

WHITE to BLUE

WHITE to RED

WHITE to GREEN

- **Z-Axis**

Mit dieser Option können Minimum und Maximum der Z-Achse auf einen fixen von den Daten unabhängigen Wert gesetzt werden. Die Default Werte für Minimum und Maximum richten sich nach den Daten.

- **Axial Layer**

Ermöglicht die Auswahl der gewünschten axialen Schnittebene.

- **X Row und Y Row**

Durch diese Optionen kann der Anwender radiale Querschnitte bei bestimmten XY-Koordinaten festlegen. In Verbindung mit der Option „Solid“, können diese Optionen, durch deutlichere Darstellungen der Datenprofile, zu einer besseren Auswertung der axialen Verteilungen beitragen.

- **Grid Lines**

Schaltet das Gitter für die kartesischen Koordinatenebenen ein oder aus.

- **Solid**

Die Querschnittebenen können mit dieser Option undurchsichtig dargestellt und somit die Datenprofile deutlicher hervorgehoben werden (siehe X und Y Row Optionen).

- **X Mesh und Y Mesh**

Schaltet das Gitter auf der datendarstellenden Fläche zur besseren Hervorhebung der Konturen ein oder aus.

- **Legend**

Schaltet die Farbskala mit dem Wertebereich der gewählten physikalischen Variable ein oder aus.

- **Bars**

Wahl der Darstellung der Daten als Balken oder Fläche.

Für jeden Kernkanal kann mit der Maus der Name des Kanals, der Zeittrend der aktuellen Variablen (z. B. Leistungsdichte) und die axiale Verteilung, also eine Kurve der Variablen als Funktion vom Ort dargestellt werden. Am unteren Rand des Diagramms in Abb. 3-21 ist eine Fußzeile zu sehen. Solch eine Infozeile wird immer dann eingeblendet, wenn der Mauszeiger auf dem Kerndiagramm bewegt wird. Sie gibt außer Namen und Position des gerade angezeigten Kanals, auch den Wert der gewählten physikalischen Größe wieder. Durch einen Klick mit der rechten Maustaste wird, so wie im Bild ersichtlich, ein Menü mit den 3 Auswahlmöglichkeiten angezeigt. Bei der Anwahl von „Distribution“ wird der im folgenden Abschnitt beschriebene Dialog zur Auswahl der Ortsdarstellungen angezeigt.

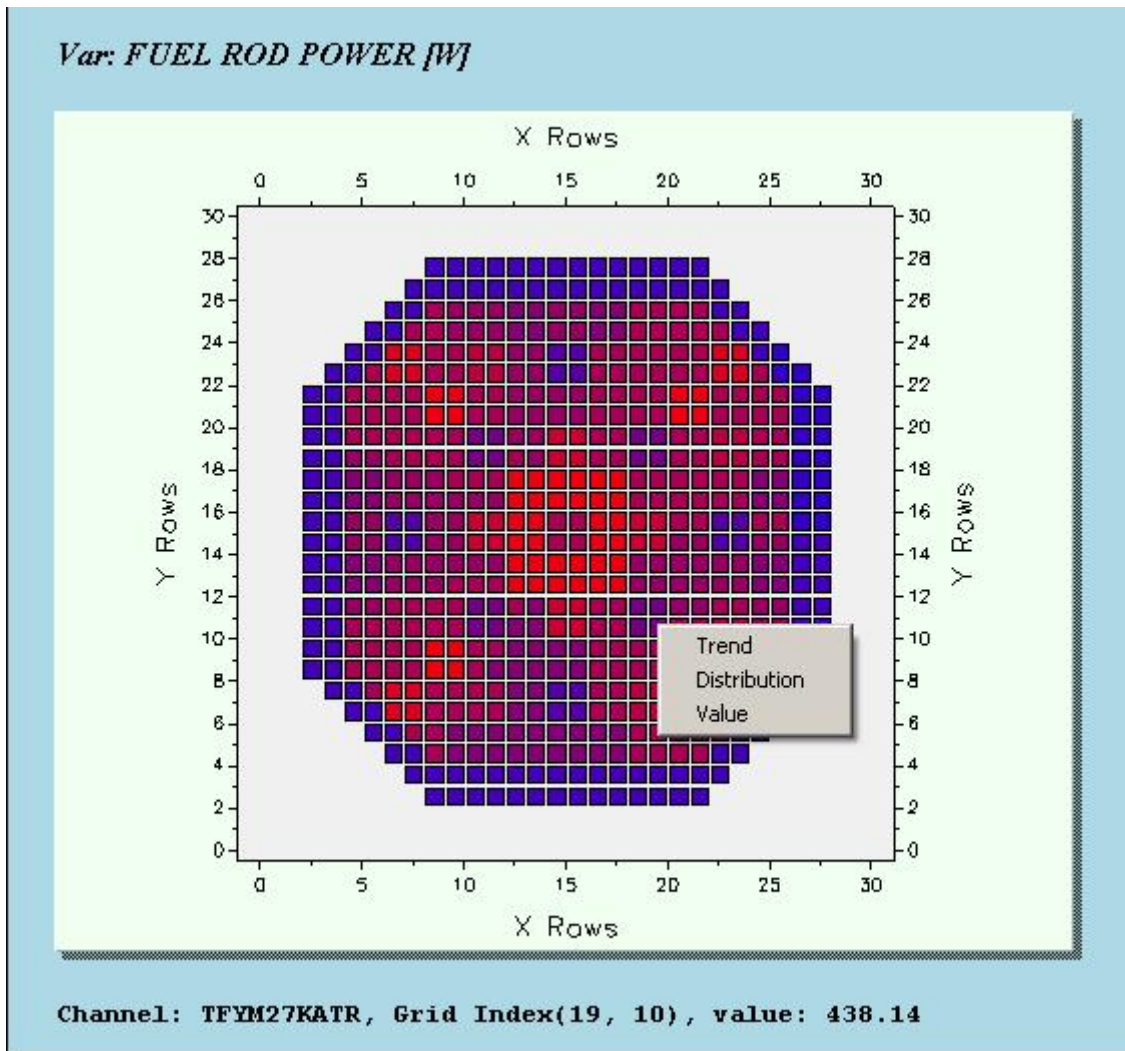


Abb. 3-21 Auswahl von Einzelkanaldaten im Reaktorkern

3.1.2.3 Ortsdiagramme für ATHLET

Da die Kanäle im Reaktorkern Fluidobjekten in ATHLET entsprechen, wurden die Ortsdarstellungen für alle Fluidobjekte verallgemeinert, d. h. es ist nun auch möglich, beispielsweise die Dampfverteilung in einem Dampferzeuger oder einer beliebigen Leitung anzuzeigen.

Durch einen Klick mit der rechten Maustaste auf ein dynamisiertes grafisches Element (siehe auch Abb. 3-21) in einem Bild oder auf eine physikalische Größe eines Fluidobjekts in der Keyword-Liste wird ein Menü mit den drei Schaltflächen „Trend“, „Distribution“ und „Value“ aktiv. Die Auswahl der Schaltfläche „Distribution“ öffnet einen Dialog (Abb. 3-22) mit Darstellungsoptionen von Ortsverteilungen von physikalischen Größen.

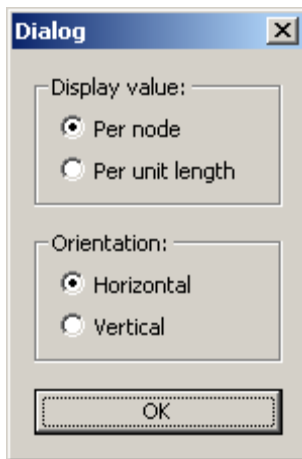


Abb. 3-22 Darstellungsoptionen von Ortsverteilungen

Die Darstellungsrichtung (waagrecht oder senkrecht) ist wählbar. Abb. 3-24 zeigt das horizontale Axialdiagramm einer Leistungsdichte im Reaktorkern. Die durch Dreiecke markierten Punkte kennzeichnen die Werte der Variablen in den vorhandenen Zonen des Fluidobjekts. Wenn der Benutzer eine spezielle Ergebnisdatei von ATHLET („*.gr-Datei“) im Arbeitsverzeichnis zu Verfügung stellt, wird auf der X-Achse die Länge des Objekts wiedergegeben und ist auch die Option „Per unit length“ aktiv.

Wenn diese Option gewählt wird, werden die angezeigten Werte durch die Länge der Zonen geteilt. Dadurch ist das Profil der Variable entlang des Objekts nicht mehr von der Länge der einzelnen Zonen abhängig und kann somit in einigen Fällen realistischer dargestellt werden (Abb. 3-24). Bei den Zonen 17 und 18 mit kleinerer Länge zeigt das Leistungsdiagramm in Abb. 3-23 einen Sprung, der aber keine physikalische Unstetigkeit im Leistungsprofil bedeutet wie aus Abb. 3-24, dem Profil der Leistungsdichte, ersichtlich.

Dabei ist zu beachten, dass die Option „Per unit length“ nur für extensive Größen sinnvoll benutzt werden kann, d. h. nur für die physikalischen Größen, deren Betrag dem Volumen der entsprechenden Zone proportional ist.

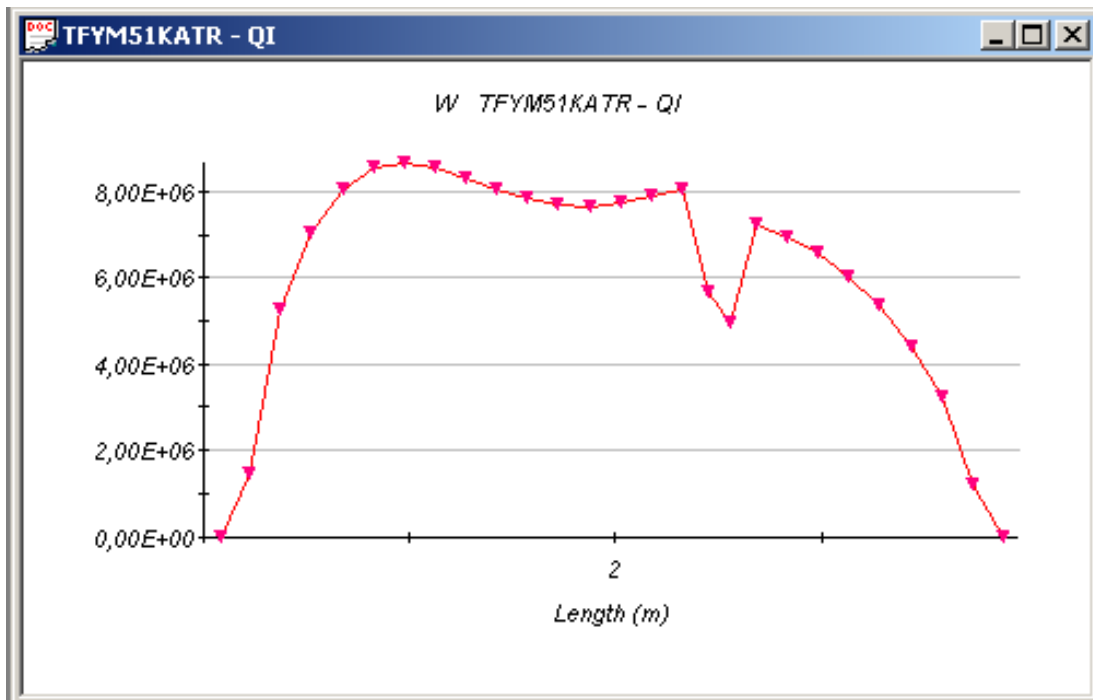


Abb. 3-23 Axialdiagramm einer Leistung im Reaktorkern

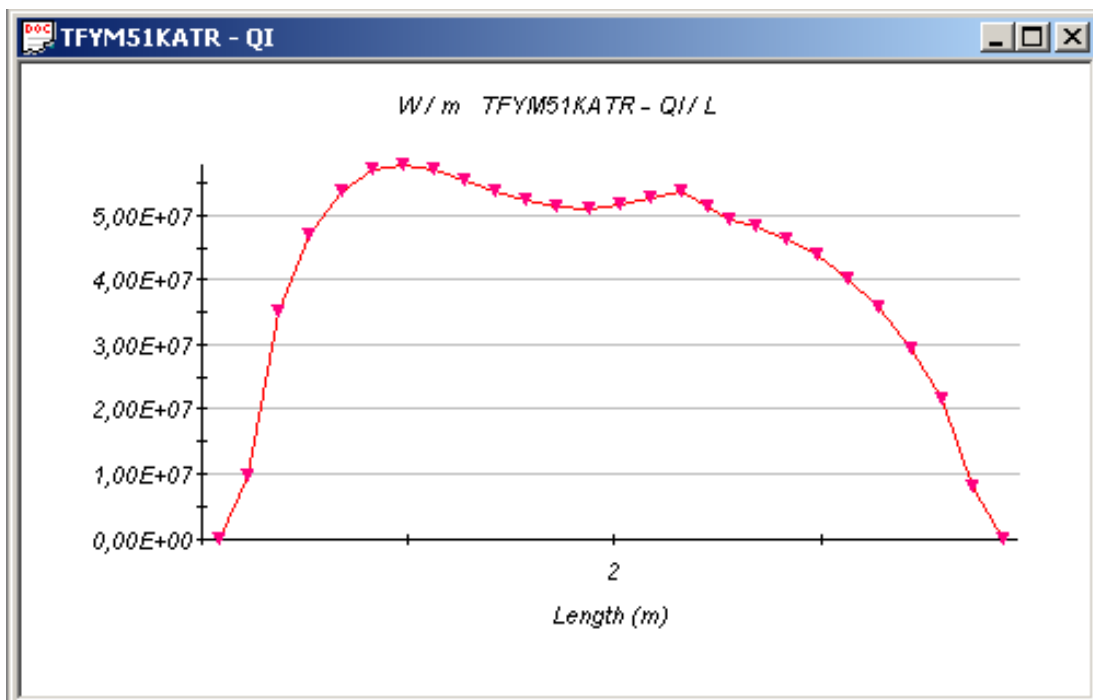


Abb. 3-24 Axialdiagramm einer Leistungsdichte im Reaktorkern

3.1.3 Darstellung der Nodalisierung und der Brennstäbe in Verbindung mit dynamischen Daten für ASTEC

Eine grafische Unterstützung zur Überprüfung und Visualisierung der Geometrie und der Verknüpfungen von Thermohydraulikelementen und Brennstäben, ähnlich wie für ATHLET vorhanden, wurde für ASTEC /ALL 05/ entwickelt. Mit dem automatisierten Import dieser Grafiken in ATLAS können auch weniger erfahrene Anwender aussagefähige Prozessbilder für die Auswertung der Simulationsergebnisse verwenden, ohne dass sie die komplexe Bilderstellung mit dem Bildeditor APG erlernen müssen.

Die benötigten Daten werden der zentralen ASTEC - Datenbank (SIGAL) entnommen und aus der Geometrie- und Verbindungsinformation wird automatisch ein Diagramm erstellt. Dieses kann interaktiv nach den Wünschen des Anwenders angepasst werden, beispielsweise durch Verschieben von Objekten oder eine Vektordarstellung der Junctions. Verschiedene Darstellungsvarianten sind speicherfähig und können ohne weiteren Aufwand als Bilder für ATLAS exportiert werden. In diesen werden die dynamischen Daten wie Temperaturen und Dampfgehalte entsprechend verschiedener Farbskalen angezeigt.

Für das ASTEC Modell zur Berechnung des Brenn- und Steuerstabverhaltens (DIVA) können ebenfalls automatisch schematische Diagramme erzeugt werden. Mit den vom Benutzer gewählten Stäben kann dann eine entsprechende Darstellung für ATLAS erzeugt werden, die automatisch mit zeitveränderlichen Daten animiert wird. Zurzeit können Temperatur und Masse mit veränderbaren Farbcodierungen angezeigt werden, mittels Mausclick sind Werte und Zeitverläufe verfügbar. Neben der Färbung ist auch die Geometrie der Stabdarstellung veränderlich. Damit kann in den einzelnen axialen Zonen die Ausdehnung bzw. das Abschmelzen der Brennstäbe während der Aufheizphase visualisiert werden.

3.1.3.1 Programmbeschreibung

Das Programm ASTEC Input Grafic (ASIG) ermöglicht die grafische Darstellung von Geometrie und Nodalisierungsdaten der Objekte (Thermofluidobjekte + Junctions) des Reaktorkühlkreislaufs sowie der Brenn- und Steuerstäbe eines ASTEC-Eingabedatensatzes.

Es dient dem Benutzer sowohl zur Überprüfung und Dokumentation der Eingabedaten als auch als Grundlage zur grafischen Darstellung der örtlichen Verteilung thermohydraulischer Größen wie Dampfgehalt oder Fluidtemperatur im Rahmen der Simulatoranwendung.

Das Programm bietet zwei Grundoptionen:

1. Darstellung der Thermofluidobjekte (TFOs) und Junctions

- Schematische Darstellung aller oder ausgewählter TFOs und deren Verknüpfungen.
- Darstellung der Geometrie und der Nodalisierung aller oder ausgewählter TFOs und deren Verknüpfungen.

2. Darstellung der Brennelemente

- Abbildung der Geometrie und der Nodalisierung aller oder ausgewählter Brennstäbe (Rods) mit oder ohne Kontrollstäbe.

3.1.3.2 Programmentwicklung

Als Grundlage für die Programmentwicklung hat die Windows-Version der ATHLET Input Grafic (AIG) gedient und Programmarchitektur und Organisation konnten in wesentlichen Teilen übernommen werden. Das Programm besteht aus zwei Hauptteilen, einem Programmkern für Berechnungen in FORTRAN und einer Bedienungsoberfläche in C++, die durch ein im FORTRAN geschriebenes Interface kommunizieren.

Da die im ATHLET definierten Thermofluidobjekte ähnliche geometrische Eigenschaften zu den ASTEC-Objekten und Junctions aufweisen, benutzt ASIG ebenfalls wie AIG eine Eingabedatei mit dem gleichen Format des AIG-Inputfiles (Dateierweiterung „*.gr“), in der die Objekteigenschaften gespeichert sind. Zu diesem Zweck ist ein eigenständiges Programm geschrieben worden, das eine solche Eingabedatei aus der Standard-Ausgabedatei von ASTEC (Restart-File) erzeugen kann. Die Ausgabedatei von ASTEC ist binär und muss daher mit speziellen Schnittstellenprogrammen gelesen werden („Sigal-Routinen“). Nach dem erfolgreichen Abschluss dieser Phase, begannen die Arbeiten der Anpassung der Bedienoberfläche an die besonderen Gegebenheiten bei ASTEC. Die letzte Phase der Programmierung bestand darin, die vorhandenen

Routinen zur Erzeugung der APG-Dateien für die Simulatoranwendung den ASTEC-Spezifikationen der Objekte, Junctions, Brennelemente und Kontrollrods anzupassen.

3.1.3.3 Programmstruktur

In der ASIG lassen sich 4 Haupt-Programmteile unterscheiden, der Berechnungsteil, die Bedienungsoberfläche, die Kommunikation zwischen Bedienungs- und Berechnungsteil und der Grafikeil zur Erstellung und Ausgabe der Bilder.

3.1.3.4 Berechnungsteil

Im sogenannten Berechnungsteil von ASIG werden die Daten eingelesen, die Koordinaten aller im Bild dargestellten Objekte berechnet, die gewählten Optionen in einer Datei gespeichert und die Bilddateien (APG-Files) für ATLAS ausgegeben. Der Berechnungsteil besteht aus folgenden Routinentypen:

Input-Routinen

Durch die Input-Routinen werden zwei Dateien eingelesen. Sie sind:

Ein durch ASTEC generiertes Restartfile mit dem Suffix “*.std“, das Geometrie-, Netzwerk- und Nodalierungsdaten aller vorhandenen Objekte enthält.

Das Design Data Set (DDS), das die notwendigen Optionen enthält, die das Programm zur gewünschten Bilddarstellung benötigt.

Berechnungsroutinen

Diese Routinen berechnen die Koordinaten aller Punkte der ausgewählten Objekte. Die in globalen Variablen gespeicherten Werte werden von den grafischen Routinen zur Bilddarstellung verwendet.

Output-Routinen

Durch die Output-Routinen werden zwei Dateien ausgegeben. Sie sind:

Das Design Data Set (DDS), das die Optionen zur gewünschten Bilddarstellung enthält.

Die ATLAS-Bilddatei, die Befehle der Grafikbeschreibungssprache APG enthält.

3.1.3.5 Bedienungsteil

Dieser Programmteil ist gänzlich in C++ geschrieben worden. Basis dafür sind die Ressourcen und die Methoden der Foundation Class Library von Microsoft. Die MFCL bietet nicht nur eine komfortable Entwicklungsumgebung für die Programmierung von Bedienungsoberflächen. Dem Programmierer stehen auch eine Reihe vorprogrammierter Tools zur Verfügung, die er ohne jeglichen Aufwand in die eigene Applikation einbeziehen kann. Einige solcher standardisierter Tools - wie z. B. Bedienungsoberfläche zum Drucken, zur Farbauswahl oder zum Auswählen und Öffnen von Dateien - konnten vorteilhaft in ASIG integriert werden. Die eigenen Menüs für die Programmbedienung sind alle mit den verfügbaren Ressourcen der MFCL entwickelt worden.

3.1.3.6 Grafikteil

Zur Programmierung des Grafikteils werden die Routinen der Grafikbibliothek OpenGL verwendet. OpenGL, das sich als neuer Grafikstandard für zwei- und dreidimensionale Grafik etabliert hat, wird auch für ATLAS verwendet. Die Funktionen können sowohl von C++ als auch von FORTRAN aus aufgerufen werden.

3.1.3.7 Programmbedienung

3.1.3.7.1 Erste Schritte

Nach dem Start des Programms müssen zwei Dateien geöffnet werden, das durch einen ASTEC-Lauf generierte Restartfile mit dem Suffix "*.std" und das Design Data Set (DDS) für ASIG Bildoptionen. Wenn das angegebene DDS nicht existiert, wird unter dem gewählten Namen ein neues angelegt.

Nach der Wahl der Dateien kann bestimmt werden, welche Objekte dargestellt werden sollen. Dafür stehen dem Anwender zwei Objektauswahllisten zur Verfügung. Sie kön-

nen unter dem Menüpunkt „View“ (Abb. 3-25) durch Anklicken von „Tfo's List“ bzw. „Rods List“ ein- oder ausgeblendet werden. Die Ausführung dieser wenigen Schritte reicht bereits zur Erstellung einer ersten grafischen Darstellung der Eingabedaten.

3.1.3.7.2 Die Hauptfunktionen des Programms

Die Hauptfunktionen von ASIG sind in Abb. 3-25 markiert und haben die folgende Funktionalität:

1. Öffnen der Inputdateien (ASTEC-Restartfile und Design Data Set)
2. Speichern des Design Data Set
3. Öffnen des Menü mit den Optionen für Objekte und Junctions
4. Öffnen des Menü mit den Optionen für Rods und Control Rods
5. Öffnen des Menü mit den Optionen für die Koordinatenachsen
6. Ausdrucken des Bilds im Hauptfenster
7. Öffnen des objektorientierten Menü
8. Auswahlliste der Objekte
9. Eingabefeld um Objekte zu suchen und im Bild einfügen bzw. ausschließen

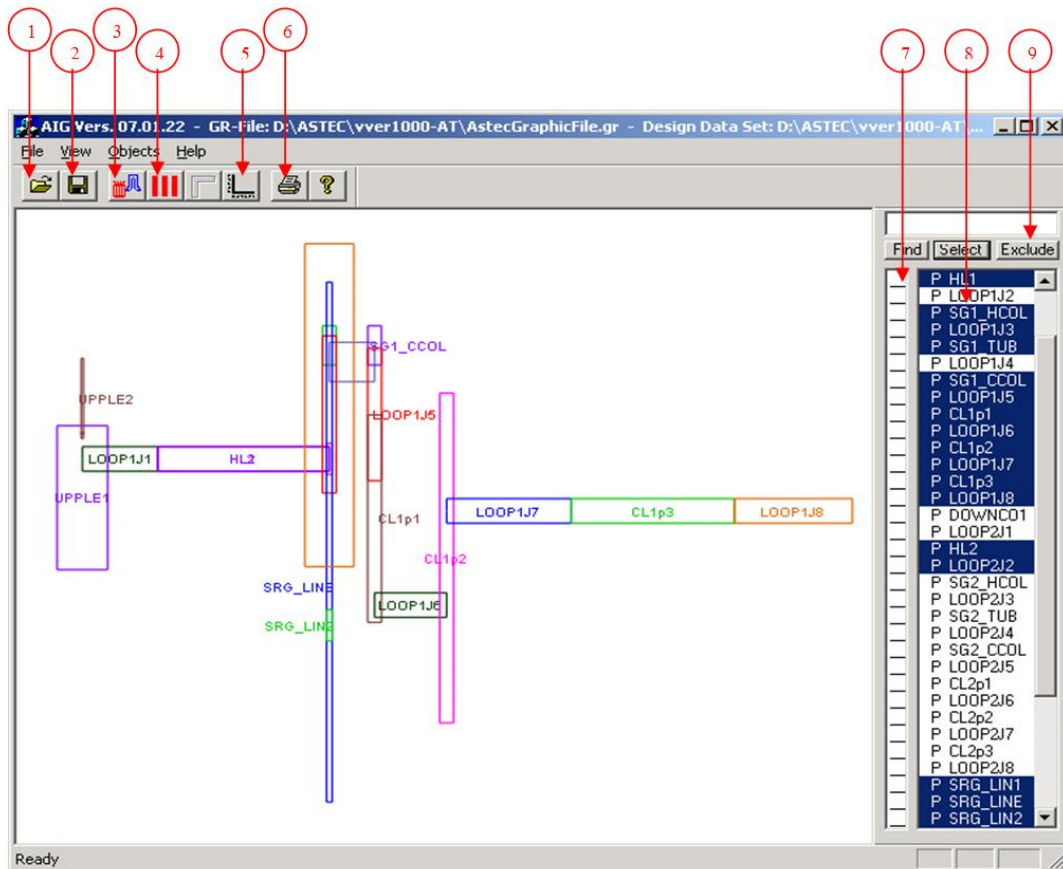


Abb. 3-25 Hauptfenster von ASIG

Die folgende Liste zeigt die unter der Option „File“ ansprechbaren Funktionen:

Open Files ...	Ctrl+O
Save Design	Ctrl+S
Save Design As ...	Ctrl+A
Picture As Bitmap...	Ctrl+B
Create APG Files ...	Ctrl+C
Print Picture ...	Ctrl+P
Exit	Ctrl+X

Die Funktionen „Open Files“ und „Save Design“ entsprechen den bereits beschriebenen Optionen (1) und (2). Die Funktion „Create APG Files“ dient zur Erstellung von Bilddateien für ATLAS mit den notwendigen Verknüpfungen zu den dynamischen Ergebnisvariablen. Diese Bilder können direkt in Simulationläufe eingebunden werden.

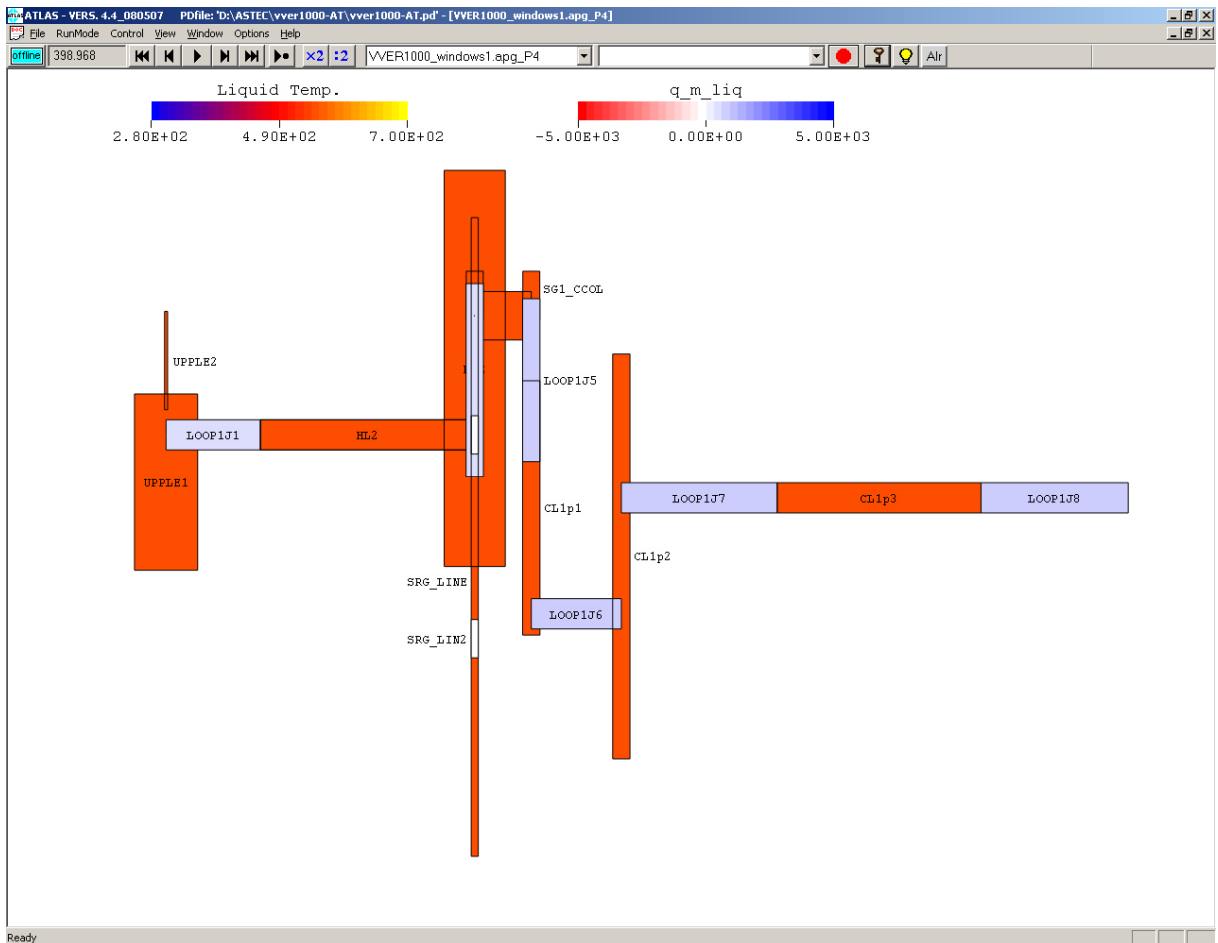


Abb. 3-26 Verwendung eines ASIG-Bildes für Thermohydraulikobjekte in ATLAS

Abb. 3-26 zeigt einen Zeitpunkt in einem ATLAS-Lauf, in dem das durch die ASIG erzeugte APG-File verwendet wurde, das dem geometrischen Aufbau von Abb. 3-25 entspricht. Für die zwei Typen von dargestellten Objekten (Thermofluidobjekte und Junctions) werden verschiedene Farbskalen verwendet. Für die Thermofluidobjekte (orangefarbig im Bild) können neben Wasser- und Dampftemperatur, auch die Verteilungen für Dampfvolumenanteil und Dampfgehalt dargestellt werden. Für die Junctions (blau im Bild) können Massenströme und Geschwindigkeiten beider Phasen dargestellt werden.

In Abb. 3-27 werden zwei Brennstäbe (Rods) mit den zugehörigen Steuerstäben (Control Rods) grafisch dargestellt. Im Bild ist erkennbar, dass die Dynamik dieser Grafiken sich nicht nur auf die Darstellung von physikalischen Größen (Temperatur und Masse) durch Veränderung der Flächenfärbung beschränkt. Die Geometrie der Flächen ist ebenfalls veränderlich, so dass, während des Laufs auch Geometrieänderungen der Stäbe angezeigt werden können. In der Abbildung wird eine Phase gezeigt, in der die

Schichten der Rods durch Aufheizung stark verändert sind, wobei der erste Rod und Control Rod bereits teilweise geschmolzen sind.

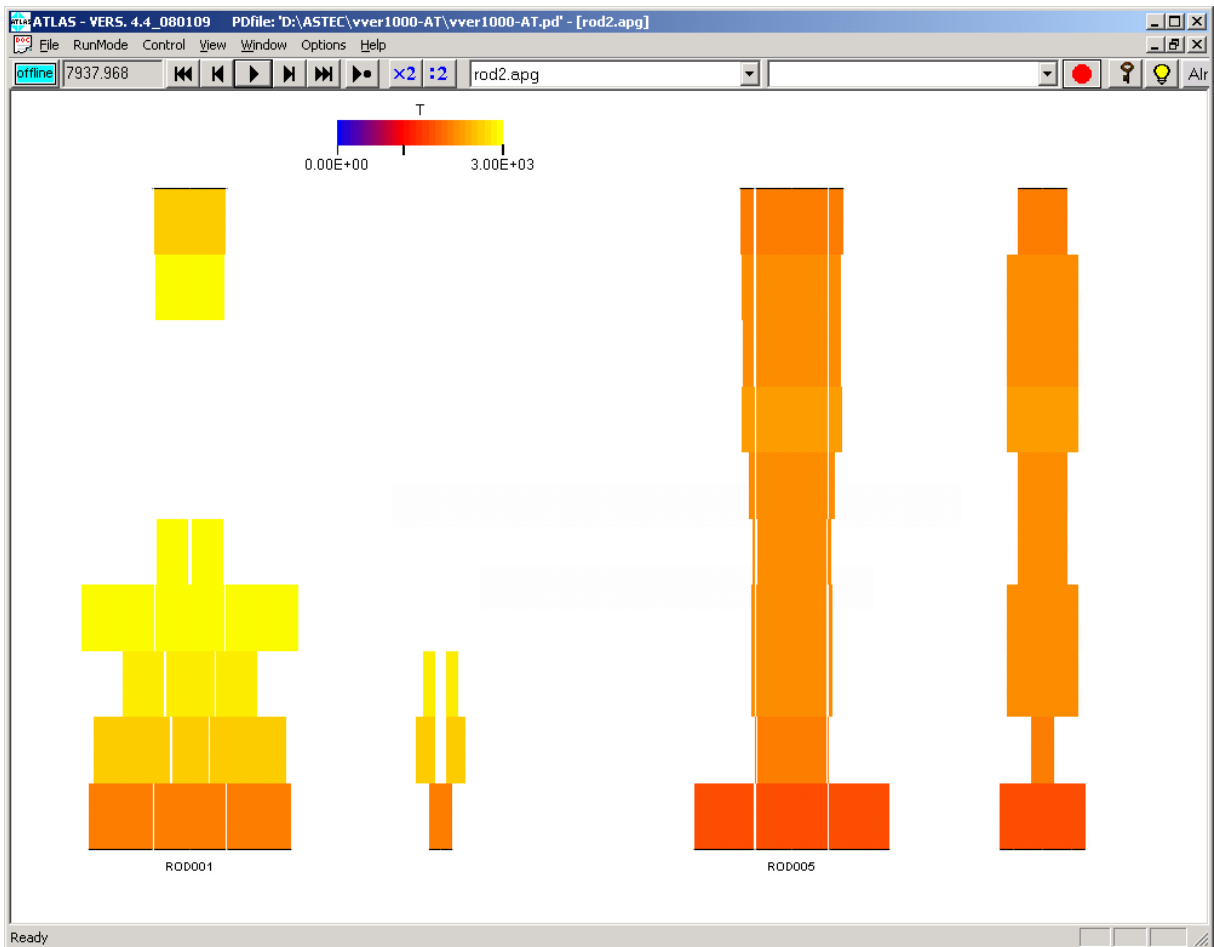


Abb. 3-27 Verwendung eines ASIG-Bildes für Brennstabobjekte in ATLAS

3.1.3.7.3 Die Auswahllisten

Über die Hauptoption „View“ aus der oberen Menüleiste lassen sich die Auswahllisten der Objekte und Junctions (in Abb. 3-25 mit (8) gekennzeichnet) oder die Auswahlliste der Rods ein- bzw. ausschalten. Sie bieten die Möglichkeit zu bestimmen, welche Objekte im Bild dargestellt werden sollen. Die weißgefärbten Bezeichnungen mit blauer Hinterlegung beziehen sich auf die im Bild ausgewählten Objekte. In der Auswahlliste der Objekte und Junctions kennzeichnet ein „P“ oder ein „S“ die Zugehörigkeit zum primären bzw. sekundären Kreislauf.

Das Anklicken einer Bezeichnung bewirkt die Auswahl bzw. den Ausschluss des entsprechenden Objekts aus der grafischen Darstellung. Durch die Tasten „Select“ und

„Exclude“ können Gruppen von Objekten in das Bild ein- oder ausgeblendet werden. Die Selektion erfolgt „Wildcards“ nach den DOS-Konventionen (z. B. *SRG). Durch An-klicken des quadratischen Feldes links vom Objekt-namen kann man das objektorien-tierte Menü (s.u.) ansprechen.

3.1.3.7.4 Die Hauptmenüs der Objekte

Die folgenden Abbildungen zeigen Gestaltung und Funktionen der Hauptmenüs der Objekte. Die angewählten Optionen wirken sich auf alle Objekte aus.

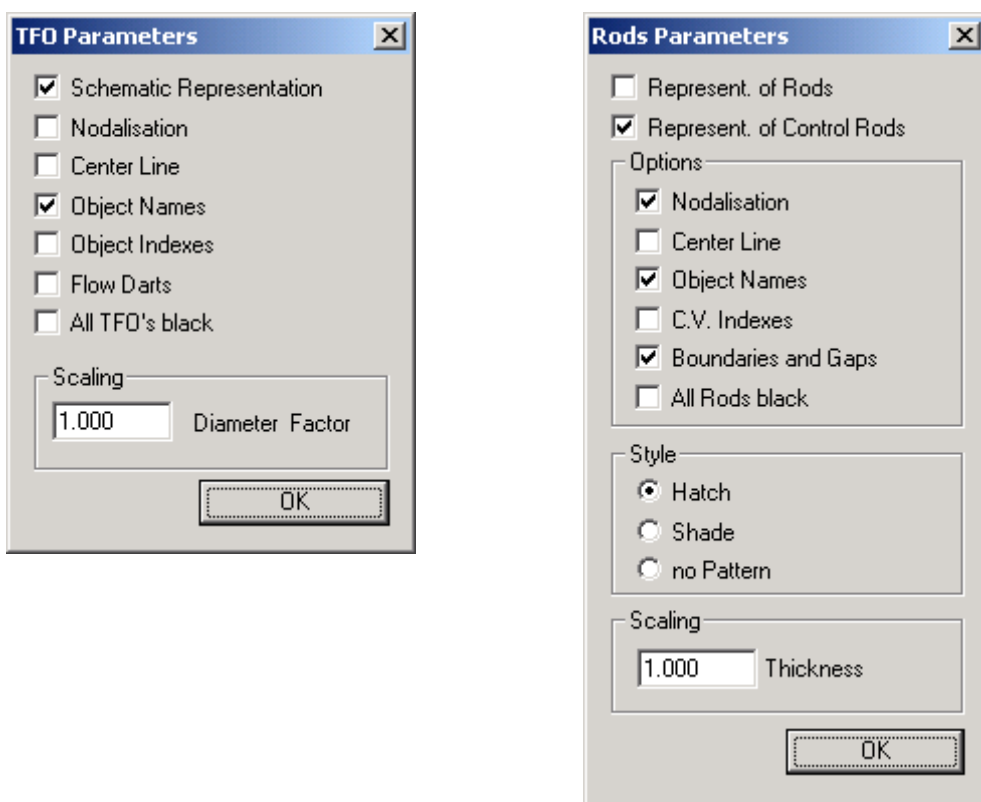


Abb. 3-28 Globale Optionen zur Darstellung von Objekten in ASIG

Das Hauptmenü der TFO kann entweder über die Symbolleiste (Ikone (3) in Abb. 3-25) oder durch die Option „View“ aus der Menüleiste des Hauptfensters aufgerufen werden.

- **„Schematic Representation“**

Nach der Auswahl dieser Funktion wird eine schematische Darstellung der vorhandenen TFOs mit deren Verknüpfungen angezeigt (siehe Abb. 3-29).

Die Objekte werden durch Vierecke, ihre Verbindungen durch Pfeile dargestellt. Die Definitionsrichtung der Strömung wird durch die Richtung der Pfeile angezeigt.

- **„Nodalisation“**

Falls Zonen (nodes) vorhanden sind, werden diese durch Trennlinien abgegrenzt, in denen die einzelnen Objekte aufgeteilt sind.

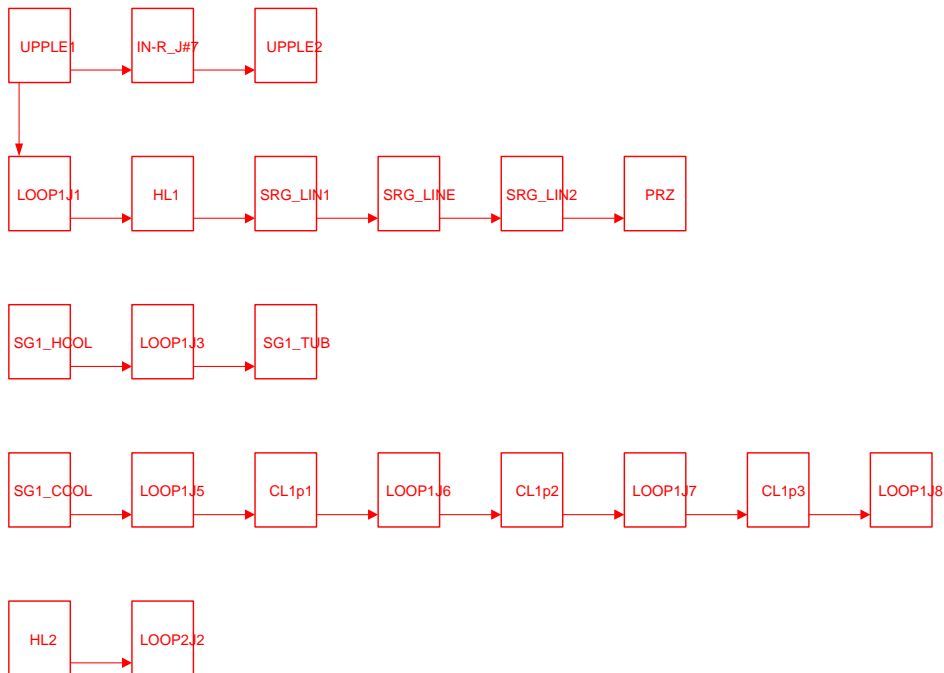


Abb. 3-29 Schematische Darstellung der Verknüpfung von Objekten in ASIG

- **„Center line“**

Die Mittellinie des Verlaufs der Objekte wird ein- bzw. ausgeblendet.

- **„Flow Darts“**

Zur Darstellung der Strömungsrichtung werden Pfeile ein- oder ausgeblendet.

- **“Object names” und “object indexes”**

Die Objektnamen bzw. -nummern werden ein- oder ausgeblendet.

- **“All TFO’s black”**

Alle Objekte werden einheitlich schwarz dargestellt (sinnvoll fürs Drucken auf einem S/W Drucker).

Das Hauptmenü der Rods kann entweder über die Symbolleiste (Ikone (4) in Abb. 3-25 oder durch die Option „View“ aus der Menüleiste des Hauptfensters aufgerufen werden.

- **“Represent. of Rods”**

TFO und Rods können nicht gleichzeitig in einem Bild grafisch dargestellt werden. Ist diese Option aktiv, dann werden nur die Rods dargestellt ansonsten werden nur die TFO dargestellt.

- **“Represent. of Control Rods”**

Durch diese Option bestimmt der Anwender, ob die Control Rods grafisch dargestellt werden sollen oder nicht.

- **„Nodalisation“**

Falls vorhanden, werden durch Trennlinien die Zonen (Nodes) abgegrenzt, in denen die einzelnen Rods aufgeteilt sind.

- **„Center line“**

Die Mittellinie des Verlaufs der Objekte wird ein- bzw. ausgeblendet.

- **“Object names”**

Die Objektnamen werden ein- oder ausgeblendet.

- **“C.V. indexes”**

Durch diese Option können die Zonen, in denen die Rods aufgeteilt sind, durchnummeriert werden.

- **“Boundaries and Gaps”**

Ist diese Option eingeschaltet, werden bei den Brennstäben die Schichten Brennstoff, Brennstabspalt und Hüllrohr als drei voneinander getrennten Zonen dargestellt. Ist die Option nicht eingeschaltet, werden die Brennstäbe als eine einzige Zone, ohne Unterscheidung von Schichten dargestellt.

- **“All Rods black”**

Alle Objekte werden einheitlich schwarz dargestellt (sinnvoll fürs Drucken auf S/W Drucker).

- **“Style”-Optionen**

Hier wird bestimmt, welche Art von Schraffierung die Fläche der dargestellten Rods erhält.

3.1.3.7.5 Die objektorientierten Menüs

Mit diesen Menüs können Optionen eingestellt werden, die sich auf die Darstellung einzelner Objekte (Abb. 3-30) beziehen.

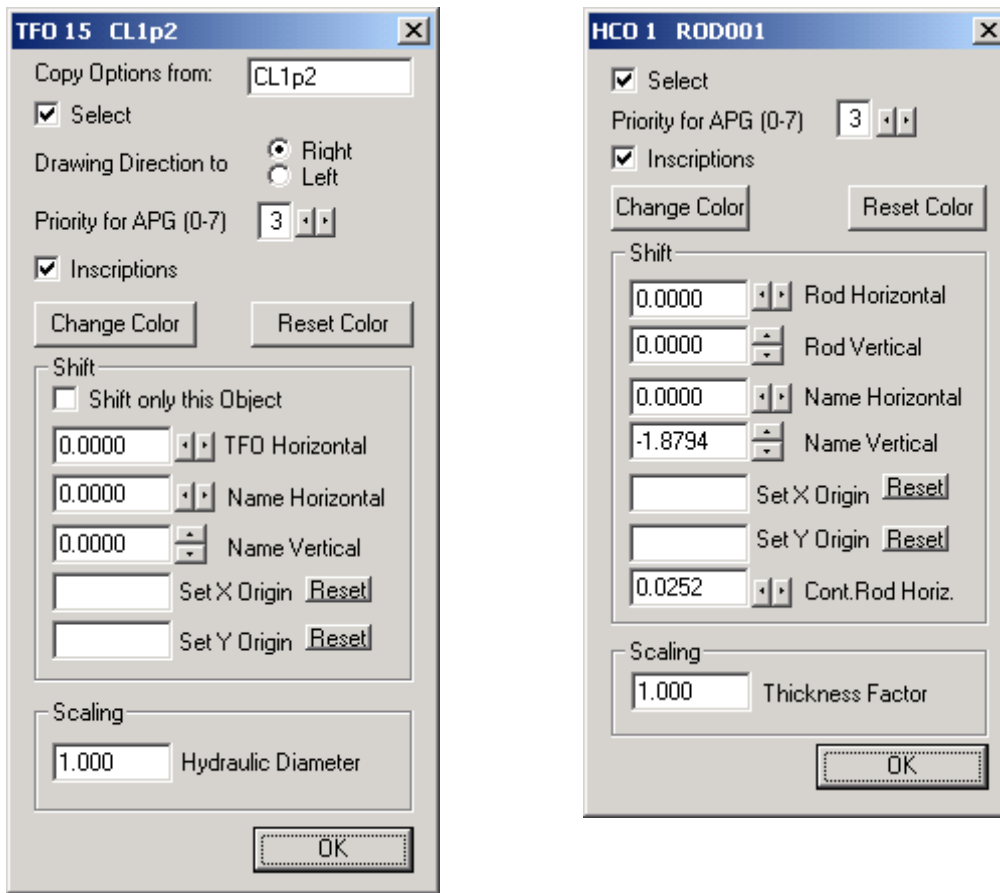


Abb. 3-30 Optionen bei der Darstellung von einzelnen Objekten in ASIG

Dem Anwender stehen zwei verschiedene Wege zur Verfügung, um diese Menüs anzuzeigen:

1. Man klickt das Objekt mit der Maus an.
2. Man klickt das Feld links vom Objektname in der Auswahlliste der Objekte an.

Die Optionen des Menüs für TFO sind die folgenden:

- **“Copy Options from:”**

Durch diese Option lassen sich für das gegenwärtige Objekt alle Optionen eines anderen beliebigen Objekts, übernehmen.

- **“Select”**

Bestimmt, ob das Objekt grafisch dargestellt werden soll oder nicht.

- **“Drawing Direction to the ...”**

Diese Option bestimmt, ob die Darstellung des Objekts von links nach rechts oder umgekehrt verlaufen soll.

- **“Priority for APG (0-7)”**

Diese Option ist für Objekte wichtig, die man bei der ATLAS-Anwendung im Bild sehen möchte, die aber von anderen ganz oder teilweise überdeckt werden. Jedem Objekt kann eine „Prioritätsstufe“ von 0 bis 7 gegeben werden, wobei die Objekte mit höherer Priorität im Vordergrund, die mit niedriger Priorität im Hintergrund erscheinen.

- **“Inscriptions”**

Für das gegenwärtige Objekt kann durch diese Option Objektname und Nummer ein- oder ausgeschaltet werden.

- **“Change Color”**

Die Farbe eines Objekts ändern.

- **“Reset Color”**

Einem Objekt seine ursprüngliche Farbe geben.

- **“Shift only this Object”**

Diese Option steht im direkten Zusammenhang mit der nächsten und bestimmt bei Verschieben eines Objekts, ob auch die verbundenen Objekte mitverschoben werden sollen.

- **“TFO Horizontal”**

Verschiebt ein Objekt mit oder ohne seine Nachfolger in horizontaler Richtung, abhängig von der vorhergehenden Option.

- **“Name Horizontal”**

Verschiebt den Objektnamen in horizontaler Richtung.

- **“Name Vertical”**

Verschiebt den Objektnamen in vertikaler Richtung.

- **“Set X Origin”**

Setzt die horizontale Koordinate des Anfangspunkts des Objekts auf den eingegebenen Wert.

- **“Set Y Origin”**

Setzt die vertikale Koordinate des Anfangspunkts des Objekts auf den eingegebenen Wert.

- **Scaling “Hydraulic Diameter”**

Verändert den Durchmesser des Objekts.

Die Optionen des Menüs für die Rods sind analog wie bei den TFO, daher werden nur die folgenden Abweichungen erklärt:

- **“Rod Horizontal”**

Verschiebt einen Rod in horizontaler Richtung. Falls zu dem Rod ein Control Rod zugeordnet ist, dann wird dieser auch mitverschoben.

- **“Rod Vertical”**

Verschiebt einen Rod in vertikaler Richtung. Falls zu dem Rod ein Control Rod zugeordnet ist, dann wird dieser auch mitverschoben.

- **“Cont.Rod Horiz.”**

Verschiebt nur den Control Rod in Horizontaler Richtung. Der zugeordnete Rod wird nicht mitverschoben

- **Scaling “Thickness Factor”**

Verändert die Breite des Rods.

4 Verwaltung, Verteilung und Archivierung von Daten

4.1 Konzept für die Verwaltung und den Zugriff der Simulationsdaten und der Benutzereingriffe

Die Simulation und die Visualisierung laufen als getrennte Prozesse (Anwendungsprogramme) auf einem oder verschiedenen Rechnern. In ATLAS besteht die grundsätzliche Anforderung, die aktuellen Simulationsdaten zu übertragen oder zurückliegende zur Verfügung zu stellen. Zusätzlich müssen interaktive Steuerbefehle (z. B. Verfahren von Ventilen) an das Simulationsprogramm weitergeleitet werden. Dabei müssen die Prozesse möglichst „synchron“ ablaufen, d. h. die Abweichungen sollten im Bereich von wenigen Simulationszeitschritten liegen.

Ausgangspunkt war der in ATLAS zur Verwaltung und zur Verteilung der Daten verwendete Datenserver („QSERVER“), der die benötigten Datenformate speichern kann und Zugriffe auf die Daten über Bibliotheksfunktionen ermöglicht. Diese Software war wegen ihrer grundlegenden Konzeption und der historisch gewachsenen Programmierung nicht mehr auf aktuelle Anforderungen erweiterbar. Als wichtigste Forderungen sind zu nennen:

- Berücksichtigung mehrerer Quellen für Simulations- und Bediendaten
- Verteilbarkeit der Prozesse auf verschiedene Rechner und Betriebssysteme
- Zeiteffizienter Zugriff bei hohen Datenraten und langen Zeitverläufen
- Optimierte Verwaltung der Datenformate in den Simulationscodes (ATHLET, COCOSYS, ASTEC)
- Einfache Erweiterbarkeit der verwendbaren Datenformate
- Weitgehender Einsatz von quellcodeoffener Software (Open Source)

4.1.1 Einsatz eines Datenbanksystems

Ein möglicher Ansatz zur Realisierung der Datenverwaltung ist die Verwendung eines Datenbankmanagementsystems (DBMS). Diese Systeme bieten eine Vielfalt von Funktionen zum Datenmanagement, ohne dass der Benutzer die Details der Speicherung und des Zugriffs kennen oder gar programmieren muss. Ebenfalls sind eine Reihe von

Open Source Produkten (MySQL, Firebird) oder kostenlose Express-Editionen kommerzieller DBMS (Oracle, DB2) verfügbar. Die vektoriellen Daten können in Tabellen (z. B. als „nested tables“ oder „vector arrays in Oracle) organisiert und gehalten werden, deren Einträge (Zeilen, Spalten) mit sogenannten Datenbankschlüsseln (Deskriptoren) gekennzeichnet sind. Mit diesem Konzept lassen sich einzelne Werte oder Gruppen von Simulationsergebnissen zu einem Zeitpunkt einfach und effizient abfragen. Wenn allerdings eine zeitabhängige Reihe eines Simulationswerts benötigt wird, muss ein Zugriff auf eine Spalte der Datenbanktabelle erfolgen.

In eine Studie /COH 05/ wurden grundsätzliche Untersuchungen zum Einsatz von DBMS für wissenschaftliche Daten durchgeführt. Die Performance von Oracle 10g bei Datenzugriffen wie Mittelwertberechnung oder auf Elemente zu diskreten Zeitpunkten im Vergleich zu rein dateibasierten Zugriffen wurde verglichen. Das Ergebnis war, dass insbesondere bei sehr großen Datensätzen (ca. 10^8 Einzelwerte, entsprechend 400MB bei 4Byte Realwert Daten), die Zugriffszeiten in der Datenbank bis zu 100 mal höher waren als beim einfachen Auswerten einer Datei. Das Datenaufkommen einer umfangreichen (z. B. mehrdimensionalen) Simulation kann noch erheblich über dem in der Studie untersuchten Wert liegen. Zusätzlich wurde in der Studie die benötigte Zeit zum Laden der Daten in das DBMS gemessen. Je nach Datenmodell ergeben sich für den betrachteten Datenumfang Zeiten zwischen 6 und 30 Minuten, erst danach sind alle Daten abfragbar. Man muss davon ausgehen, dass auch nicht getestete DBMS ähnliche Performance bei solchen Zugriffen aufwiesen.

Nimmt man zu den dargestellten Problemen noch den Installationsaufwand für die Datenbanksysteme hinzu, kann man feststellen, dass die verfügbaren relationalen DBMS, zumindest zum gegenwärtigen Zeitpunkt, nicht geeignet sind, die in ATLAS gestellten Anforderungen zu erfüllen.

4.1.2 Dateibasierter Ansatz

In verschiedenen anderen Entwicklungsprojekten wurden bereits gute Erfahrungen mit einem Datenaustausch von Anwendungsprozessen über eine Dateischnittstelle gemacht. Besonders vorteilhaft ist die einfache Gestaltung dieser Schnittstelle, da Sie keine eigenen Prozesse zur Erzeugung und zur Speicherung der Daten benötigt. In ATLAS ist ein Großteil der Anwendungen die Auswertung von bereits berechneten Simulationsdaten, die also in Form von Dateien vorliegen und daher prinzipiell nur geeig-

nete Zugriffsroutinen zum Lesen der gewünschten Daten benötigen. Diese Dateien werden bei den verwendeten Simulationsprogrammen durchwegs von FORTRAN-Programmen erzeugt, die die Simulationsergebnisse in die Dateien schreiben, aus Speicherplatzgründen allerdings in binärer Form und mit verschiedenen Ausgabeformaten. Die erzeugten Daten können außerdem von den verwendeten FORTRAN-Compilern, der Binärdarstellung von Gleitkommazahlen, der verwendeten Computer und dem Betriebssystem abhängen. Diese Probleme erschweren eine einheitliche Auswertung dieser Daten erheblich. So kann beispielsweise ein fehlerfreies Lesen von einer anderen Programmiersprache wie ANSI-C nicht sicher garantiert werden.

Die dargestellten Schwierigkeiten haben zur Entwicklung von Softwarebibliotheken geführt, die zum Ziel haben, extrem große Datenmengen, wie sie bei wissenschaftlichen Anwendungen (z. B. meteorologische Daten) anfallen, mit einheitlichen, plattformunabhängigen Schnittstellerroutinen zu schreiben und zu lesen. Als wichtigste Standardformate haben sich das Network Common Data Format (NetCDF) und das Hierarchical Data Format (HDF) etabliert.

NetCDF ist ein Dateiformat für den Austausch wissenschaftlicher Daten [WIK 09.1]. Es handelt sich um ein binäres Dateiformat, das durch die Angabe der Byte-Reihenfolge im Header maschinenunabhängig ist. NetCDF ist ein offener Standard; das Projekt wird von der University Corporation for Atmospheric Research (UCAR) betreut. Das Dateiformat ist selbstbeschreibend – es gibt einen Header, in dem neben Metadaten (in Form von geordneten Paaren aus Schlüsseln und Attributen) auch die Struktur des Datenbereichs beschrieben ist. Die grundsätzlichen Kernfunktionen sind in C implementiert, es gibt jedoch Schnittstellen zu C++, Fortran4, Fortran90, Perl, Java, Python, Ruby, Matlab, IDL und Octave. Zusätzlich gibt es eine Reihe von Auswertungs- und Visualisierungsprogrammen, die dieses Dateiformat direkt unterstützen.

HDF ist ein Datenformat zur effizienten und flexiblen Speicherung von Messdaten, Zwischen- und Endergebnissen, insbesondere für große Datenmengen [WIK 09.2]. HDF wurde vom National Center for Supercomputing Applications entwickelt. HDF ist sehr flexibel und unterstützt viele verbreitete Datenstrukturen (z. B. multidimensionale Arrays, Raster Image, Tabellen mit Texten und Daten). Zusätzlich werden Annotationen (Metadaten) sehr gut unterstützt, die jeden Datensatz sowie jedes einzelne Element mit einer Beschreibung versehen können. Die neuere Version, HDF5, führt eine objektorientierte Sicht auf die Daten ein, in denen Gruppen von Daten, mit definierten Eigenschaften, zusammengefasst werden können.

Aus programmtechnischer Sicht ist die Nutzung von NetCDF wegen der eingeschränkten Funktionsumfangs und der übersichtlichen Programmierschnittstellen (API) wesentlich einfacher zu implementieren als dies bei HDF der Fall wäre. Eine erste grundsätzliche Prüfung der verfügbaren Datentypen und Zugriffsmechanismen ergab, dass NetCDF für die in ATLAS auftretenden Fälle ausreichend ist. Zusätzlich nutzt die Weiterentwicklung NetCDF4 bei der internen Speicherung der Daten HDF5, ist also in gewisser Hinsicht zu diesem kompatibel.

4.1.3 Grundsätzliche Untersuchungen zum Dateiformat NetCDF

Die Durchführung der Untersuchungen hatte zum Ziel, die grundsätzliche Eignung des NetCDF-Dateiformats für die in ATLAS auftretenden Aufgaben und Randbedingungen zu prüfen. Zu diesem Zweck wurde eine stark vereinfachte Testumgebung entwickelt, in der die Speicherung und der Zugriff auf Ergebnis- und Steuerdaten, ähnlich wie in einer typischen ATHLET-ATLAS-Simulation, abgebildet wurde. Für die benötigten Datenformate, insbesondere Vektoren reeller Zahlen und ASCII-Texte, wurden Schreib- und Leseroutinen in verschiedenen Varianten basierend auf den Bibliotheksfunktionen von NetCDF erstellt. Mit diesen wurden die für ATLAS wichtige Funktionalität wie z. B. ein gleichzeitiger Zugriff auf Simulationsdaten von mehreren Prozessen oder die Geschwindigkeit beim Anzeigen großer Datenmengen getestet. Zusätzlich wurden systematisch alle bisher in ATLAS verwendeten Funktionen für die Datenkommunikation (Initialisierung, Synchronisierung, Lese- und Schreibfunktionen) erfasst.

In der Testumgebung konnte gezeigt werden, dass der gleichzeitige Zugriff auf eine Datei im NetCDF-Dateiformat mit einem schreibenden und beliebig vielen lesenden Prozessen erfolgen kann. Insbesondere bietet die Funktion „nf_sync“ aus NetCDF die Möglichkeit eine konsistente Sicht aller Prozesse auf die Daten sicherzustellen, also die Verfügbarkeit der jeweils aktuellsten Daten. Eine Synchronisierung der Prozesse und der Zugriff auf Daten zu beliebigen Simulationszeiten konnten mit geeigneten Mechanismen unter Nutzung elementarer NetCDF-Bibliotheksaufrufe ebenfalls erreicht werden.

Das NetCDF-Format ist bezüglich der Zugriffsgeschwindigkeit besonders effizient, wenn auf Vektordaten zugegriffen wird, die sequentiell im Speicher bzw. in der Datei abgelegt sind. Zur Behandlung von zeitabhängigen Daten besitzt NetCDF eine Dimension („Record Dimension“), die beliebig wachsen kann. Zeitmessungen haben ergeben,

dass sich Lesezeit für einen einzelnen Skalarwert nur wenig von dem des ganzen Ergebnisvektors zu einem Zeitpunkt unterscheidet, also hauptsächlich von der Positionierungszeit auf den gesuchten Zeitpunkt (Record) abhängt. Aus den Messungen ist ebenfalls ersichtlich, dass die Lesezeit für einen Zeitverlauf wesentlich von der Anzahl der benötigten Records abhängt.

Die Geschwindigkeiten beim Datenzugriff auf eine Netzwerkplatte unterscheiden sich bei den heute verfügbaren Netzwerkgeschwindigkeiten (LAN) nur unwesentlich gegenüber lokalen Laufwerken, so dass auch eine Verteilung der Prozesse auf verschiedene Rechner ohne weitere Probleme möglich ist. Für den Zugriff auf Dateien im Netzwerk („Remote File Access“) haben sich eine Reihe von Standardwerkzeugen etabliert. In rein WINDOWS basierten Rechnersystemen ist bei einem eingerichteten Netzwerk (z. B. als Domäne) lediglich eine Freigabe der gewünschten Verzeichnisse erforderlich. Sollen zusätzlich UNIX oder LINUX-basierte Systeme integriert werden, bietet SAMBA dazu eine einfach handhabbare Möglichkeit. SAMBA ist in der Lage Dateidienste eines Windows-Servers zu übernehmen. Der Aufwand für Installation und Administration ist relativ gering und meist in heterogenen Netzen bereits in Betrieb. Im Bereich des Internets (Wide Area Network) bietet sich mit WEBDAV („Web-based Distributed Authoring and Versioning“), ein offener Standard zur Bereitstellung von Dateien im Internet, eine weitere Möglichkeit, auf entfernte Dateien zuzugreifen. Der Zugriff erfolgt über das Protokoll HTTP und Port 80, die bei Firewalls in der Regel nicht blockiert werden. Der Aufwand für Einrichtung und Rechtevergabe ist ähnlich gering wie bei SAMBA.

Die Zugriffsgeschwindigkeit auf extrem lange Zeitverläufe einzelner Größen liegt im Bereich der bisher verwendeten Lösung, kann also im Einzelfall relativ langsam sein. Das Problem kann aber durch eine Reduktion der genutzten Punktdichte erheblich vermindert werden.

Zur Steuerung der Simulation wird nur eine Datei genutzt, in die, bei Bedarf auch von mehrere Steuerprogrammen, sequentiell die Steuerbefehle eintragen werden. Damit kann diese Datei auch die Logbuchfunktion für alle Simulationseingriffe übernehmen.

Die Tests haben somit ergeben, dass die für die speziellen Anforderungen in ATLAS notwendige Funktionalität mit NetCDF abgebildet werden kann. Besonders vorteilhaft erscheint die einfache Erweiterbarkeit der Dateien durch Speicherung von Zusatzinformationen oder der möglichen Datenformate für die Ergebnisdaten. Diese ist theoretisch sogar zur Laufzeit der Prozesse möglich, beispielweise bei der Aktivierung

zusätzlicher Simulationsmodelle. Diese erzeugen zusätzliche Ergebnisdaten, die durch eine Neudefinition der NetCDF-Datei berücksichtigt werden können. Allerdings wird die Datei zur Laufzeit neu erzeugt (kopiert), wenn die Dimension der Daten die zu Beginn definierte überschreitet, wodurch im Einzelfall ein erhöhter Zeitaufwand entstehen kann.

4.2 Implementierung der in ATLAS benötigten Funktionen

Die benötigten Funktionen im Visualisierungsmodul von ATLAS bestehen im Zugriff auf eine Beschreibungsliste (sog. „Keyword-Liste“) mit den Namen der Simulationsvariablen und auf alle Datenwerte dieser Variablen zu den diskreten Simulationszeitpunkten. Zusätzlich müssen zur interaktiven Steuerung des Simulationsablaufs („Online Modus“) Benutzerbefehle an die Simulationsmodelle gesendet werden. Diese Modelle benötigen im Gegenzug Funktionen zum Übertragen der Ergebnisdaten und zum Empfangen der Steuerdaten.

Die benötigten Informationen sollen in Dateien mit dem NetCDF-Dateiformat bereitgestellt werden. Daher wurden Schnittstellenprogramme entwickelt, die solche Dateien anlegen oder öffnen können und die benötigten Datenstrukturen definieren oder auslesen können. Weitere Programme sind für das Ein- und Auslesen der Beschreibungsliste der Variablen, die Zeitvektoren der Simulationsergebnisse und die Steuerbefehle in diese Dateien entwickelt worden. Diese Programme können sowohl zur Laufzeit der Simulation als auch durch eine Konversion der bestehenden Ergebnisdateien aus vorliegenden Simulationsläufen verwendet werden. Die Programme wurden mit den Bibliotheksfunktionen von NetCDF in FORTRAN90 erstellt und ausführlich im Quellcode dokumentiert. Die Programme werden von verschiedenen Anwendungen, die auf unterschiedlichen Programmiersprachen (C++, FORTRAN, VB.NET) basieren, genutzt. Da sich bei diesen die Parameterübergabe, Namenskonventionen (siehe Anhang 4) und die verwendeten Laufzeitbibliotheken unterscheiden, werden aus den erstellten Schnittstellenprogrammen verschiedene, für die jeweilige Anwendung geeignete Programmbibliotheken generiert. Dies wird durch entsprechende Anpassungen der Compiler- und Linkeroptionen erreicht.

In der folgenden Tabelle sind die wesentlichen Grundfunktionen in der Reihenfolge des notwendigen Ablaufs beschrieben wie sie in den Prozessmodellen und im Monitor ver-

wendet werden. Details zu allen verfügbaren Routinen können dem Anhang 5 entnommen werden.

Routine	Funktion / Erklärung
Prozessmodell	
nc_put_keywords	Erstellen einer neuen NetCDF-Ergebnisdatei, Definition der benötigten Variablen und Dimensionen und speichern der Keyword-Informationen (Namen, Einheit, Typ) in dieser Datei
nc_send_complete_gvector	Speichern eines Ergebnisvektors zu einem Zeitpunkt auf der NetCDF-Ergebnisdatei
nc_get_message	Lesen eines Benutzerkommandos von der NetCDF-Steuerdatei
Visualisierung (Monitor)	
nc_get_keywords	Öffnen einer vorhandenen NetCDF-Ergebnisdatei und lesen der enthaltenen Keyword-Informationen
nc_get_gvalues	Lesen einer Liste von ausgewählten Ergebniswerten zu einem Zeitpunkt
nc_get_trend_values	Lesen einer Liste von ausgewählten Zeitkurven für ein Zeitfenster
nc_send_message	Speichern eines Benutzerkommandos in einer NetCDF-Steuerdatei (inklusive eines Echtzeitstempels)

Tab. 4-1 Basisfunktionen für die Datenschnittstelle zwischen Simulation und Visualisierung

4.3 Senden von Nachrichten an die Prozessmodelle

Zur Steuerung der Simulation im „online“-Modus (z. B. „Pause“ oder interaktive Eingriffe „Ventilöffnen“) werden Nachrichten vom Grafikmodul an das Prozessmodell verschickt. Diese liegen in einem definierten Format vor und werden als Zeichenkette („String“) der Länge 255 kodiert. Zusätzlich erhält jede Nachricht einen Echtzeitstempel („mestime“), also den Zeitpunkt, zu dem die Nachricht generiert wurde. Der Zeitpunkt wird als ein Vektor von 8 Integer-Werten gespeichert, die die folgende Bedeutung haben:

1: Jahr, z. B. 1993. 2: Monat. 3: Monatstag. 4: Zeitdifferenz zur Weltzeit in Minuten. 5: Stunde des Tages. 6: Minute innerhalb der Stunde. 7: Sekunden 8. Millisekunden.

Die Nachrichten und Zeiten werden in eine Steuerdatei geschrieben und aus dieser Datei vom Prozessmodell gelesen. Da in ATLAS i.a. mehrere Quellen (gleichzeitig aktive Monitorprozesse) für die Steuerung eines Simulationsmodells möglich sind, wurde ein spezielles Verfahren entwickelt, das die Nutzung einer einzigen, gemeinsamen Datei zum Übertragen der Steuernachrichten ermöglicht, ohne dass Nachrichten überschrieben werden können und damit verloren gehen. Beim Eintrag neuer Nachrichten wird die Datei geöffnet und in die Datei wird ein Statuseintrag (filestatus = „Locked“) geschrieben, aus dem erkennbar ist, dass eine Quelle gerade neue Nachrichten in die Datei schreibt. In diesem Fall wird das Schreiben für andere Quellen zeitweilig blockiert. Nach Beendigung des Schreibvorgangs wird der Statuseintrag auf „New Message“ umgesetzt und die Datei geschlossen.

Das Prozessmodell prüft nach jedem Zeitschritt die Steuerdatei auf neue Einträge. Die NetCDF-Bibliothek stellt Funktionen zur Synchronisierung bereit, die den Zugriff auf die jeweils aktuellsten Einträge in der Datei sicherstellen.

Die Steuerdatei wird standardmäßig mit dem Namen „messages.ncc“ im Arbeitsverzeichnis der ATLAS-Anwendung erstellt. Falls das Grafikprogramm und das Prozessmodell auf verschiedenen Rechnern ausgeführt werden, muss das Arbeitsverzeichnis auf einem Netzwerkpfad liegen, auf das diese Rechner und der Benutzer lesend und schreibend zugreifen können.

4.4 Anpassung der Startoberfläche und Startprozeduren

In der ursprünglichen Version von ATLAS wurde zur Datenverwaltung und zum Datenaustausch der „QSERVER“ verwendet. Dieses Programm musste jeweils zu Beginn einer ATLAS-Anwendung gestartet werden. Als Quelle für die Simulationsdaten kann entweder ein laufendes Simulationsprogramm oder eine vorliegende Datei mit Simulationsergebnissen dienen. Das Laden der gespeicherten Ergebnisse in den Datenserver wurde mit dem Hilfsprogramm „QFILL“ durchgeführt. Die Vorgabe von Parametern erfolgt über die Startoberfläche von ATLAS, die Programme selbst werden, wie auch alle anderen in ATLAS verwendeten Programme, über die Skriptdatei „qserver.js“ mit Hilfe des „WINDOWS Script Host“ gestartet /BOR 02/.

In der neuen Version wird für die Bereitstellung der Simulationsdaten eine NetCDF-Ergebnisdatei verwendet. Somit werden die oben genannten Programme nicht mehr

benötigt. Wenn das Simulationsprogramm direkt von ATLAS gestartet wurde, liegt eine Datei in diesem Format normalerweise vor, d. h. sie wurde zur Laufzeit bereits erzeugt. Im Fall von ATHLET befindet sich diese Datei im Ausgabeverzeichnis bzw. Arbeitsverzeichnis von ATHLET und besitzt, entsprechend den Namenskonventionen von ATHLET, die Bezeichnung „Problem_Id.Run_Id.ncf“.

Im Fall einer Simulation ohne ATLAS muss diese Datei nachträglich erzeugt werden. Dies geschieht automatisch durch die Ausführung des Programms „NCFConvertFiles.exe“. Dieses konvertiert dann die in der Startoberfläche (Abb. 4-1) angegebenen Ergebnisdateien in das NetCDF-Format. Das Programm kann alle in ATLAS unterstützten Binärformate (ATHLET, ASTEC, COCOSYS, MELCOR, S-RELAP) in das Zielformat konvertieren. Zum Starten des Programms wird das Skript „ncfiles.js“ benutzt. Dieses Skript erkennt, ob bereits eine konvertierte Datei (neueres Änderungsdatum, gleicher Namensstamm wie Simulationsdaten) existiert. In diesem Fall wird keine Neukonvertierung durchgeführt, um die Plattenzugriffe (IO-Belastung) zu reduzieren. Der Start und die Anzeige von Daten mit dem Monitor können bereits während einer laufenden Konvertierung erfolgen, da diese im Fall von sehr großen Simulationsdateien im Einzelfall längere Zeit andauern kann. Die erfolgreiche Erstellung der NetCDF- Ergebnisdatei wird beim Programmende mit einem Hinweis auf die größte Simulationszeit und die Anzahl der Zeitpunkte (Records) signalisiert.

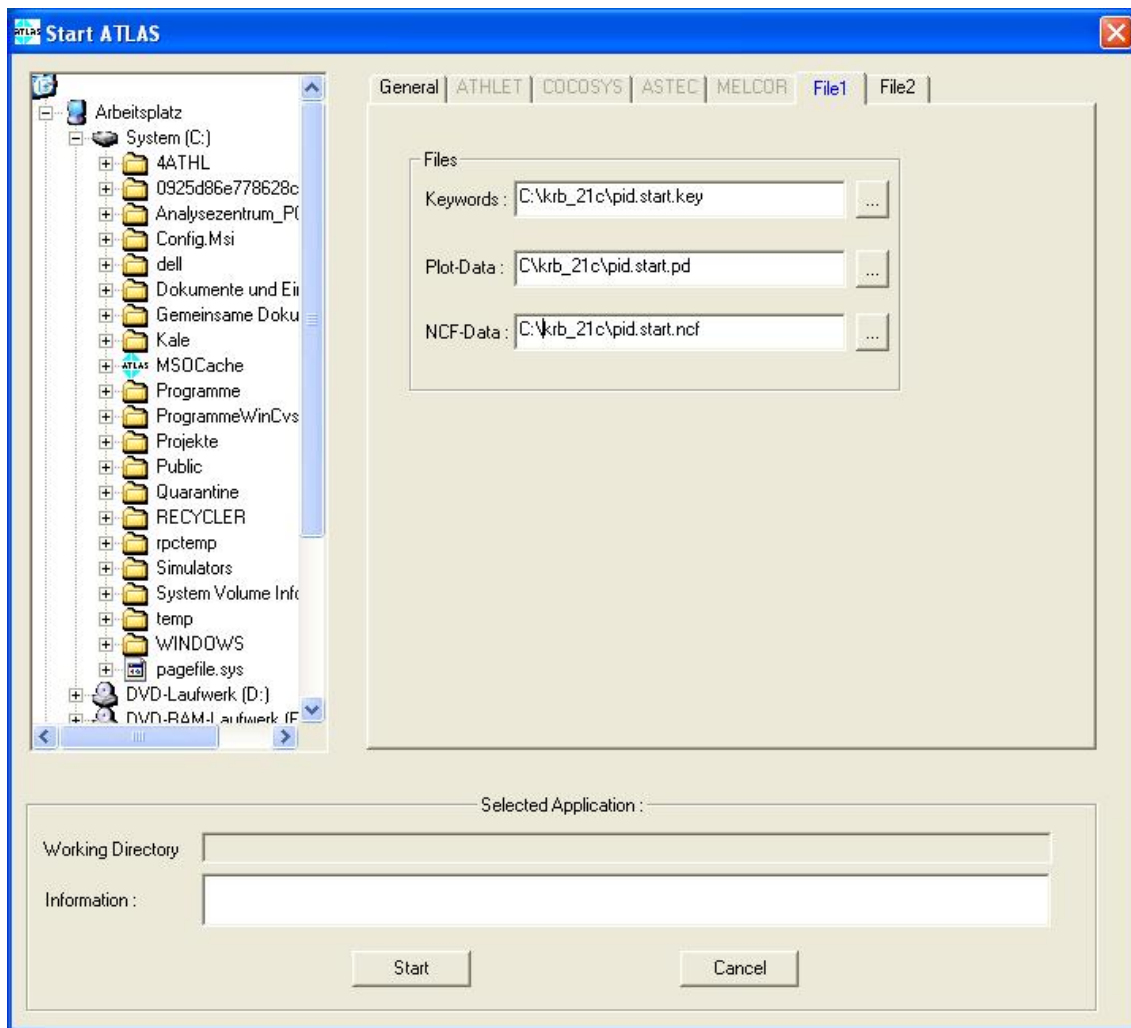


Abb. 4-1 Startoberfläche von ATLAS – File Dialog

4.5 Interaktives Tool zur schnellen Auswertung von NetCDF-Dateien

Es wurde ein interaktives Tool für WINDOWS realisiert, mit dem bestehende Simulationsergebnisse für alle in ATLAS vorhandenen Modelle in das neue Dateiformat überführt werden können. Aus der erzeugten Datei können dann interaktiv Namen von Variablen, Einzelwerte und Tabellen von Zeitverläufen abgefragt und angezeigt werden. Der Analytiker kann damit schnell und effizient, unabhängig vom verwendeten Simulationsmodell, Zahlenwerte und Zeittabellen der Simulationsvariablen erzeugen und überprüfen.

Die in Abb. 4-2 dargestellte Benutzeroberfläche enthält im oberen Teil Eingabefelder für die Ergebnisdatei, Keyworddatei und eine Auswahlmöglichkeit des Binärdatentyps entsprechend des Simulationsmodells. Aus diesen Vorgaben kann dann mit dem Knopf „Convert“ die darunter angegebene NetCDF-Datei erzeugt werden. Alternativ kann in diesem Feld auch eine bestehende Datei eingetragen werden. Mit der Betätigung des Knopfs „Open“ werden dann die Felder mit der Anzahl der Records (Zeitpunkte), Anzahl der Variablen und erster sowie letzter Zeitpunkt auf der Datei angezeigt. Anschließend kann ein einzelner Wert zu einem Zeitpunkt abgefragt werden („Get Value“), wobei dann ebenfalls die zugehörige Bezeichnung („Keywords“) eingeblendet werden. Zuletzt besteht die Möglichkeit, für einen Adressbereich von Variablen eine Reihe von Records anzuzeigen („Get Vector“).

Mithilfe der Bedienoberfläche konnte gezeigt werden, dass die neu entwickelten Programme ihre Funktionen korrekt erfüllen und die Daten im neuen Dateiformat exakt die vorliegenden Ergebnisse ohne Unterschiede zum ursprünglichen Format wiedergeben.

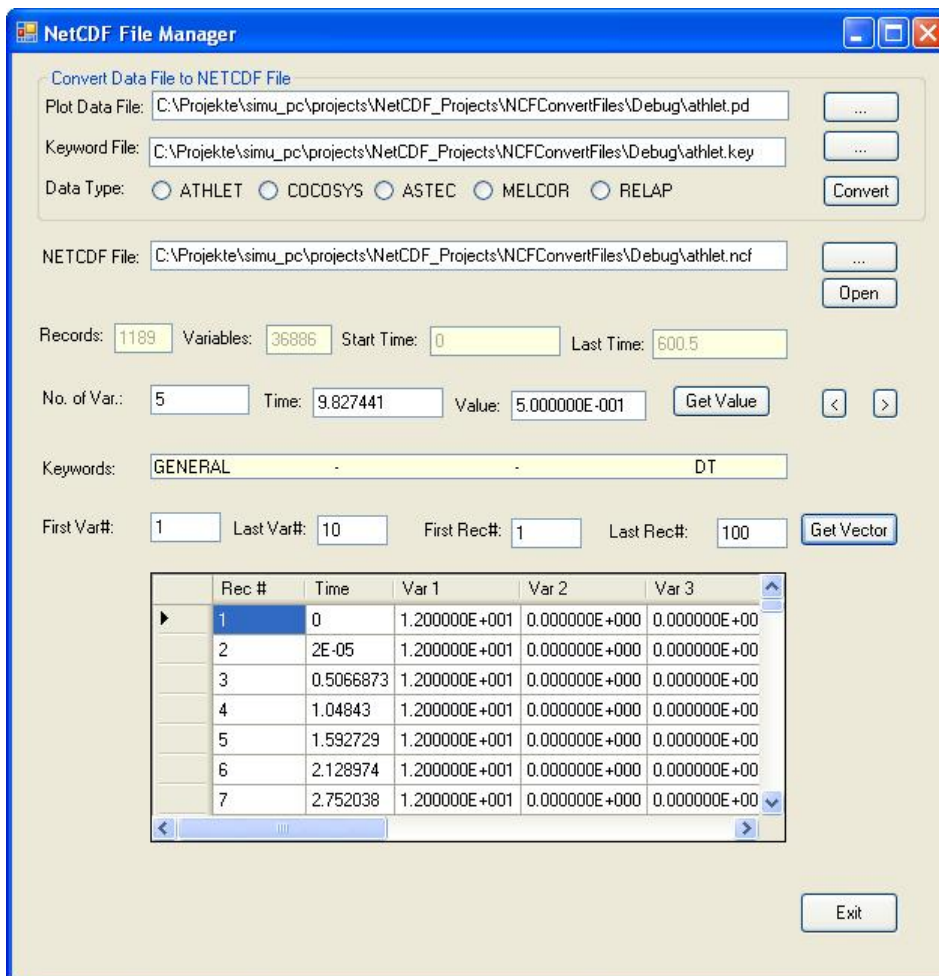


Abb. 4-2 Bedienoberfläche zur Auswertung von NetCDF-Dateien

4.6 Erreichte Vorteile gegenüber der Server basierten Version

Der wichtigste Vorteil besteht sicherlich darin, dass mit der Datenschnittstelle eine gut dokumentierte Lösung mit vollständiger Verfügbarkeit des Quellcodes realisiert werden konnte. Das NetCDF-Format ist für wissenschaftliche Daten weit verbreitet und wird ständig weiterentwickelt. Diese Binärdaten können unabhängig von dem benutzten Betriebssystem und der verwendeten Hardware ausgetauscht und von verschiedenen Programmiersprachen gelesen werden. Dies lässt erwarten, dass auch auf lange Sicht die Daten sicher lesbar sind, was insbesondere bei Messdaten von elementarer Bedeutung ist.

Die Datei ist selbst beschreibend, d. h. das Format der Daten ist in der Datei gespeichert und das auswertende Programm braucht daher keine Informationen über die Art, Länge und Formate der gespeicherten Daten. Neben ATLAS gibt es eine große Anzahl weiterer Programme zur Datenauswertung und Visualisierung (z. B. ESRI, IDL, MATLAB, PV-WAVE), die dieses Datenformat direkt unterstützen.

Für die Nutzung in ATLAS ist die relativ einfache Erweiterbarkeit der gespeicherten Daten, z. B. das Hinzufügen von Metadaten wie Messstellen, Skalierungsfaktoren etc., besonders interessant. Dies kann sogar bei bereits bestehenden NetCDF-Dateien, also beispielweise auch während einer laufenden Simulation, geschehen, indem die Zusatzinformation einfach durch die Definition und das Schreiben der entsprechenden Daten ergänzt wird. Ebenso ist es prinzipiell möglich, in der Visualisierung mehrere Ergebnisdateien zu benutzen und damit beispielweise eine Simulation mit Messdaten zu vergleichen.

Die Ergebnisdarstellung wurde mit der bisherigen QSERVER-Schnittstelle und externen Visualisierungstools verglichen. Gegenüber der alten Lösung konnte eine Verbesserung bei der Zeitkurvendarstellung, d. h. eine geringere künstliche Glättung, erreicht werden. Dazu wurde zusätzlich das Verfahren zur Verwaltung der Trenddaten im Speicher verbessert (Anhang 1: Datenmanagement bei den Trendkurven in ATLAS). Hinsichtlich der Geschwindigkeit ist die neue Lösung vergleichbar, bei großen Zeitfenstern und vielen Datenpunkten deutlich schneller. Durch den Wegfall des QSERVER-Programms nutzt ATLAS einen Prozess weniger als bisher. Dies ergibt weitere Vorteile bei der Fehleranfälligkeit und Robustheit des Gesamtsystems. Außerdem entfällt die bisher notwendige Installation des „ONC-RPC-Dienstes“ und die Aktivierung des Dienstes. Dies erleichtert die Gesamtinstallation von ATLAS für die Benutzer erheblich.

4.7 Archivierung von Versuchs- und Simulationsdaten

4.7.1 Anforderungen

Hauptzielsetzung war die Bereitstellung eines Systems, das die langfristige Speicherung von Simulationsergebnissen und Versuchsdaten gewährleistet und die zukünftige Verfügbarkeit in einfacher und übersichtlicher Weise garantiert. Bei der Auswahl des Systems war zu beachten, dass es die folgenden Anforderungen erfüllt:

- Ausreichendes Speichervolumen für große Datenmengen
- Speicherung und Zugriff auf Zusatzinformationen zur Simulation
- Geeignete Benutzeroberflächen für Einfügen neuer Informationen, Zugriff auf die Daten mit Suchabfragen oder hierarchischer Navigation
- Angemessener Realisierungsaufwand bei hoher Funktionalität

4.7.2 Realisierung

Microsoft Office SharePoint Server (MOSS), eine Portal-Softwarelösung von Microsoft, wird in der GRS seit geraumer Zeit zur Organisation und Speicherung wichtiger Daten und Dokumente verwendet. Da mit der Software auch alle Anforderungen erfüllt werden konnten, wurde die Verwaltung von Versuchs- und Simulationsdaten auf dieser Basis realisiert. Die Wahl von Sharepoint im Vergleich zu anderen DBMS (z. B. ORACLE) war in den guten Erfahrungen bei andern Portalanwendungen in der GRS begründet und die Verfügbarkeit von standardisierten Benutzerschnittstellen, die keinen hohen Programmieraufwand erfordern.

Hauptkomponenten von SharePoint sind die SQL-Datenbank, das Webinterface, und spezielle Dienste. In der Datenbank werden Metadaten, wie beispielsweise Anlage, Modellierung, Störfalltyp und Referenzen auf den Speicherort der Nutzdaten, abgelegt werden können. Diese Daten können im Webinterface über speziell entwickelte Eingabeformulare eingetragen oder angezeigt werden. Die SharePoint-Dienste bieten u.a. die Möglichkeit, thematisch getrennte Bibliotheken zur Speicherung von Dokumenten anzulegen und zu verwalten. Damit ist es möglich, die Metadaten in Listenform nach beliebigen Sortier- und Filterkriterien in eine Listenform, ähnlich wie im WINDOWS Da-

tei-Explorer anzuzeigen. Gezielte Volltextsuche mit logischen Begriffsverknüpfungen wird über Suchmasken ebenfalls unterstützt.

Die eigentlichen Nutzdaten, wie Rechenergebnisse, Datensätze, Geometriedaten werden getrennt auf einem speziellen Fileserver des Herstellers NetApp gespeichert, da Sharepoint für sehr große Dateien nicht optimal geeignet ist. Von besonderer Bedeutung ist dabei die Möglichkeit, ein virtuelles Laufwerk unabhängig von physikalischen Gegebenheiten zu definieren. Der Zugriff auf dieses Laufwerk erfolgt über einen virtuellen Hostnamen, dessen freigegebene Verzeichnisse sowohl unter UNIX und WINDOWS mittels NFS und SAMBA ansprechbar sind. Die Geschwindigkeit des Datenzugriffs in der Visualisierung mit ATLAS wurde mit lokaler Speicherung verglichen. Bei durchschnittlicher Auslastung von Netz und Server ergeben sich etwa die doppelten Zugriffszeiten auf dem Fileserver. Die Datensicherung erfolgt in regelmäßigen Zeitabständen mit dem Magnetband basierten Backup System „Tivoli Storage Manager“ von IBM.

4.7.2.1 Datenbibliotheken

Die Simulationsergebnisse der GRS können aus verschiedenen Programmen stammen. Die Ergebnisdaten, die mit ATLAS visualisiert werden, können aktuell durch Simulationen mit ATHLET, COCOSYS, ASTEC oder MELCOR generiert werden. Auch Ergebnisse aus anderen Thermofluidprogrammen können mit ATLAS in Postprozessläufen visualisiert werden, wenn ihr Format dem vom ATLAS angepasst ist. Aus diesem Grund ist es angebracht für jeden Analysencode eine separate Dokumentbibliothek anzulegen. Die Namen der Bibliotheken sollen auch die gleichen wie die der Programme sein. Das heißt, dass die Simulationsergebnisse aus ATHLET-Läufen z. B., über ein Hyperlink mit der Bezeichnung „ATHLET“ erreicht werden sollen. In ähnlicher Form werden Messdaten organisiert, also z. B. nach Versuchstand und Versuchsbezeichnung geordnet.

4.7.2.2 Definition der Metainformationen

Das wichtigste Merkmal der internen Organisation einer Datenbibliothek sind ihre Metainformationen, denn im Allgemeinen lassen sich diese Daten nur durch eine zweckmäßige Auswahl von thematischen Eigenschaften auch leicht identifizieren. In unserem Fall wollen wir die Daten nicht in Form einzelner Dateien ablegen, sondern für den Si-

mulationslauf zusammengehörige und relevanten, wie z. B. Textausgabe und Grafikdaten. Voraussetzung dafür ist es deswegen, dass alle Dateien unter einem einzigen Verzeichnis vorliegen. Die Entscheidung darüber, welche Dateien eines bestimmten Simulationslaufs durch SharePoint langfristig gespeichert werden, wird dem Anwender überlassen werden. Es werden allerdings Richtlinien für Kontrollen festgelegt, die dem Anwender dabei helfen, eine vollständige Auswahl von Dateien bei der Speicherung von Simulationsergebnissen zu treffen.

Im Allgemeinen kann man die Files, die zur Speicherung der Simulationsergebnisse benötigt werden in drei Kategorien unterscheiden. Zur ersten Kategorie gehören die Daten, die für die Auswertung der Ergebnisse unbedingt notwendig sind. Im Bezug auf ATHLET-Läufen sind sie: das die Plot- und Schlüsselwortdatei (*.pd und *.key Files).

In Abstimmung mit erfahrenen Störfallanalytikern wurde eine erste Liste von Metainformationen festgelegt. Diese Auswahl ist zweckmäßig für das Wiederfinden der Simulationsergebnissen sein und auch so konzipiert, dass sie dem Analytiker die Aufgabe der Archivierung der Ergebnisdaten erleichtert.

Die Liste der Metainformationen umfasst gegenwärtig folgende Deskriptoren und kann bei Bedarf leicht ergänzt werden:

- Analysename (z. B. „Analyse des Notstromfalls mit Athlet für den EPR“)
- Anlage (z. B. „KWBA“)
- Störfallgruppe (z. B. „SE 3 Abnahme Reaktorkühlmittelinventar“)
- Störfalltyp (z. B. „E3-11“)
- Analysecode (z. B. „ATHLET 1.2A“)
- Ziel der Analyse (Beschreibung über mehrere Zeile)
- Modellierung (Beschreibung über mehrere Zeile)
- Ergebnisse (Beschreibung über mehrere Zeile)
- Datenverzeichnis (Link zum Datenverzeichnis auf dem Fileserver)
- Datenfile (Link zum Datenfile auf dem Fileserver z. B. *.pd File)
- Graphikfile(Link zum Graphikfile auf dem Fileserver z. B. *.gr File)

- Datum
- Analyst
- Dokumentation (z. B. „GRS-A-2795“. Beschreibung über mehrere Zeilen)
- Literatur (Beschreibung über mehrere Zeilen)
- Sonstiges (Beschreibung über mehrere Zeilen)

Die Metainformationen unter den Deskriptoren „Ziel der Analyse“, „Modellierung“, „Ergebnisse“, Dokumentation, Literatur und Sonstiges, können in mehreren Zeilen eingegeben werden, sind aber nicht unbedingt erforderlich. Es wird darauf hingewiesen, dass umfangreiche Zusatzinformation den Wert der Daten für andere erheblich erhöhen kann. Andererseits sollte die Anforderungen für den Analytiker zum Beschreiben der Daten auch nicht zu hoch sein, um das freiwillige Einstellen der Daten nicht mit hohem Zusatzaufwand zu behindern.

Alle anderen Angaben mit einer einzelnen Zeile sind obligatorisch. Unter dem Deskriptor „Datenverzeichnis“ soll ein Hyperlink auf dem Verzeichnis eingegeben werden, wo die Dateien gespeichert werden, die durch den Simulationslauf erzeugt wurden.

4.7.2.3 Eingabe der Metainformationen

Die Eingabe der Metainformationen erfolgt über ein durch InfoPath entworfenes Formular (Abb. 4-2). InfoPath ist ein WINDOWS-Anwendungsprogramm, das zum Office-Paket von Microsoft gehört. Es ermöglicht sowohl die Gestaltung von Formularen als auch deren späteres Ausfüllen mit Eingabedaten. InfoPath kann leicht in Verbindung mit SharePoint verwendet werden. Durch die Veröffentlichung von Formularvorlagen auf dem SharePoint-Server können neue Dokumentbibliotheken angelegt und mittels des entworfenen Formulars für die Eingabe der Daten direkt verwendet werden. Die Eingabe der Daten kann durch Verwendung geeigneter Steuerelemente wie Dropdown-Listen und Datenauswahlfeldern erheblich erleichtert werden.

The screenshot shows a web browser window with a form titled "Simulationsergebnisse". The form contains the following fields and controls:

- AnalyseName:** A text input field with a red asterisk on the right.
- Anlage:** A dropdown menu with "GKN2" selected.
- StörfallGruppe:** A dropdown menu with "SE 2 Abnahme Reaktorkühlmittelinventar" selected.
- StörfallTyp:** A text input field with "SE-2" entered.
- AnalyseCode:** A text input field with "ATHLET" entered.
- Ziel der Analyse:** A large empty text area.
- Modellierung:** A large empty text area.
- Ergebnisse:** A large empty text area.
- Datenverzeichnis:** A text input field containing "\grsvrd\versuchsdaten\simulation".
- DatenFile:** A text input field containing "\grsvrd\versuchsdaten\simulation\sbloca.test.pd".
- GraphikFile:** A text input field containing "\grsvrd\versuchsdaten\simulation\sbloca.test.gr".
- Datum:** A date input field with a calendar icon, showing "11.08.2010".
- Analyst:** A dropdown menu with "Pointner (POI)" selected.
- Dokumentation:** A large empty text area.
- Literatur:** A large empty text area.
- Sonstiges:** A large empty text area.

The browser's address bar and menu bar are visible at the top and bottom of the window.

Abb. 4-2 Formular mit den Eingabefeldern für die Metainformationen für Daten

Abb. 4-2 zeigt den Inhalt der Dropdown-Listen der Anlagen bzw. der Störfallgruppen. Die Eingabe über diese Listen ist nicht nur komfortabel, sondern es können dadurch auch Tippfehler vermieden werden.

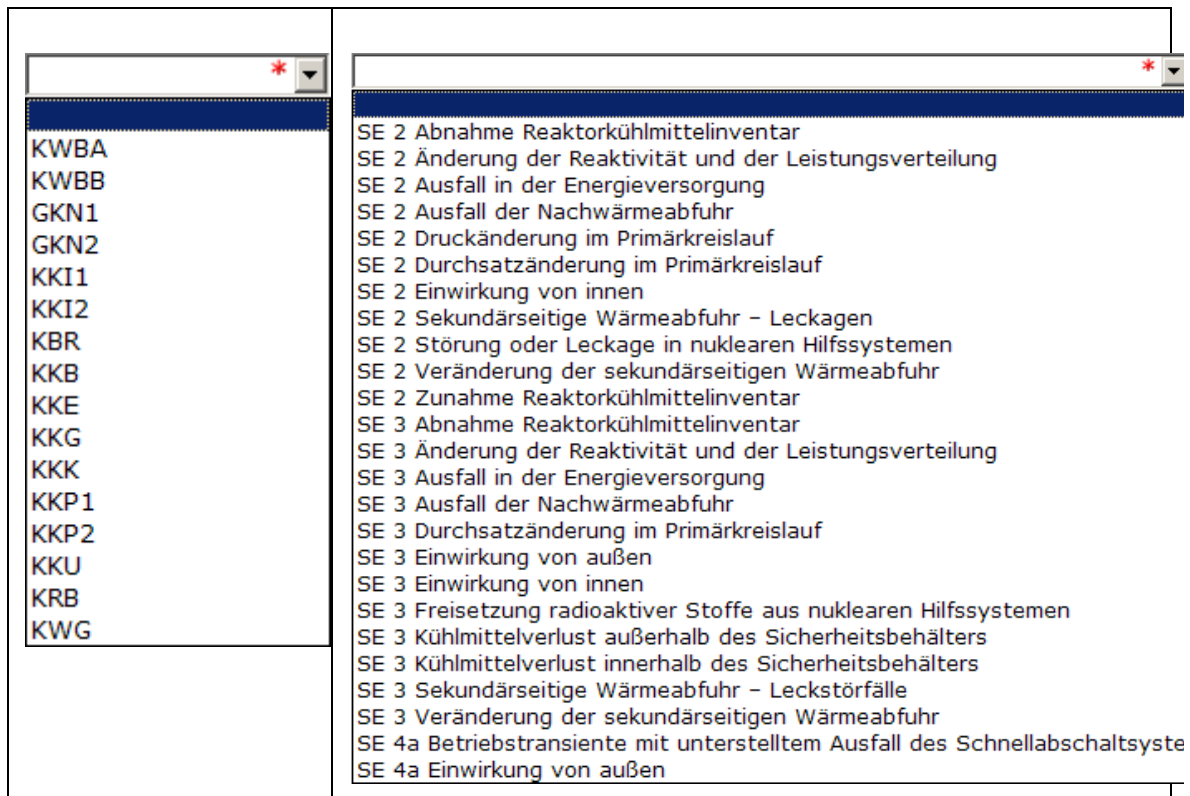


Abb. 4-3 Dropdown-Liste der Anlagen und der Störfallgruppen

Die erste Liste umfasst alle deutschen Anlagen, die zweite Liste umfasst unter den Kennzeichen SE-2, SE-3 und SE-4a die Störfallgruppen „Anomaler Betrieb“, „Störfälle“ und „Sehr seltene Ereignisse“, die dritte Liste unter der Bezeichnung „Analyst“ die Namen aller Mitarbeiter, die zur Speicherung der Simulationsläufe berechtigt sind. Alle Dropdown-Listen lassen sich um weitere Begriffe beliebig erweitern.

4.7.3 Bedienung von SharePoint-Listen

SharePoint zeigt die Metainformationen standardmäßig in tabellarischer Form als Listen von Einträgen an. Abb. 4-4 zeigt eine SharePoint-Liste mit Metainformationen von Simulationsergebnissen, die mit Hilfe des oben beschriebenen InfoPath-Formulars in das GRS-Portal eingegeben worden sind. Wenn man die im Bild ersichtlichen Metainformationen mit den im InfoPath-Formular definierten Eingabefeldern vergleicht, merkt man, dass nicht alle Begriffe des Formulars, sondern nur die einzeiligen erforderlichen Eingabefelder in die SharePoint-Liste aufgenommen worden sind. So eine mit reduzierten Informationen ausgestattete Liste ist für die Anwender leichter zu handhaben. Dabei erhöht die so getroffene Auswahl die Übersicht der Dokumentliste, ohne die erforderliche Fülle an Informationen einzuschränken. Dem Anwender bleibt zusätzlich

die Möglichkeit, sich über die weiteren Informationen im der Formularansicht (Abb. 4-2) anzeigen zu lassen. Das Formular kann durch das Anklicken des Icons in der Spalte „Typ“ abgerufen werden.

Im Folge zeigen Abb. 4-4 bis Abb. 4-7 die wichtigsten Funktionen beim Arbeiten mit den SharePoint-Listen, darunter: die Eingabe neuer Daten, das Löschen und Ändern bereits bestehender Daten, so wie das Sortieren und Einschränken der Listen nach einem der vorhandenen Deskriptoren.

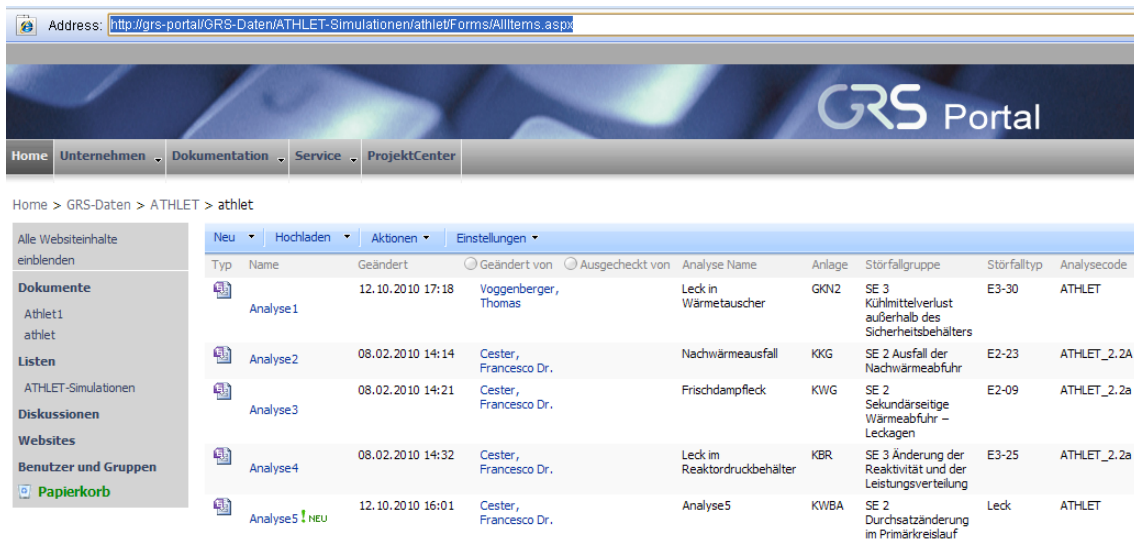


Abb. 4-4 SharePoint-Liste mit den Metainformationen von Simulationsrechnungen

Die Eingabe neuer Analysedaten erfolgt über die Option „Neu“. Der Anwender kann damit sowohl eine neues Dokument (Analyse) als auch einen neuen Ordner anlegen (siehe Abb. 4-5).

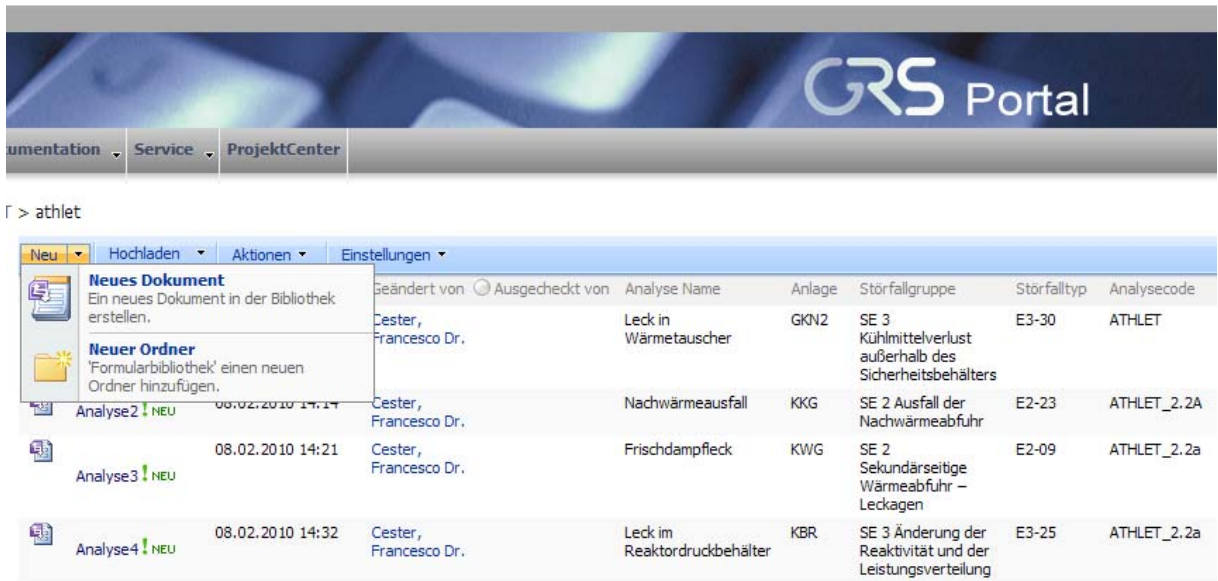


Abb. 4-5 Menü zur Erzeugung neuer Dokumente und Ordner

Abb. 4-6 zeigt das Updatemenü der Dokumente. Dieses Menü ist durch Anklicken des Pfeiles rechts des Dokumentnamens ansprechbar. Über dieses Menü kann der Anwender u.a. die Metainformationen des Dokuments ändern oder auch das Dokument löschen.

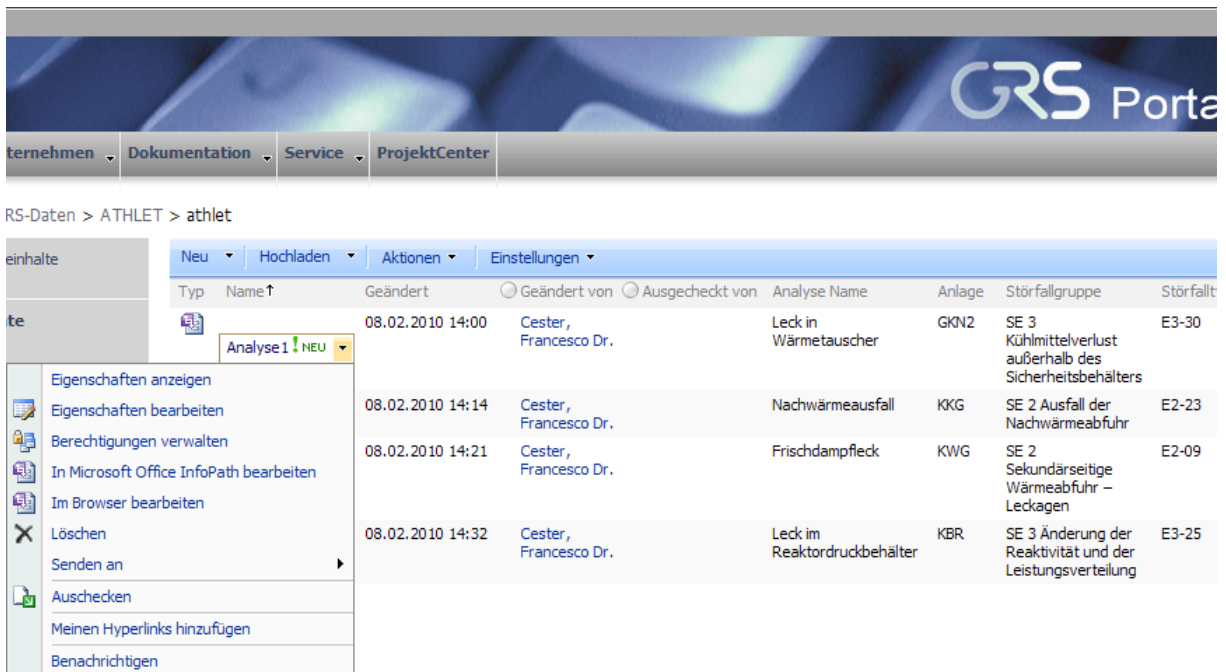


Abb. 4-6 Das Updatemenü der Dokumente

Durch Anklicken der Pfeile rechts der Deskriptoren der Metainformationen, lässt sich ein Pulldownmenü öffnen (Abb. 4-7). Der Anwender hat somit die Möglichkeit, die Liste nach den Metainformationen des gewählten Deskriptors zu sortieren oder deren Inhalt nach einem der vorhandenen Begriffe einzuschränken.

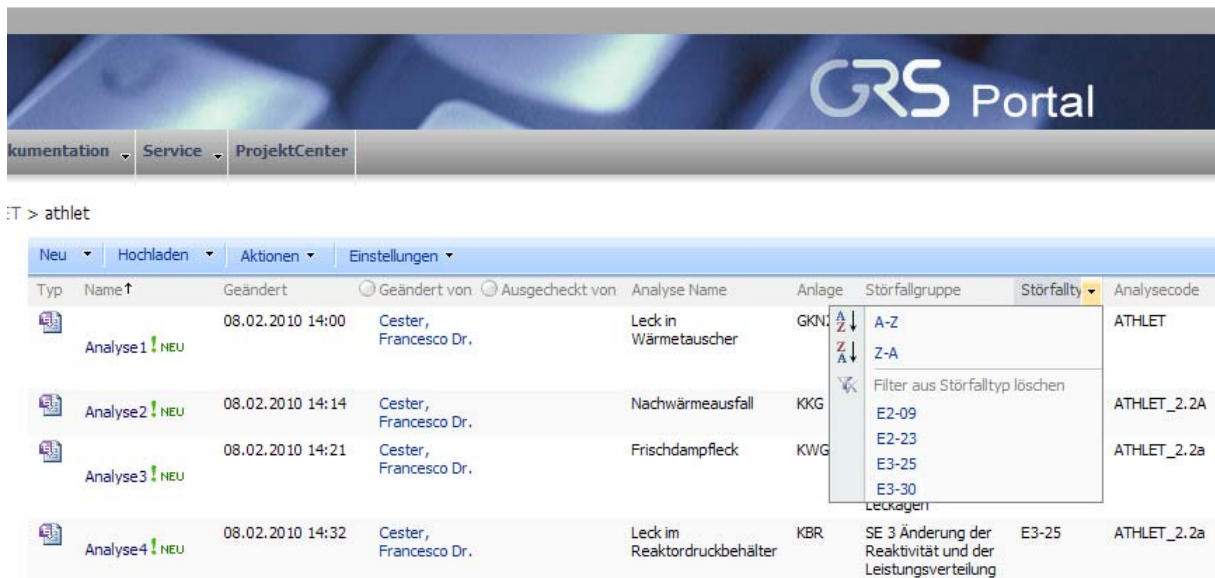


Abb. 4-7 Menü zum Sortieren und Filtern der Metainformationen

Weitere nützliche Funktionen, wie „Spalten und Ansicht Erstellen“ usw., sind unter der Optionsgruppe „Einstellungen“ erreichbar.

4.7.3.1 Die Speicherung der Daten

Die physische Speicherung der Daten erfolgt auf dem Fileserver NetApp FAS2050 /NET 10/ am Standort Garching der GRS (Abb. 4-8). Die aktuelle gesamt Plattenkapazität von 8,9 TB lässt sich um weitere Plattensysteme bis zu ca. 80 TB erweitern. Die FAS2000 Serie bietet Storage für Abteilungen und Niederlassungen in verteilten Unternehmensumgebungen, integrierten Zugriff Datei-Ebene, intelligente Management-Software und Datensicherheit. Kern der Lösung ist das Betriebssystem „Data ONTAP 7G“. Es ist auf Speicher-Effizienz ausgelegt und vereinheitlicht zugleich die Speicherung und das Management der Daten. Von besonderer Bedeutung ist die Möglichkeit, virtuelle Laufwerke unabhängig von physikalischen Gegebenheiten zu definieren. Dazu können Systemadministratoren Datenbäume einfach innerhalb der zugrunde liegenden Storage-Infrastruktur abbilden, managen und migrieren, ohne dass dabei nachgelagerte Applikationen oder Datenbanken betroffen werden. Über die intuitive „NetApp

SnapDrive“ Oberfläche können virtuelle Laufwerke online hinzugefügt werden und die Kapazität bei Bedarf erweitert werden. Der Zugriff auf dieses Laufwerk erfolgt über den virtuellen Hostnamen „grsvrd“, was für GRS Versuchs- und Rechen­daten steht. Die freigegebenen Verzeichnisse sind sowohl unter UNIX und WINDOWS mittels NFS und SAMBA ansprechbar. Ähnlich wie bei den Sharepoint Metadaten ist auch bei den Nutzdaten der Zugriff bislang nur aus dem Intranet möglich, da der Fileserver durch die Firewall der GRS geschützt ist.

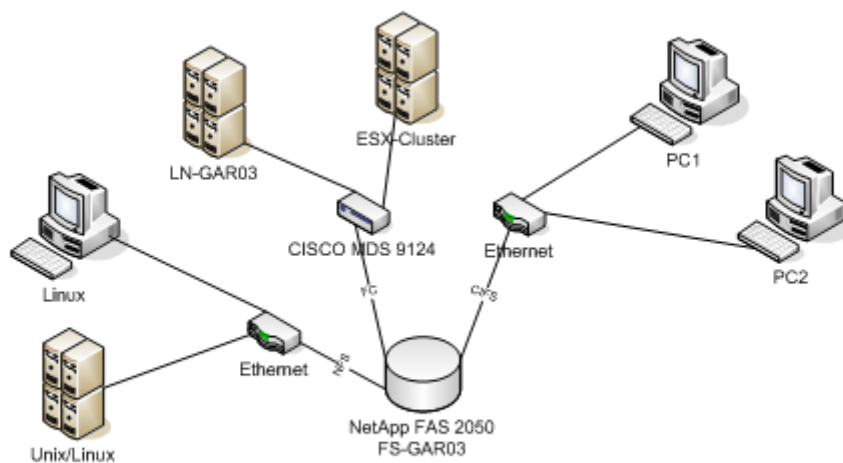


Abb. 4-8 FAS2050 GRS (Garching)

Die Datensicherung erfolgt automatisch in regelmäßigen Zeitabständen mit dem Magnetband basierten System IBM Ultrium 3 /IBM 10.1/ und der Backup Software „Tivoli Storage Manager(TSM)“ /IBM 10.2/, ebenfalls von IBM. Die TSM-Server und das zusätzlich notwendig Datenbanksystem zur Verwaltung der Dateinformationen (DBMS) läuft auf einer IBM-Workstation mit 1,5TB Plattenspeicher zur Pufferung der Daten. Mit diesem System ist selbstverständlich auch die Speicherung verschiedener zeitlicher Versionen von Dateien und Verzeichnissen möglich. Das System kann in der gegenwärtigen Ausbaustufe 300 Magnetbänder mit jeweils ca. 700GB Nutzdaten (komprimiert) für eine automatische Datenrückholung (Retrieval) zur Verfügung stellen. Die Datenrate beim Zurückspielen auf eine magnetische Festplatte von etwa 100GB pro Stunde ist ausreichend schnell.

4.7.4 Test des Systems

Im Folgenden wird anhand eines einfachen Beispiels beschrieben, wie Simulationsergebnisse aus einer ATHLET-Rechnung mit Hilfe der dafür angelegten Dokumentationsbibliothek gespeichert werden können:

- Zuerst wird ein Internet-Browser (z. B. Microsoft IE) gestartet.
- Über die Adresse: <http://grs-portal/GRS-Daten/ATHLET-Simulationen> wird die ATHLET-Dokumentbibliothek geöffnet.
- Der Reiter „Neu“ (unmittelbar über dem Deskriptor „Typ“ der Metainformationsleiste) sowie die Fläche „Neues Dokument“ in dem sich öffnenden Menü werden sukzessiv mit der linken Maustaste angeklickt.
- Mit dem Reiter „Neu / „Neues Dokument“ wird ein Formular mit den Eingabefeldern für die Metainformationen geöffnet (Abb. 4-2).
- Der Anwender sollte die Einträge im Formular möglichst umfassend eingeben. Dabei ist zu beachten, dass die Angaben in den durch einen roten Stern gekennzeichneten Eingabefeldern immer erforderlich sind. Die Eingabefelder „DatenVerzeichnis“, „DatenFile“ und „GraphikFile“ sind mit der Bezeichnung [\\grsvrd\versuchsdaten\simulation](#) vorbelegt, dem Namen des Verzeichnisses in dem die Daten auf dem Fileserver gespeichert werden. Die Angaben der Metadaten im Formular lassen sich durch Nutzung des Reiters „Speichern unter...“ abspeichern“. In dem in Abb. 4-9 gezeigten Dialog muss ein eindeutiger Name (z. B. eine Kombination aus Fallbezeichnung und Anlagenname) vergeben werden.
- Nach der Speicherung des Formulars entsteht ein neuer Eintrag (Zeile) in der Dokumentenbibliothek. In der ersten Spalte steht der soeben vergebene Name.
- Sofern noch nicht geschehen, müssen nun noch die Nutzdaten auf den Fileserver übertragen werden. Man klickt den Hyperlink unter der Bezeichnung „DatenVerzeichnis“ in der gerade erzeugten Zeile an. Ein Explorer Fenster-Directory mit dem Dateipfad dem Fileserver öffnet sich automatisch. Dorthin werden dann alle zu speichernden Dateien kopiert werden müssen.

Speichern unter

Unterstützt von:
 InfoPath Forms Services

Sie können die Datei nur auf der aktuellen Website speichern.

Vorhandene Datei überschreiben?	<input checked="" type="checkbox"/>
Speichern in:	[http://grs-portal/GRS-Daten/ATHLET-Simulationen/athlet/]
Dateiname:*	<input type="text" value="mytest.xml"/>

[Speichern](#) | [Abbrechen](#)

Abb. 4-9 Dialog zum Speichern der Formulareingabe

4.7.5 Beschränkungen

Sonderfunktionen, wie beispielsweise Datenanzeige in Diagrammform in der WEB-Oberfläche, sind nur mit Sharepoint-Programmierung realisierbar. Wegen des relativ hohen Aufwands werden solche Optionen erst bei entsprechendem Bedarf realisiert. Die Datenauswertung muss daher gegenwärtig mit anderen Werkzeugen (z. B. ATLAS) erfolgen.

Zusätzliche Kontrollen im System wären wünschenswert, z. B. um festzustellen ob die Dateien und Verzeichnisse wirklich vorhanden sind. Auch andere Daten wie Eingabedaten, Restart-Datei und Programm-Version sollten erfasst werden, falls man die Ergebnisse reproduzieren möchte. Wichtig für eine spätere Auswertung sind Daten, die die Visualisierung der Anlagengeometrie in Verbindung mit der Ortsdiskretisierung im Modell ermöglichen. Zu dieser Kategorie gehören auch alle anderen Files die zur Visualisierung mit ATLAS verwendet werden können, z. B. die APG-Files. Zur dritten Kategorie gehören schließlich eventuelle vorhandene Dokumentationsdateien.

Zurzeit kann der Datenzugriff nur aus dem Intranet der GRS erfolgen, ist aber auch auf externe Benutzer erweiterbar. Für diesen Fall entstehen geringfügige zusätzliche Lizenzkosten, die entsprechend der autorisierten Benutzeranzahl gestaffelt sind.

5 Programmoptimierung und Dokumentation

5.1 Optimierung des Visualisierungsprogramms

5.1.1 Speicherung von Visualisierungsoptionen

In früheren ATLAS-Versionen wurden die Optionen für Bilder und Zeitdiagramme in verschiedenen Dateien gespeichert, die dann beim Programmstart über zwei Eingabefelder eingelesen wurden. Der Vorteil eines solchen Konzepts bestand vor allem darin, Bilder und Zeitdiagramme unabhängig voneinander anzusprechen und beliebig miteinander kombinieren zu können. Die mehrjährige Anwendung von ATLAS hat indessen gezeigt, dass in der Praxis die Vorteile der Handhabung eines einzigen Eingabedatensatzes für Bilder und Zeitdiagramme, die der Flexibilität überwiegen, die man sich durch deren unabhängige Abspeicherung versprochen hatte. Es stellte sich insbesondere heraus, dass für eine bestimmte Analyse Bilder und Zeitdiagramme eher aufeinander angepasst sein müssen. Vor allem diese Erkenntnis in Verbindung mit dem Bestreben nach einfachen Lösungen führte zum neuen Konzept einer zentralen Optionsdatei für Bilder und Diagramme.

Für ATLAS wurde daher eine zentrale Options- und Bilddatei (Graphic Definition File GDF) mit der Erweiterung „*.atl“ für Bilder und Trends und einige generelle Einstellungen eingeführt. Diese Datei beinhaltet alle Angaben, die das Programm zum Aufbau der Fenster für Bilder und Zeitdiagramme braucht und ersetzt somit bei der ATLAS-Eingabe die früheren APG- und Trend-Definitionsdateien. Das folgende Beispiel eines GDF beinhaltet die Spezifikationen, die dazu benötigt werden, um zwei Fenster mit Bildern, ein Trendset mit zwei Zeitdiagrammen und ein Fenster mit einem Zeitdiagramm aufzubauen.

```
=simu
[mainwindow_positions]
  -4 1204 -4 1894 365920256 256
[time_control]
  3 10000.0000000 1.0000000 1.0000000
[picture_files]
=== 'D:\Simulators\gkn1\npa\inputgrafik.apg' ====> comment card
'..\..\..\npa\inputgrafik.apg'
window: 4 0 740 0 612 0 0
=== 'D:\Simulators\gkn1\npa\notspeisewasser.apc' ====> comment card
'..\..\..\npa\notspeisewasser.apc'
window: 3 183 923 445 1057 0 0
[timeaxis]
  0 1 0 0 0
```

```

[trendsets]
=====
'Trend set 1'
window: 2      0  789      0  658      0      0
=plt_start
Page(:nbx 1, :nby 1);
...
...
EndPage();
=plt_end
[extended_trends]
=====
'P1-SG-UT      -      NODE5      PRESS '
window: 1      745  1065      564  847      0      0
=plt_start
Page(:nbx 1, :nby 1);
...
...
EndPage();
=plt_end

```

Aus dem hier wiedergegebenen Beispiel lässt sich erkennen, dass ein GDF aus 6 verschiedenen Abschnitten besteht, die jeweils durch ein in eckigen Klammern eingeschlossenes Schlüsselwort eingeleitet werden:

- **mainwindow_positions**

Diesem Schlüsselwort folgen 6 Ganzzahlen: **left, right, top, bottom, style, exstyle**.

Die ersten vier Zahlen geben Abstände in Pixels an („left“ und „right“ zwischen der linken bzw. rechten Seite des Fensterrahmens und dem linken Bildschirmrand; „top“ und „bottom“ zwischen der oberen bzw. unteren Seite des Fensterrahmens und dem oberen Bildschirmrand) und bestimmen somit Position und Dimensionen des ATLAS-Hauptfensters. Die letzten zwei Ganzzahlen, hängen mit der Gestaltung des Fensterrahmens zusammen. Sie werden vom Programm gesetzt und dürfen nicht vom Anwender manuell verändert werden.

- **time_control**

Die unter diesem Schlüsselwort vorhandenen Zahlen entsprechen den Angaben, die man im Menü „Time Control“ (unter der Option „Control“ / „Times“) eingeben kann:

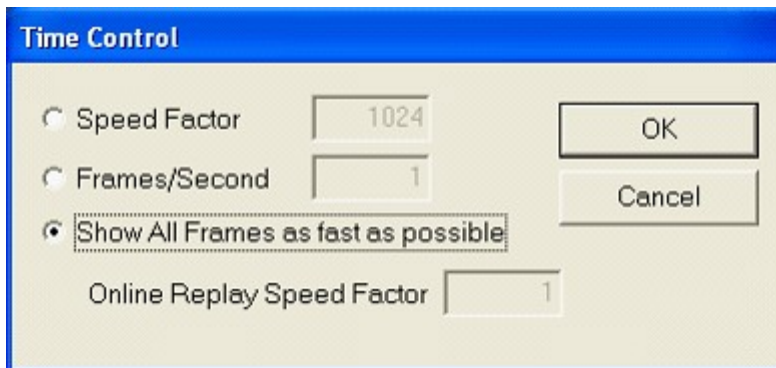


Abb. 5-1 Optionen zur Darstellung verschiedener Zeitpunkte in ATLAS

Die erste Zahl kann die Werte 1, 2 und 3 annehmen und gibt an, welche der drei Optionen „Speed Factor“ (Wert = 1), „Frames/Second“ (Wert = 2), bzw. „Show All Frames as fast as possible“ (Wert = 3), angewählt wird. Es folgen 3 weitere reelle Zahlen. Sie geben in Folge die Werte für „Speed Factor“, „Frames/Second“ und „Online Replay Speed Factor“ an.

- **picture_files**

In diesem Abschnitt werden die Prozessbilder (APG-Files) angegeben, die eingelesen und anschließend in die Picture-Liste von ATLAS eingefügt werden sollen. Das beige-fügte GDF-Beispiel zeigt, dass bei der Dateiangabe immer der relative Verzeichnispfad zum Arbeitsverzeichnis spezifiziert wird. Das hat den Vorteil, dass die Kompatibilität des Graphic Definition Files auch nach der Übertragung der ATLAS-Daten in andere Verzeichnisse erhalten bleibt. Durch die Angabe des vollen absoluten Pfads in einer Kommentarzeile ist sichergestellt, dass die Information über die Herkunft der Files nicht verloren geht. Jeder Dateiname entspricht einem Bild. Folgt dem Dateinamen noch eine durch das Wort „**window**:“ eingeleitete Zeile, dann wird das Bild nach dem Start von ATLAS in einem eigenen Fenster sofort angezeigt. Die erste Zahl entspricht der Reihenfolge (Priorität), in der das Fenster erscheinen soll. Bei niedrigen Zahlen werden die Fenster im Vordergrund bei hohen Zahlen im Hintergrund eröffnet. Durch die restlichen 6 Zahlen werden Position und Maße der Fenster genau wie unter dem Schlüsselwort „**mainwindow_positions**“ festgelegt.

- **timeaxis**

Unter diesem Schlüsselwort wird die Länge der Zeitachse für die Zeitdiagramme gespeichert. Die dafür erforderliche Zeitangabe wird durch die ersten zwei Ganzzahlen für die Stunden und Minuten, und durch eine bis zur dritten Nachkommastelle genaue Dezimalzahl für die Sekunden angegeben. Die vierte Zahl kann 0 oder 1 sein; sie gibt an in welcher Form die Zeitangaben in den Diagrammen ausgegeben werden sollen. 0 steht für eine Angabe in Sekunden. 1 steht für eine Angabe von Typ hh:mm:ss. Die fünfte Ganzzahl entspricht der Option „Increasing Time Axis“. Ist sie auf 1 gesetzt, dann verlängert sich die Zeitachse der Diagramme bei fortschreitender Zeit automatisch, ist sie hingegen gleich 0, dann bleibt die Länge der Zeitachse unverändert und das Diagramm wird nach links verschoben.

- **trendsets**

In diesem Abschnitt werden sogenannte „Trendsets“ definiert, die aus mehreren Einzeldiagrammen („Achsenkreuzen“) bestehen und zu einer zusammenhängenden Darstellung kombiniert sind (Abb. 5-2).

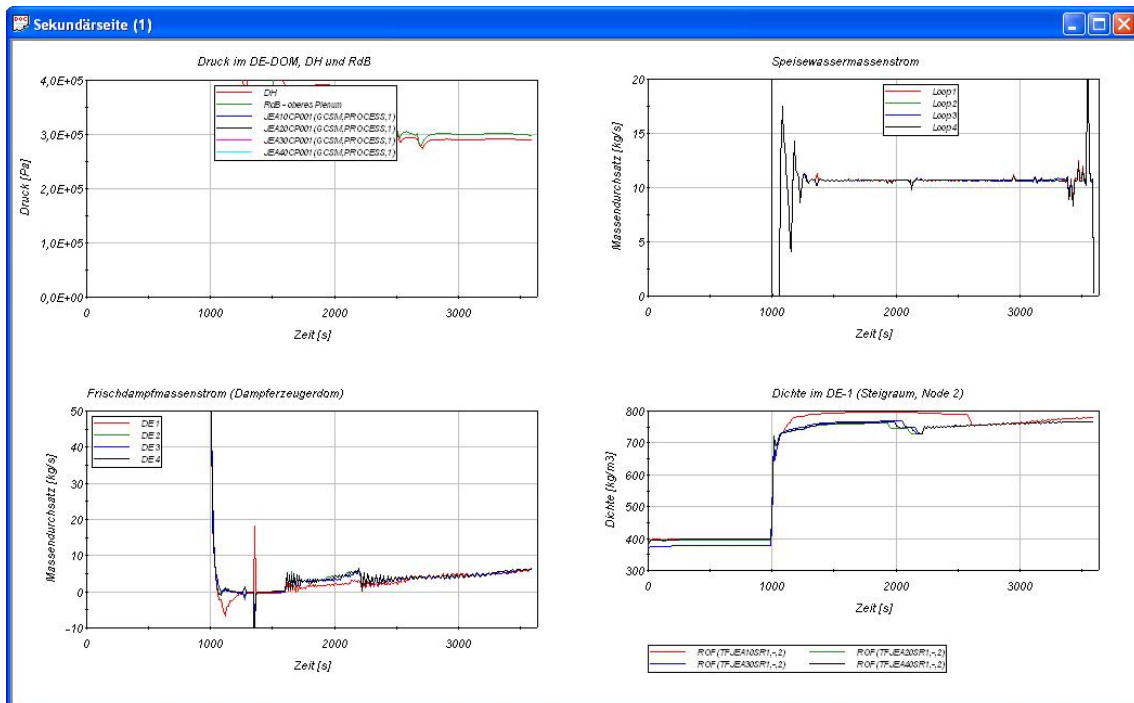


Abb. 5-2 Beispieldarstellung eines Trendsets in ATLAS

Im Anhang 2 ist die Syntax zur Definition der Trendsets, der Diagramme und für die

Kurven detailliert beschrieben. Diese ist sehr ähnlich zu der im Standard-ATLHLET-Plotprogramm verwendeten, die in der Online-Hilfe "Function Reference" von JSPLLOT /SIE 09/ beschrieben ist. Dies erleichtert den Austausch der Definitionsdateien für die beiden Anwendungen erheblich. Alle Definitionsdateien für JSPLLOT können direkt in ATLAS importiert werden, wobei Darstellungsoptionen, die in ATLAS nicht unterstützt werden, ignoriert werden. Umgekehrt ist es ebenso möglich aus der Bedienoberfläche von ATLAS Definitionsdateien für JSPLLOT(Erweiterung *.plt) zu erzeugen, die dann zu einer weitgehend ähnlichen Darstellung in dem ATHLET Werkzeug genutzt werden können.

- **extended_trends**

In diesem Abschnitt werden die die Trenddiagramme mit einem einzelnen Achsenkreuz und genau einer Variablen aufgeführt, die der Benutzer durch einen Doppelklick auf eine Schaltfläche im Bild oder eine Variable im Keyword-Baum erzeugt hat. Die einzelnen Eigenschaften werden entsprechend wie bei „trendsets“ definiert.

5.2 Optimierung des Bildeditors APG

Im interaktiven Editor zur Erzeugung der Bilder werden einige Funktionen zur Definition der dynamischen Bildeffekte nicht unterstützt. Diese mussten bisher manuell in der Bilddatei vorgegeben werden, was sehr fehleranfällig war. Häufig benötigte Funktionen wurden daher als interaktive Optionen im Bildeditor hinzugefügt.

5.2.1 Farbänderungen nach Farbskalen

5.2.1.1 Einfache Farbskala

In der neuen Version des grafischen Bildeditorprogramms APG ist unter dem Reiter der Hauptbedienleiste „Utilities“ die Funktion „Color Scale“ eingeführt worden (Abb. 5-3). Dabei handelt es sich nicht um eine von Grund auf neue Funktion, denn die Möglichkeit eine Farbskala zu definieren hat es im APG-Programm schon immer gegeben. Die interaktive Erzeugung im Bildeditor wurde aber kaum unterstützt. Erstens zählte sie nicht zu den Hauptoptionen des Programms und deswegen konnte sie auch nicht unter einem der Reiter der Hauptbedienleiste gefunden werden. Vielmehr war sie der Funktion zur Dynamisierung von Flächen hierarchisch untergeordnet mit der Folge, dass ein

Bediendialog nur dann erschien, wenn man eine physikalische Größe einem bestimmten grafischen Element zuordnete. Außerdem war ihre gesamte Funktionalität nicht in einer einzigen Bedienungsfläche integriert, denn beispielweise zur Definition der Farbskalenanzeige (Balkendarstellung) musste der Anwender das entsprechende Makro aufrufen und dessen Parameter manuell ändern. Durch die Einführung der neuen Funktion „Color Scale“ ist die Bedienung einfacher und übersichtlicher geworden.

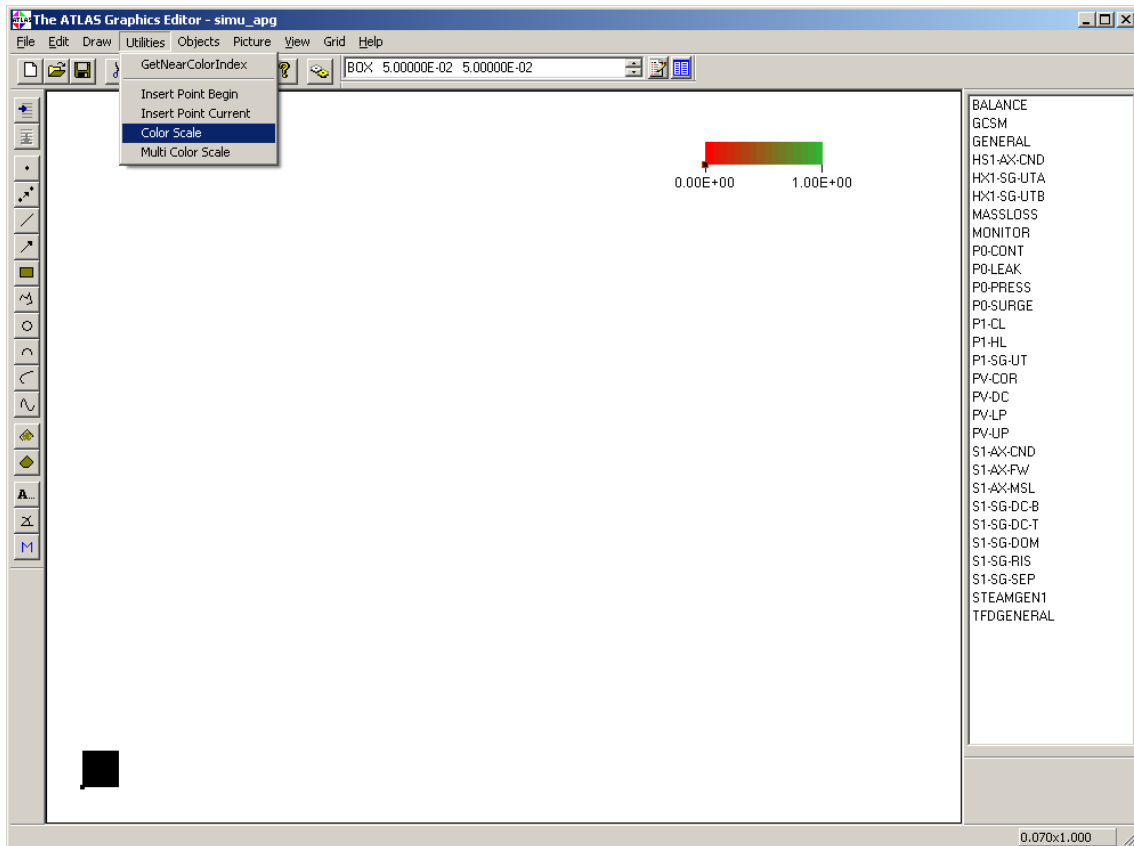


Abb. 5-3 Bildeditor APG - Farbskalenerzeugung

Der Anwender definiert zuerst die von ihm benötigten Farbskalen (momentan maximal 20) in dem in Abb. 5-4 gezeigten Dialogfenster.

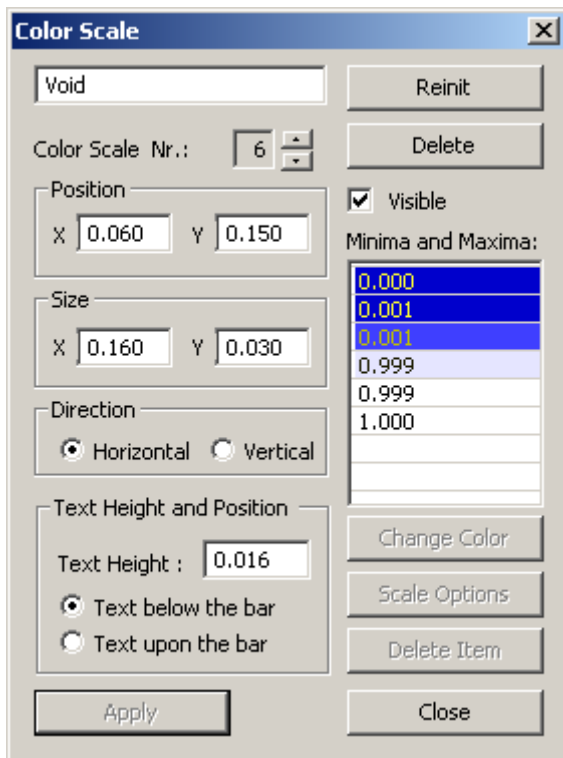


Abb. 5-4 Definition von einzelnen Farbskalen

Im Folgenden werden die Dialogelemente beschrieben, mit denen man die Optionen für die Farbskalen einstellen kann.

- **Color Scale Nr.**

Mit dieser Option lässt sich zu den Eintragungen einer bestimmten Farbskala umschalten. Von diesen sind die ersten 8 vorbelegt. Ihre Minima, Maxima und Farben eignen sich für folgende, häufig verwendete physikalische Größen:

Ventile (Valves), Leistung (Power), Druck (Pressure), Temperatur (Temperature), Dichte (Density), Dampfinhalt (Void), Wassermassenstrom (Liquid flow), Dampfmassenstrom (Vapor flow)

- **Reinit**

Initialisiert die unter der entsprechenden Nummer definierte Farbskala mit ihren Anfangswerten.

- **Delete**

Löscht die unter der entsprechenden Nummer definierte Farbskala.

- **Visible**

Diese Option gibt an, ob die Farbskala im Bild sichtbar sein soll oder nicht. Nur dann wenn „Visible“ vermerkt ist, sind die Optionen für „Position“, „Size“, „Direction“ und „Text“ aktiv und können gesetzt werden.

- **Position**

Unter „Position“ können die Koordinaten der oberen linken Ecke der Farbskala-Anzeige gesetzt werden. Die Werte liegen zwischen 0.0 und 1.0.

- **Size**

Unter „Size“ kann die Länge der horizontalen bzw. vertikalen Seite der Farbskala-Anzeige angegeben werden.

- **Direction**

Mit dieser Option lässt sich bestimmen, ob die Farbskala waagrecht oder senkrecht gezeichnet werden soll.

- **Text Height**

Eingabe der Schrifthöhe.

- **Text below the bar (Text right of the bar)**

In Abhängigkeit von der Option „Direction“ legt diese Option die Position der Schriften unterhalb bzw. rechts der Farbskala-Anzeige fest.

- **Text upon the bar (Text left of the bar)**

In Abhängigkeit von der Option „Direction“ legt diese Option die Position der Schriften oberhalb bzw. links der Farbskala-Anzeige fest.

- **Minima and Maxima**

Durch Betätigung einer Maustaste im Bereich „Minima and Maxima“ lassen sich die einzelnen Zeilen zum Zwecke des Löschens bzw. der Eingabe der Minimal- und Maximalwerte markieren, wodurch auch die Funktionen zur Bestimmung der Farbe und zur Veränderung der Attributen (siehe unten) aktiviert werden.

- **Scale Options**

Der Dialog für spezielle Farbskalenoptionen wird angezeigt (Abb. 5-5). Er ermöglicht die Eingabe der Anzahl der benutzten Farbstufen bis zur nächsten definierten Farbe und Ein- bzw. Ausschaltung der Anzeige dieses Skalenwerts in der Skalenbeschriftung.

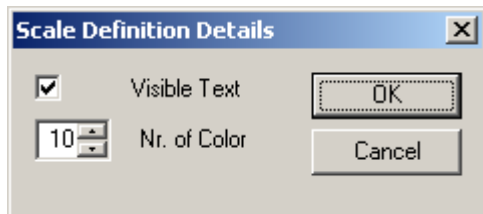


Abb. 5-5 Scale Definition Details

- **Change Color**

Änderung der Farbe, ein Farbauswahlwählfenster entsprechend dem WINDOWS-Standard erscheint (Abb. 5-6).



Abb. 5-6 Dialogfenster zur Farbauswahl

- **Delete Item**

Beim Betätigen dieser Schaltfläche wird aus der Liste die Zeile gelöscht, die vom Anwender durch einfachen Klick zuletzt markiert wurde.

- **Apply**

Diese Schaltfläche dient zum Übernehmen der im Dialog durchgeführten Änderungen der gewählten Farbskala. Ist sie inaktiv, dann ist die Farbskala bereits definiert mit den im Dialog eingetragenen Spezifikationen. Dabei ist zu beachten, dass die 1. Farbskala für Ventile auch ohne Benutzung des Dialogs immer vordefiniert ist und verwendet werden kann.

Bei der Dynamisierung von grafischen Elementen, kann dann die Farbe entsprechend der gewählten physikalischen Größe einer geeigneten Farbskala zugeordnet werden (Abb. 5-7). Dabei ist zu beachten, dass nur die zuvor definierten Farbskalen wählbar sind.

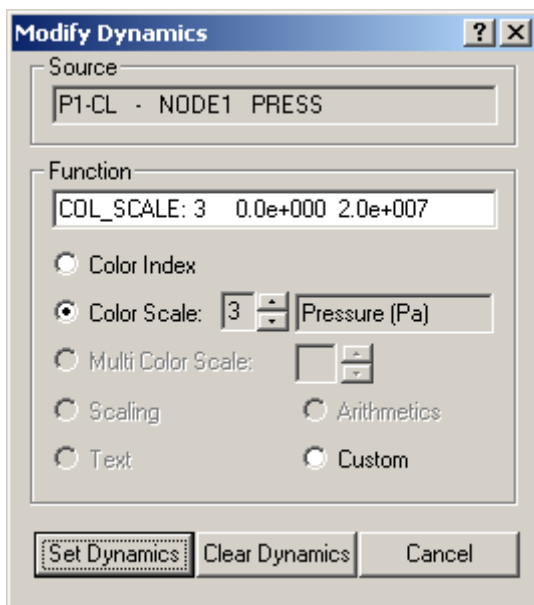


Abb. 5-7 Zuordnung einer Farbskala zu einem Grafikelement

- **Close**

Schließt das Dialogfenster.

5.2.1.2 Umschaltbare Farbskala

In der neuen Version des grafischen Bildeditorprogramms APG ist unter dem Reiter der Hauptbedienleiste „Utilities“ die neue Funktion „Multi Color Scale“ (Abb. 5-3) eingeführt worden. In vielen Prozessbildern von ATLAS-Anwendungen werden Darstellungen des örtlichen Verlaufs lokaler Größen; wie z. B. Temperatur oder Dampfgehalt in Leitungen; mittels Farbschattierungen dargestellt. Für jede physikalische Größe ist es üblich, geeignete Farbskalen zu definieren. Die aktuell angezeigte Größe kann dann mit einem Mausklick in einem Menü ausgewählt werden. Zwar hat es in der grafischen APG-Metasprache schon immer die Möglichkeit gegeben, eine umschaltbare Multifarbskala, zu definieren. Die Benutzung war aber aufwendig und erforderte seitens des Benutzers sehr gute Kenntnisse der APG-Syntax, insbesondere wegen der vielen manuellen Eingaben, die in mehreren Abschnitten der APG-Datei notwendig waren und vom interaktiven Editor nicht unterstützt wurden. Da diese Funktion für viele ATLAS-Anwendungen immer wieder benötigt wurde, musste man zwangsläufig zeitaufwendige Bearbeitungen der APG-Files und oft auch kostspielige Einarbeitungen externer Benutzer durchführen.

Durch die Einführung der neuen Funktion „Multi Color Scale“ ist die Aufgabe für die Benutzer einfach und übersichtlich geworden. Der Anwender definiert zuerst die von ihm gebrauchten Multifarbskalen (bis zu acht) in dem in Abb. 5-8 gezeigten Dialogfenster. Bei der Dynamisierung kann er dann die Farbe eines grafischen Objektes mit dem im vorigen Abschnitt beschriebenen Dialog (Abb. 3-7) der dazu geeigneten Multifarbskala zuordnen. Dabei ist zu beachten, dass nur die bereits definierten Multifarbskalen abrufbar sind. Zur Umsetzung der Funktion in der APG-Sprachsyntax wurde ein neuer Befehl „MULTISCALE“ programmiert, der alle Benutzervorgaben übersichtlich an einer Stelle der Datei zusammenfasst. Ein Beispiel wird in Anhang 3 gezeigt.

Im Folgenden werden die Dialogelemente beschrieben, mit denen man die Optionen für die Multifarbskalen einstellen kann. Da der Dialog ähnlich dem Dialog für eine einzelne Farbskala aufgebaut ist, werden hier nur die davon abweichenden Optionen beschrieben.

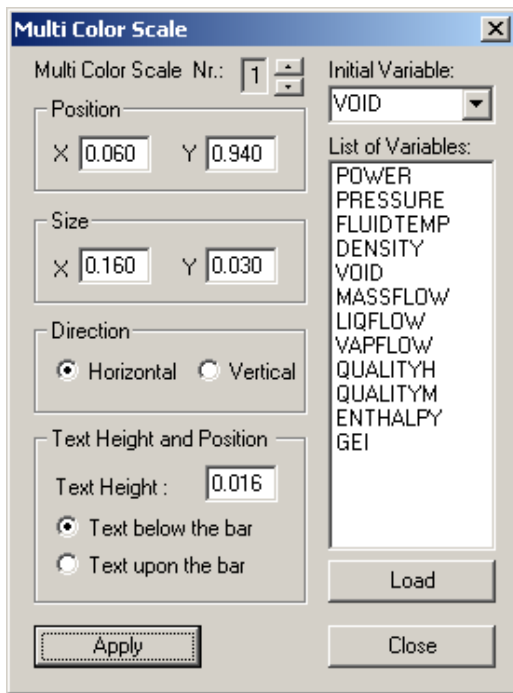


Abb. 5-8 Dialog zur Definition einer Multifarbsskala

- **Initial Variable**

„Initial Variable“ ist die physikalische Größe, mit der die Multifarbsskala initialisiert wird, d. h. die im Prozessbild am Anfang angezeigt werden soll. Sie muss eine der in der Liste eingetragenen Variablen sein.

- **Load**

Diese Option ermöglicht die Vorbelegung einer Multifarbsskala mit Größen, die sich für eine der in Abb. 5-9 wiedergegebenen Objektkategorien (Fluidobjekte, Wärmeleitobjekte, Brennstäbe) eignen. Es ist zu beachten, dass diese Vorgaben speziell auf die Bedürfnisse von ATHLET zugeschnitten sind. Bei Nutzung von anderen Prozessmodellen (z. B. ASTEC) sind im Einzelfall Anpassungen notwendig, die bisher noch nicht automatisiert wurden.

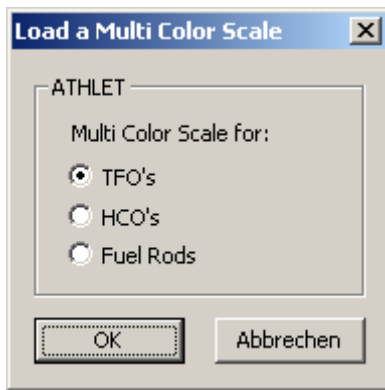


Abb. 5-9 Auswahlmönü für häufig benötigte Multifarbskalen

- **List of Variables**

In „List of Variables“ werden die physikalischen Größen eingegeben, deren lokale Verteilung im Prozessbild angezeigt werden soll.

Nach einem Klick mit der linken Maustaste im Bereich „List of Variables“ auf eine Zeile mit einem Variablenamen kann der Anwender unter einer der folgenden Optionen wählen:

New: Eingabe einer neuen Zeile zur Definition einer thermohydraulischen Größe.

Delete: Löschen einer Zeile.

Edit: Editieren der Spezifikationen der gewählten physikalischen Größe durch folgenden Bedienungsdialog (Abb. 5-10):

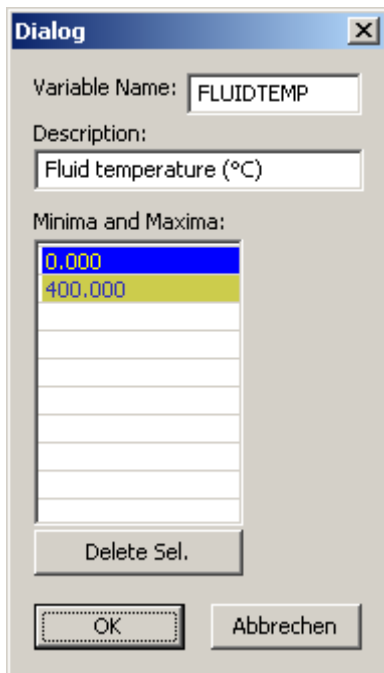


Abb. 5-10 Definition einzelner Farbskalen und Variablen für Multifarbskalen

Es ist zu beachten, dass die Vorgabe des Variablennamens („Variable Name“) exakt der Bezeichnung der physikalischen Größe in der Keyword-Liste entsprechen muss. Die Vorgabe von Werten und Farben im Feld „Minima and Maxima“ erfolgt analog der Beschreibung bei der einfachen Farbskala (Abs. 5.2.1.1). Die im Feld „Description“ eingegebene Beschreibung der Variablen wird im Auswahlménü des Prozessbildes als Zusatzinformation angezeigt.

5.2.2 Steuerung der Sichtbarkeit von Bildelementen

Das Ein- oder Ausblenden von Bildinhalten ist eine von den Anwendern häufig gewünschte Funktion, um beispielsweise Redundanzen in Reaktorsystemen anzuzeigen. Diese Funktionalität kann nun auch interaktiv im Bildeditor erzeugt werden, wodurch aufwendige manuelle Eingaben in der Bilddatei entfallen. Bei der Nutzung im APG wird automatisch ein Skript erzeugt, das in der ATLAS Visualisierung zum Aus- bzw. Einblenden graphischer Objekte, also einer Gruppe von Bildelementen, dient. Dieses Skript muss einem graphischen Element zugeordnet werden (z. B. eine Fläche oder ein Text), das zur Ausführung dieser Funktion als „Schaltknopf“ benutzt werden soll. Nachdem der Anwender das als „Schaltknopf“ gewählte graphische Element angeklickt hat, stehen ihm zur Definition der Funktion „Visibility Script“ zwei Bedienungsmöglichkeiten zur Verfügung.

Die Erzeugung der Funktion ist über direkt über das Objektmenü des Schalters oder den Properties-Dialog zugänglich. Ein Mausklick in Bereich „Pick“ öffnet den „Pick Dialog“, danach muß der Radiobuttons „Visibility Script“ gesetzt werden. (Abb. 5-11).

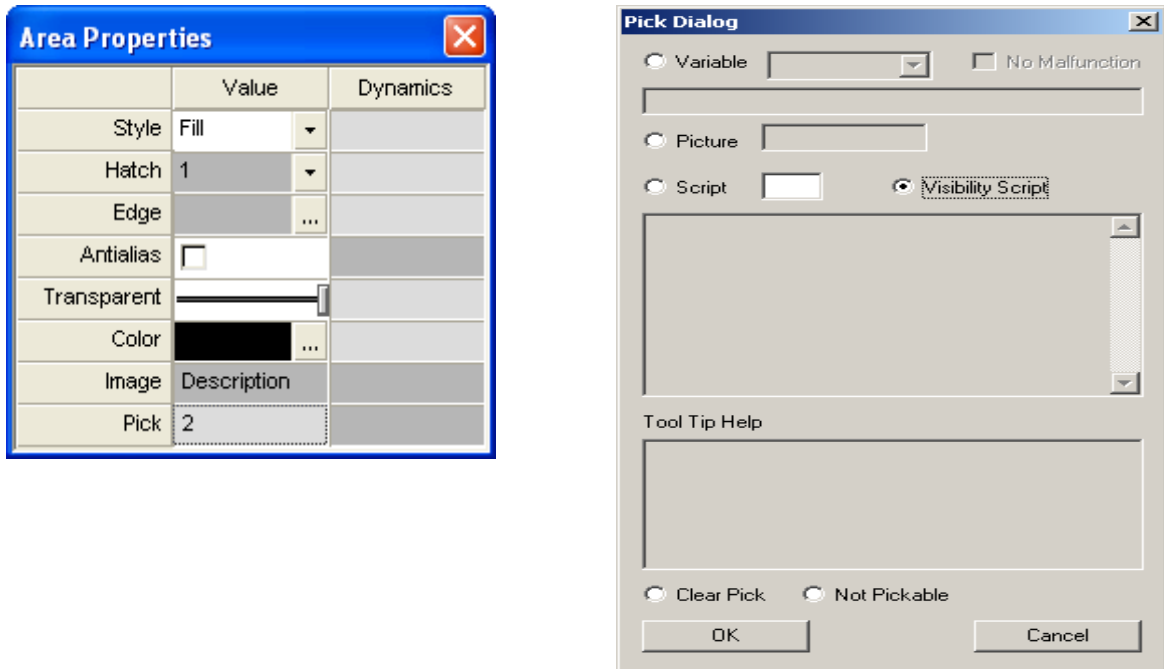


Abb. 5-11 Definition der Funktion „Visibility Script“ im „Pick-Dialog“

Anschliessend können über ein Listenmenü (Abb. 5-12) der APG-Objekte diejenige ausgewählt werden, die bei Betätigung des Schalters aus- bzw. eingeblendet werden sollen. Dabei ist zu beachten, dass der Name des Objekts, das als „Schalter“ dient, in der Liste nicht enthalten ist.

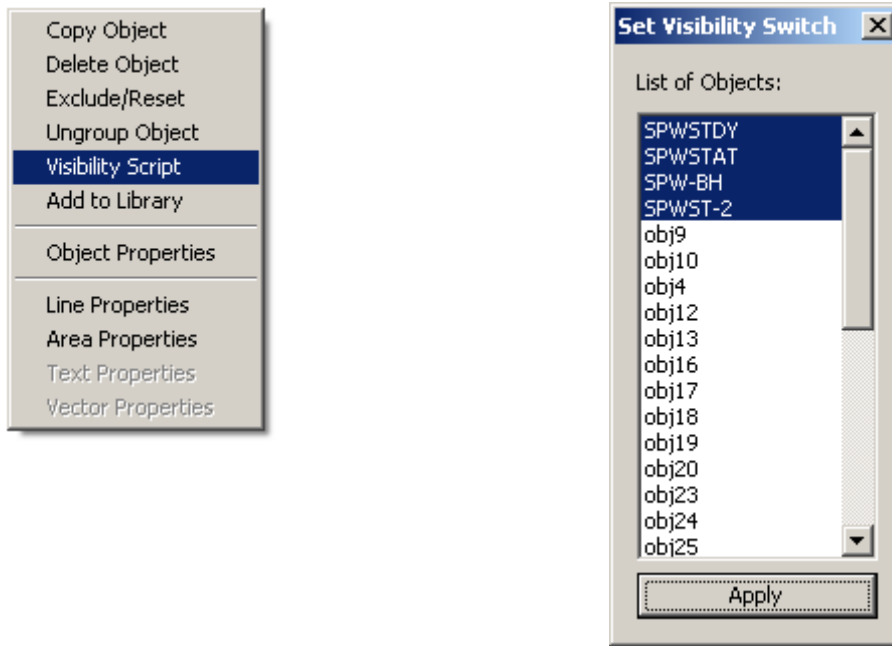
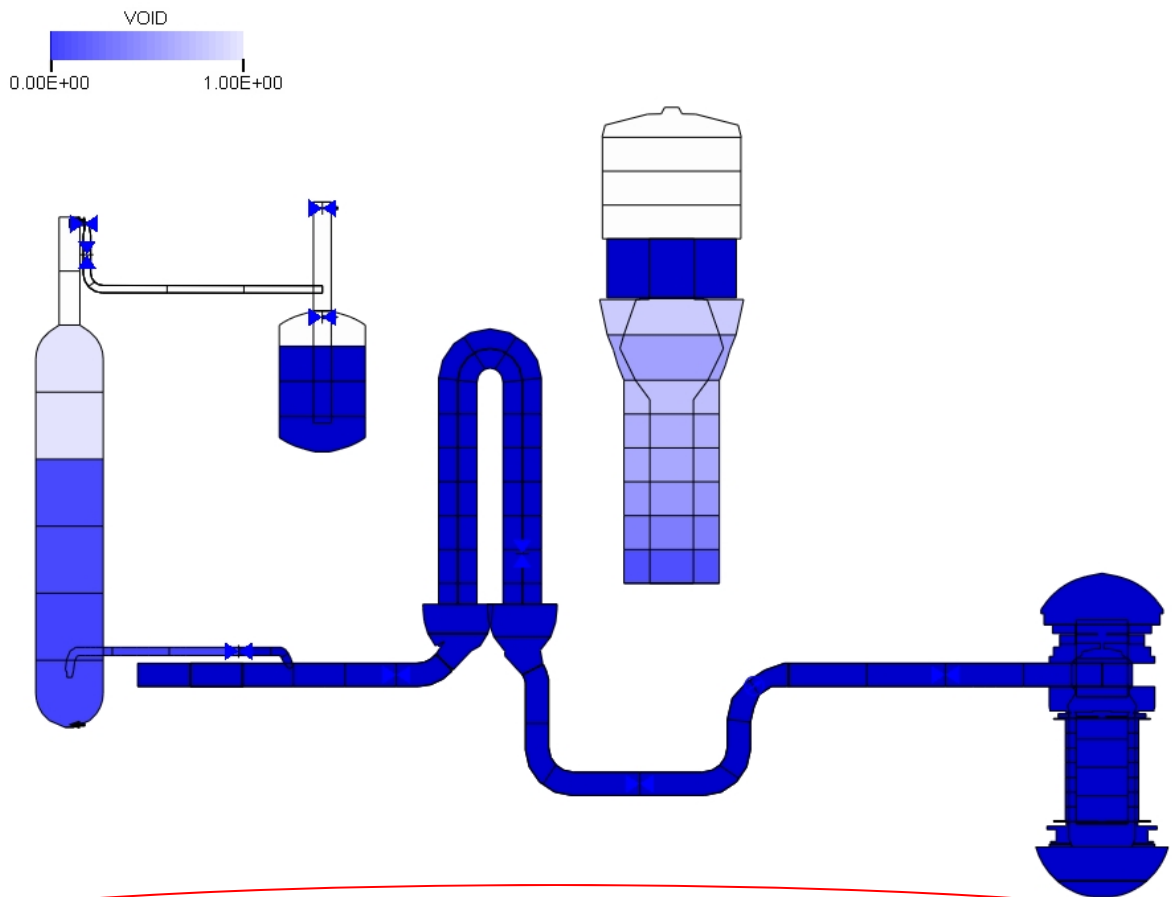


Abb. 5-12 Ansteuerung und Objektauswahl für „Visibility Script“

5.2.3 Zusatzinformation zur Dokumentation der ATLAS-Bilder

ATLAS-Bilder sind in vielen Fällen ein unverzichtbarer Bestandteil für die Dokumentation von Simulationsläufen, z. B. in Berichten oder Veröffentlichungen. Da sie dynamische Eigenschaften besitzen, also zeitveränderliche Informationen anzeigen, ist es wichtig zu wissen, welcher Zeitpunkt der Rechnung einem bestimmten „Screenshot“ des Bilds entspricht, da die interaktive Anzeige der Problemzeit von ATLAS nicht mehr vorhanden ist. In einigen Fällen werden auch andere Details benötigt, wie z. B. das Datum der Durchführung der Simulationsrechnung und die verwendeten Ergebnisdateien.



D:\Aig\work\gkn1.test1.pd

Problem Time: 53.50000 Date: 30.09.2009 10:09:58 ATLAS - VERS. 5.0_090922

Abb. 5-13 ATLAS-Graphik mit Informationszeile

Für diesen Zweck wurde die APG Syntax um einige neue Befehle erweitert. Mit der Eingabe folgender ESCAPE-Befehle ist es ab der ATLAS-Version 5.0 möglich im Bild, ohne großen Aufwand, eine Info-Zeile mit relevanten Informationen über den durchgeführten Rechenlauf einzublenden:

- **ESCAPE DATAFILE**

Links in der Fußzeile wird der Dateiname der Grafikdatei eingeblendet

- **ESCAPE PROBLEMTIME**

In der Mitte der Fußzeile wird die Problemzeit zum Zeitpunkt des mit dem Zusatz „Problem Time“ eingeblendet

- **ESCAPE TODAYDATE**

Das Datum mit dem Zusatz „Date“ wird eingeblendet

- **ESCAPE CLOCKTIME**

Die Uhrzeit zum Zeitpunkt des Ausdruckes wird eingeblendet

- **ESCAPE VERSION**

Die ATLAS-Version wird eingeblendet

Bei der Definition der Fußzeile hat der Anwender natürlich die Möglichkeit einen oder mehrere der hier beschriebenen ESCAPE-Befehle wegzulassen. Alle Befehle müssen in einem APG-Objekt definiert werden. Es ist zu beachten, dass in allen durch die „ATHLET INPUT GRAPHIC“ erzeugten APG-Files eine Fußzeile automatisch generiert wird. Diese lässt sich mit Klick auf die drei Punkte (links unten in Abb. 5-13) ein- oder ausblenden.

5.3 Benutzerhandbücher

Die Benutzerhandbücher für ATLAS sind in konsistenten Online- und Druckversionen als Microsoft Compiled Help Module (CHM) und Adobe Portable Document Format (PDF) in englischer Sprache verfügbar. Es existieren getrennte Manuals zur Bedienung der ATLAS Visualisierung /VOG 10.1/ und zum Bildeditor APG /VOG 10.2/. Zur Erstellung und Wartung dieser Handbücher wird die Software „Fast-Help“ von „DevHost Ltd.“ eingesetzt. Fast-Help ist ein Hilfedatei Editor (Abb. 5-14) zur Erzeugung von HtmlHelp (.CHM), WinHelp (.HLP), Website Hilfe, ausdrucksbaren Anleitungen, MS Word Anleitungen, PDF und PocketPC Hilfe. Die Hilfedaten werden in einem Fast-Help Projekt interaktiv eingegeben und können dort auf vielfältige Weise formatiert, verlinkt und mit Bildmaterial versehen werden. Für die Ausgabe in die unterstützten Zielformate stehen ebenfalls eine Reihe von flexiblen Optionen zur Verfügung. Besonders hervorzuheben ist der Vorteil, dass eine Anpassung der Hilfe nur an einer zentralen Stelle erfolgt und die Zieldokumente danach auf Knopfdruck erzeugt werden können.

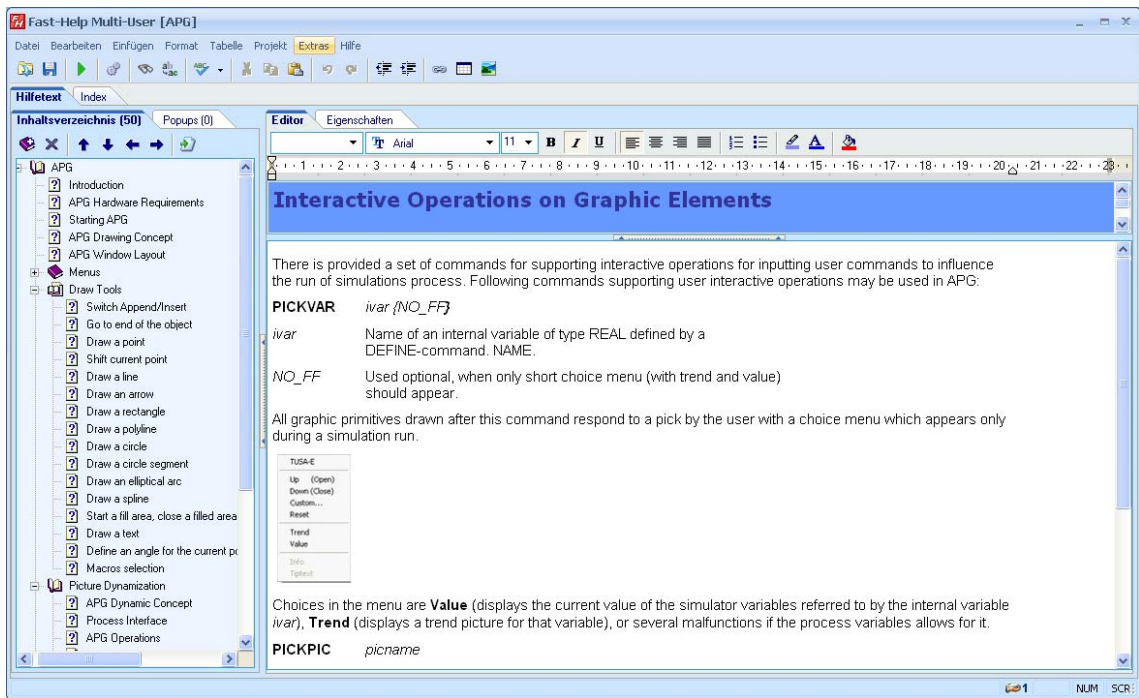


Abb. 5-14 Erstellung des APG User's Manual in Fast-Help

Die Hilfefunktionen im Visualisierungsprogramm und im Bildeditor APG wurden mit dem Benutzerhandbuch verknüpft. In vielen Fällen besitzen die Bediendialoge „Hilfe-Buttons“, die direkt ein zugeordnetes Hilfe-Thema der Online-Hilfe anzeigen. Im Bildeditor APG wird beim Drücken der "F1-Taste" direkt der Hilfeeintrag zu dem aktuell verwendeten grafischen Element angezeigt wird. Die Bedienung der ATLAS Visualisierung wird durch sogenannte „Tooltips“, grafische Texterklärungen die bei der Benutzung von Bedienelementen eingeblendet werden und Informationen in der Statuszeile erheblich erleichtert (Abb. 5-15).

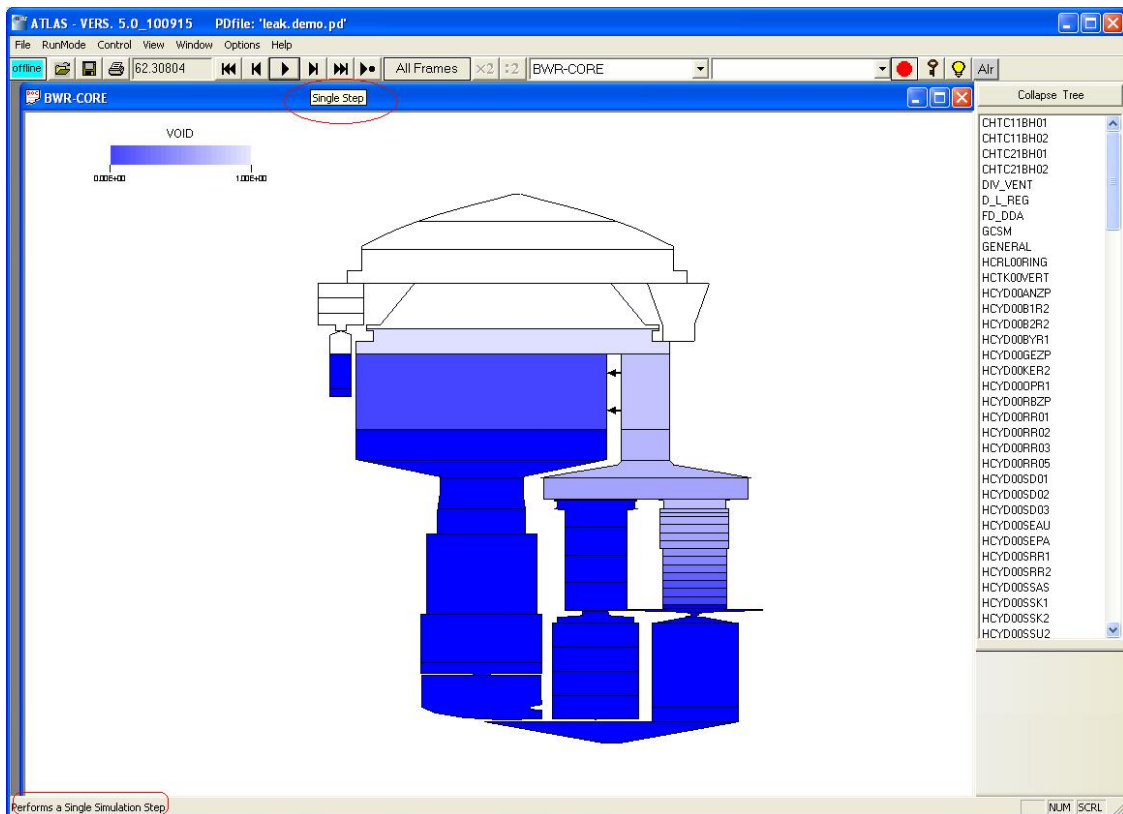


Abb. 5-15 Hilfe mit Tooltip und Statuszeile

5.4 Benutzerforum

Ein Benutzerforum für ATLAS wurde eingerichtet. Dort ist der Zugriff auf relevante Informationen (z. B. Veröffentlichungen, Benutzerhandbücher, aktuelle Ankündigungen) möglich und eine Diskussionsplattform zu Problemen und Erfahrungen mit der ATLAS Software vorhanden. Die Verfügbarkeit des Forums kann für interne und externe Nutzer mit den benötigten Zugriffsrechten versehen werden. Die für das Forum verwendete Basissoftware ist „Lotus Quickr“ von IBM, die auch für die "ATHLET User Area" eingesetzt wird und daher eine ähnliche Benutzeroberfläche bietet. Das Benutzerforum wird für externe Anwender freigeschaltet, sobald die neue ATLAS Version V5 freigegeben wird.

Lotus Quickr ist eine Weiterentwicklung von Lotus Quickplace, einer Collaboration Software von IBM. Die Software kann je nach Bedarf auf einer IBM Lotus Notes/Domino oder auf einer IBM WebSphere (WAS) Infrastruktur aufgesetzt werden.

Quickr ermöglicht es, Inhalte gemeinsam zu nutzen, online zusammenzuarbeiten, innerhalb oder außerhalb von Firewalls. Mithilfe von Bibliotheken in Quickr können Dokumente ohne großen Aufwand eingestellt, verwaltet und freigegeben werden. Gemein-Gemeinsames Arbeiten an den Dokumenten ist möglich und eine Versionsverwaltung ist vorhanden.

Man kann durch Anpassen von vordefinierten Komponenten, z. B. Teamkalendern, Diskussionsforen, Blogs, Wikis etc., effizient im Team zusammenarbeiten. Die folgenden 2 Abbildungen zeigen Beispiele für das Arbeiten in der „ATLAS User's Area“.

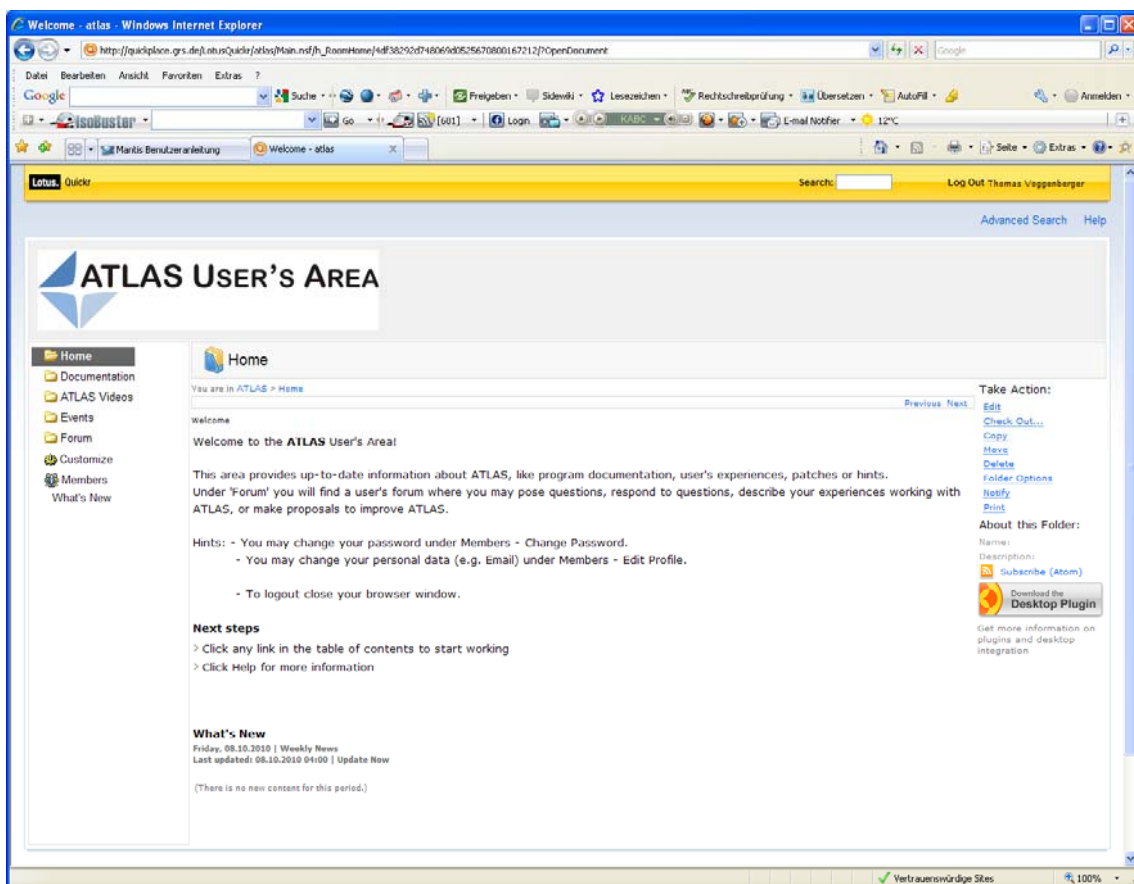


Abb. 5-16 Willkommensseite der ATLAS User' Area

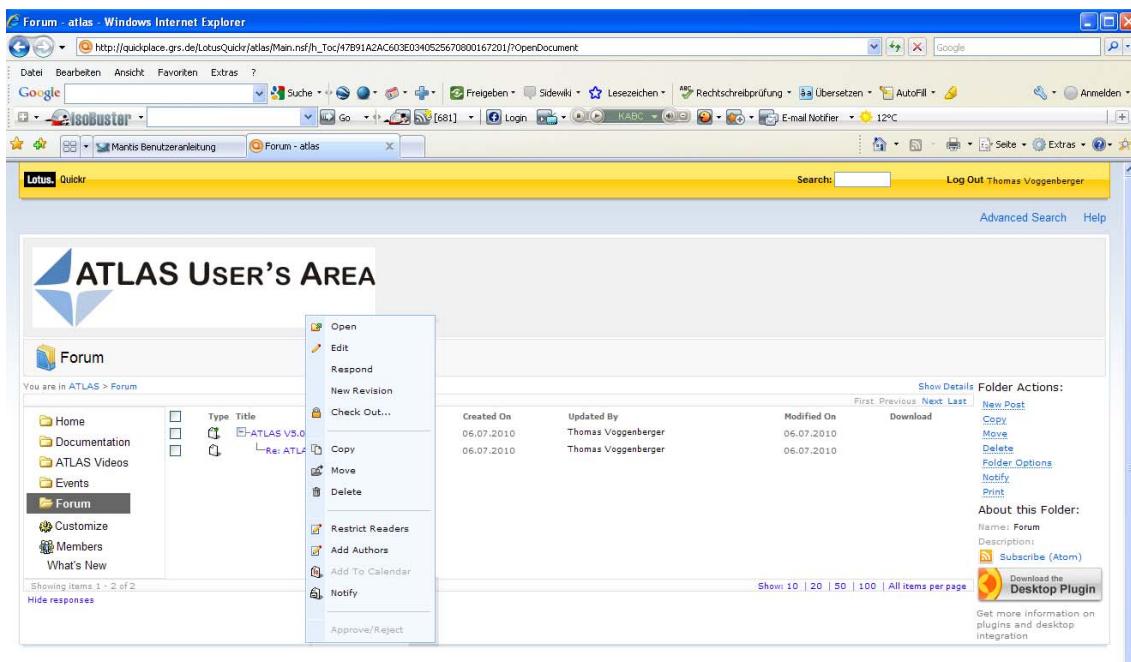


Abb. 5-17 Verwaltung des Benutzer und Beiträge in der ATLAS User' Area

5.5 Qualitätssicherungsmaßnahmen

Die durchgeführten Programmänderungen sind in einem Versionsmanagementsystem dokumentiert und können dort zurückverfolgt werden. Die erkannten Programmfehler werden in einer Fehlerverwaltungssoftware („Bugtrackingsystem“) erfasst und die Bearbeitung und die durchgeführten Maßnahmen werden dort dokumentiert. Das Programmsystem kann mit dem Microsoft WINDOWS Installer einfach dialoggesteuert installiert werden.

5.5.1 Versionsmanagement

In ATLAS wurde schon in den Vorläufer-Versionen ein Versionsmanagementsystem, das „Concurrent Version System (CVS)“ eingesetzt. Dieses System ist zwar bewährt, wird aber als OpenSource nicht mehr weiter entwickelt wird und besitzt einige Schwächen, wie z. B. keine einfache Möglichkeit, Änderungen über mehrere Dateien abzufragen oder Dateien im Projektarchiv umzubenennen oder zu verschieben. Da auch der Server Rechner für die Datenbank der ATLAS Programmdateien (Repository) durch ein LINUX-basiertes System ersetzt werden musste, wurde das CVS durch ein anderes System zur Versionenverwaltung, das „Subversion (SVN)“- System, abgelöst, für das bereits in anderen Entwicklungsprojekten positive Erfahrungen vorlagen.

Dazu wurden zunächst nicht mehr benötigte Programmdateien aus dem Repository entfernt und die Verzeichnisstruktur optimiert und dem aktuellen Entwicklungsstand angepasst. Um weiterhin auf die während der Entwicklung durchgeführten Änderungen (Historie) zugreifen zu können, wurde das bestehende CVS-Repository erfolgreich mit dem Werkzeug „cvs2svn“ nach SVN konvertiert. Für SVN existieren eine Reihe von interaktiven Bedienoberflächen (Clients), mit denen man alle wesentlichen Operationen wie z. B. Ein- und Auschecken oder einen Versionsvergleich durchführen kann. In der gegenwärtigen Entwicklung wird „RapidSVN“ genutzt (Abb. 5-18), das mit seiner eingängigen Bedienoberfläche eine einfache Nutzung von SVN-Funktionen ermöglicht. Im Bild sind für eine Datei des Visualisierungsprogramms die verschiedenen Versionen mit den Änderungszeitpunkten sichtbar.

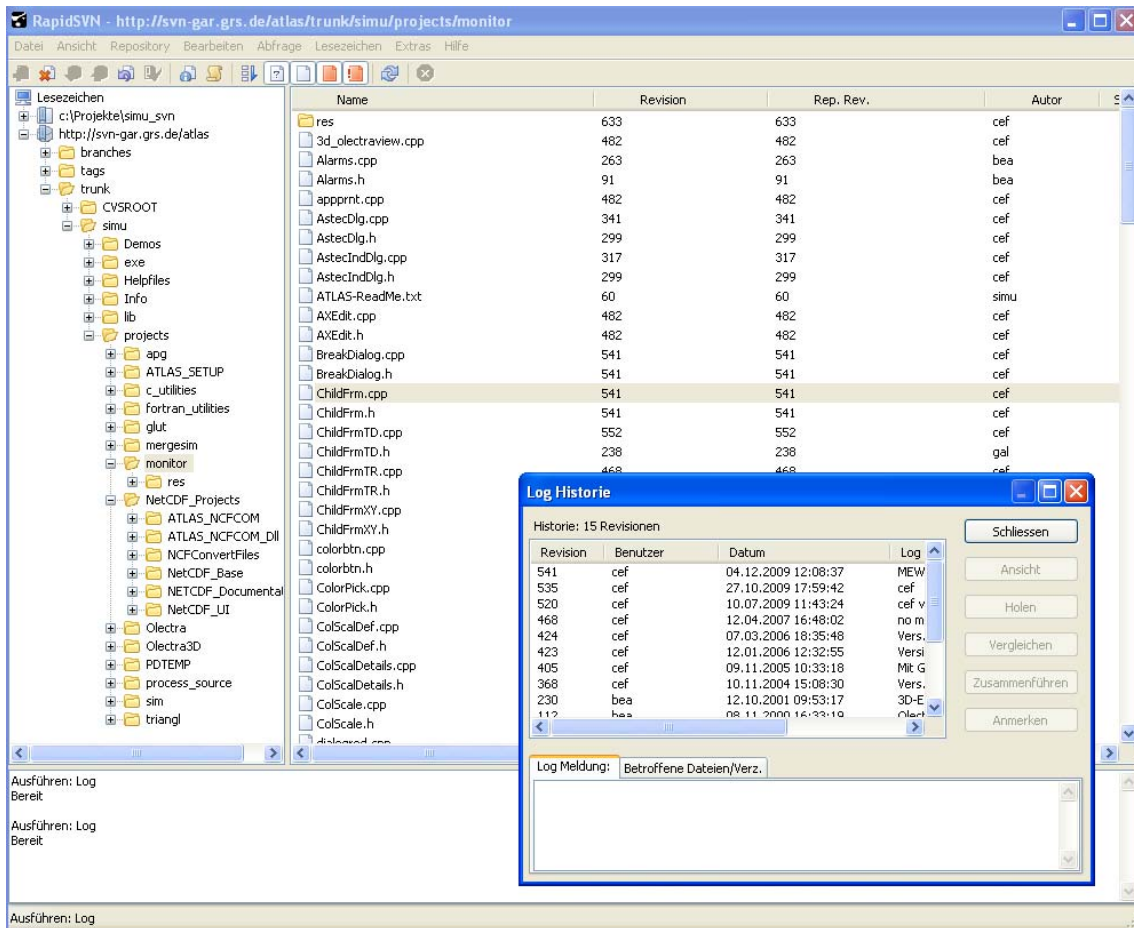


Abb. 5-18 Bedienoberfläche von RapidSVN

5.5.2 Fehlerverfolgung

Für ATLAS wird nun ein System zur Fehlerverfolgung („Bugtracking System“) verwendet, das dazu dient, Programmfehler zu erfassen und zu dokumentieren. Die Realisierung basiert auf der Open-Source Lösung „Mantis“ [WIK 09.3/], einer datenbankgestützten WEB-Anwendung, in dem berechnigte Nutzer beobachtete Programmfehler eintragen können. Dies führt zu einer automatischen Benachrichtigung der Entwickler, die dann die Fehlermeldungen bearbeiten können. Die durchgeführten Maßnahmen werden im System festgehalten und bei Beseitigung eines gemeldeten Fehlers können die Nutzer automatisch benachrichtigt werden. Zurzeit können aus Sicherheitsgründen nur Nutzer innerhalb der GRS auf das System zugreifen. Nach einer Erprobungsphase wird das System auch für externe ATLAS-Anwender freigeschaltet.

5.5.2.1 Überblick

Mantis basiert auf der Skriptsprache PHP und benötigt einen Webserver. Der Benutzer bedient Mantis über eine HTML-Oberfläche. Diese lässt sich über eine Konfigurationsdatei oder in Dialogmasken Benutzer bezogen einstellen. Eine Datenbank (MySQL, MS SQL oder PostgreSQL) wird verwendet, um die Einträge zu verwalten. Mantis ist unter mehreren Betriebssystemen (darunter WINDOWS und LINUX) lauffähig.

In Mantis können verschiedene Projekte angelegt werden, das System ist also für mehrere Entwicklungsprojekte gleichzeitig einsetzbar. Den Projekten können dann Projektteilnehmer mit unterschiedlichen Zugriffsrechten zugeteilt werden. Die Zugriffsrechte sind hierarchisch festgelegt: höhere Zugriffsebenen (z. B. Developer) schließen die Rechte niedrigerer Ebenen (z. B. Reporter) ein. Insgesamt gibt es sechs vorgegebene Zugriffsebenen: Viewer, Reporter, Updater, Developer, Manager und Administrator.

Hat ein Projektteilnehmer mindestens Reporter-Status innerhalb eines Projekts, kann er einen Problembenicht (Issue) anlegen. Jeder Problembenicht befindet sich bei einer gewissen Phase seines Bearbeitungsablaufs in einem bestimmten „Status“ (new, feedback, acknowledged, confirmed, assigned, resolved, closed), der seinen aktuellen Bearbeitungszustand beschreibt. Für Zustandsänderungen bedarf es wiederum entsprechender Zugriffsrechte.

Während des Bearbeitungsablaufs eines Problems können von allen berechtigten Projektteilnehmern Kommentare zum Bericht hinzugefügt werden. Das System fordert bei Zustandswechseln ebenfalls erläuternde Kommentare, so dass der Lebenszyklus eines Berichts nachvollzogen werden kann.

Mantis bietet umfangreiche Filtermöglichkeiten, die dazu beitragen, die Übersicht über anstehende und behobene Fehlerberichte zu behalten, sowie den Überblick über den Gesamtzustand eines Projekts zu gewinnen.

Im Folgenden werden einige wichtige Funktionen und Eingabemasken von Mantis /FUC 10/ näher beschrieben, insbesondere die für einen Standardbenutzer wichtigen.

5.5.2.2 Anmeldung

Die gegenwärtige Installation von Mantis besteht auf einer LINUX-Maschine im Intranet der GRS. Die Anmeldeseite ist über einen beliebigen WEB-Browser (z. B. Internet Explorer) über die Adresse „http://swr-os111a.grs.de/mantis/mantisbt-1.2.0/login_page.php“ erreichbar. Im Anschluss gibt der Anwender im Anmeldedialog (Abb. 5-19) seinen Benutzernamen und das Passwort ein, anschließend klickt er auf die Schaltfläche „Login“.

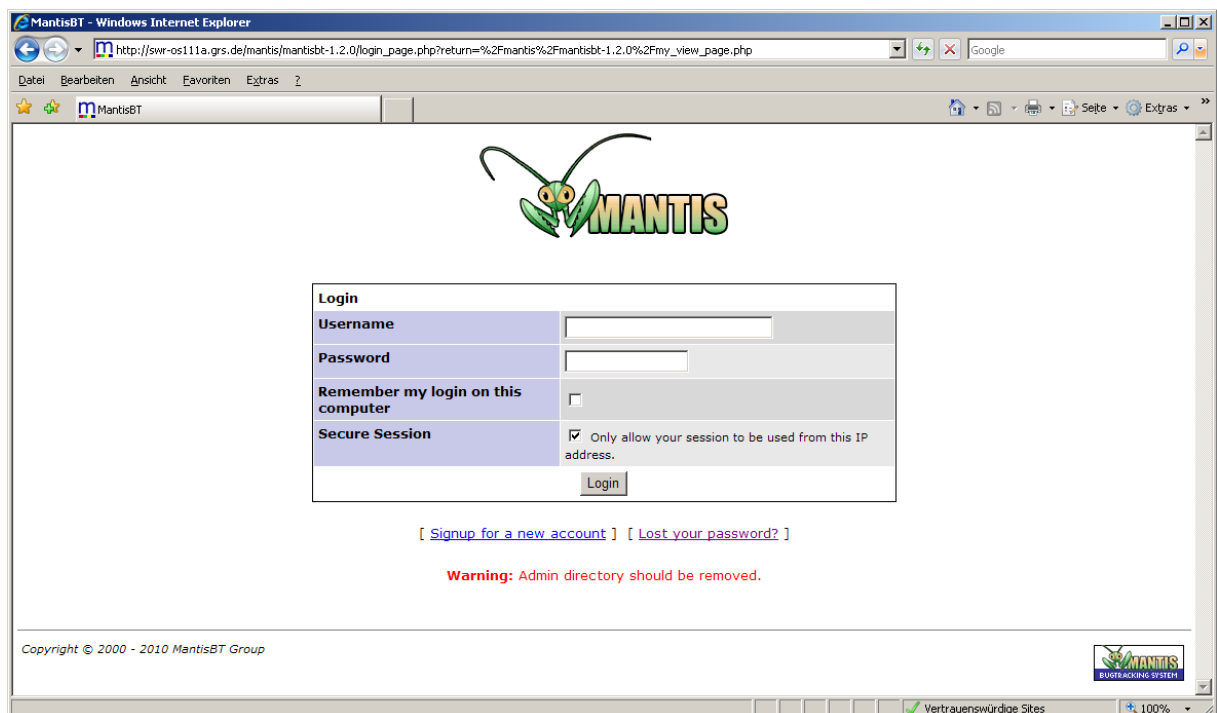


Abb. 5-19 Anmeldeseite von Mantis

Wenn der Benutzer die Option „Remember my login on this computer“ wählt, bleibt er zwischen verschiedenen Browser-Sitzungen angemeldet. Es besteht die Möglichkeit, mittels Email einen neuen „Account“ oder ein neues Passwort anzufordern.

5.5.2.3 Übersicht und grundsätzlicher Aufbau

Nach erfolgter Anmeldung erscheint eine Übersicht zu aktuellen Problembereichten (Abb. 5-20).

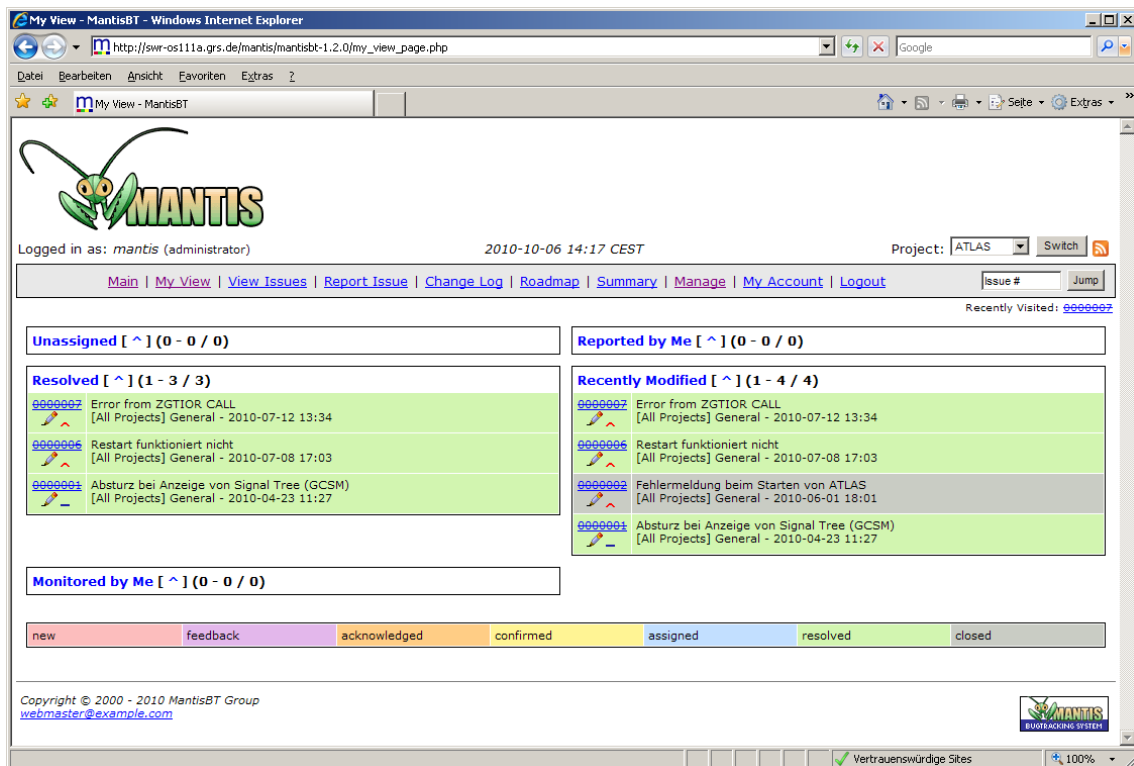


Abb. 5-20 Persönliche Übersichtsseite in Mantis

Mit Hilfe der Auswahlbox oben rechts ist es möglich, das aktuelle Projekt zu wechseln. Auch ist es möglich, sich eine Übersicht über Problembereichte aller Projekte, zu denen man eine Berechtigung hat, zu erhalten.

Die Navigationsleiste im oberen Bereich bietet Zugriff auf alle wesentlichen Funktionen von Mantis:

- **Main (Startseite):**

Diese Seite zeigt Nachrichten vom Projektmanager (Administrator) an.

- **My View (Übersicht):**
Dies ist die Standardseite des Systems (Abb. 5-21).
- **View Issues (Probleme anzeigen)**
Diese Seite zeigt alle Änderungsanforderungen an, die im System gespeichert sind.
- **Report Issue (Probleme eingeben)**
Dient zur Eingabe neuer.
- **My Account (Konto):**
Hier können persönliche Angaben und Einstellungen vorgenommen werden.
- **Logout:**
Damit meldet sich der Benutzer aus Mantis ab.

Im Folgenden werden die einzelnen Bedienfunktionen von Mantis ausführlicher beschrieben.

- **Probleme anzeigen (View Issues)**

Diese Seite (Abb. 5-21) gibt alle Problembenachrichtigungen anhand eines definierten Filters aus. Der Filter kann im oberen Bereich der Seite festgelegt werden. Im unteren Bereich werden dann die passenden Problembenachrichtigungen angezeigt.

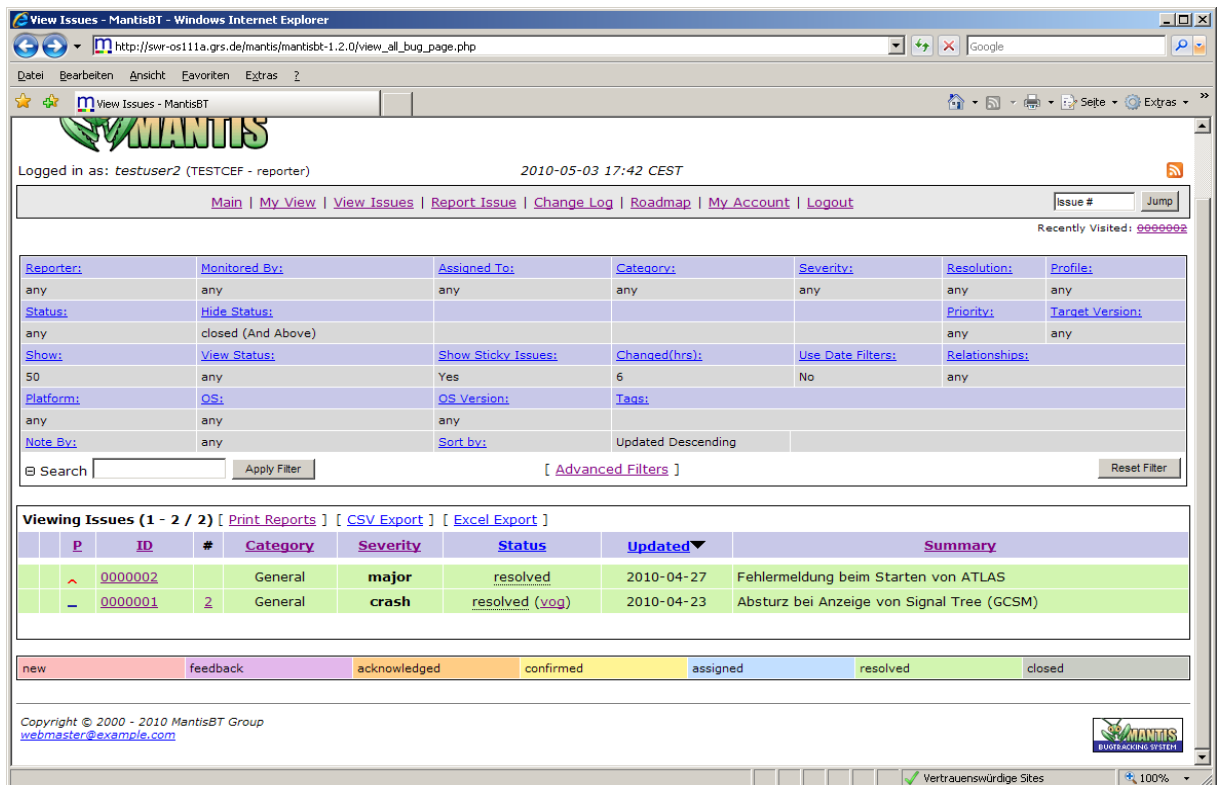


Abb. 5-21 Standardseite des Mantis Systems

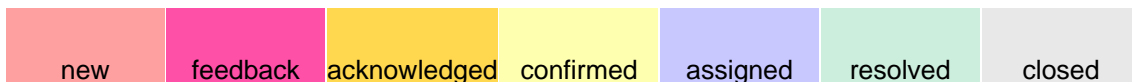
Wenn ein Filterelement angeklickt wird, wird eine Auswahlliste für das dazugehörige Feld eingeblendet. Auf diese Weise können zum Beispiel nur die Problembereiche mit einer bestimmten Auswirkung (Severity) angezeigt werden.

Nach Auswahl des Schalters „Apply Filter“ wird der gewählte Filter angewandt.

Die Problembereiche sind in einer Tabelle mit Attributen gelistet. Ein Klick auf eine Spaltenüberschrift (von der Nummer der Anmerkungen abgesehen) ordnet die Tabelle dieser Spalte entsprechend. Ein zweiter Klick kehrt die Sortierrichtung um.

Mit der Suchfunktion kann man nach einfachen Schlüsselwörtern in den Feldern: „Summary“, „Description“, „Reproducibility“ oder in der ID einer Problembereiche suchen. Diese Felder werden weiter unten detaillierter beschrieben.

Jeder Problembereiche bekommt eine systemweit eindeutige ID. Das bedeutet, dass diese ID auch nicht in anderen Projekten vergeben wird. Zusätzlich wird eine Zusammenfassung (Summary) gezeigt und wann der Datensatz zuletzt aktualisiert wurde (Updated). Weiterhin wird der Status angezeigt. Dieser Status spiegelt sich auch in der Farbe der Zeile wieder. Dabei kommen folgende Färbungen zum Einsatz:



Im Änderungsmanagement kommt dem „*Status*“ eine besondere Bedeutung zu. Wenn ein Problembereich aufgenommen wird, so erhält er den Status „*new*“. Im Normalfall sollte nun der Projektverantwortliche eingreifen und festlegen, ob der Problembereich berücksichtigt wird. Dazu dient der Status „*acknowledged*“. Bei Wechsel des Status kann derjenige, der diesen Wechsel durchführt der Änderungsanforderung eine Notiz beifügen.

Nachdem der Problembereich anerkannt wurde, sollte nun vom Programmentwickler geprüft werden, ob das beschriebene Fehlverhalten zutrifft, das heißt, ob es sich wirklich um einen Fehler handelt. In diesem Fall wechselt der Status nach „*confirmed*“. Nun kann die Problembereich einem Teammitglied zugewiesen (*assigned*) werden. Sobald das gemeldete Problem beseitigt ist, bekommt der Problembereich den Status „*resolved*“. Danach muss noch geprüft werden, ob die Anforderung korrekt umgesetzt wurde. In diesem Fall wechselt der Status auf „*closed*“.

Der Text in der Spalte "Severity" wird in „**Bold**“ angezeigt, wenn die Art als „*major*“ (schwerer Fehler), „*crash*“ (Programm-Absturz) oder „*block*“ (System-Absturz) gekennzeichnet ist und die Anfrage noch nicht gelöst ist.

Eine wichtige Empfehlung ist, niemals geschlossene Problembereiche zu löschen. Zwar ist dies möglich, jedoch oft gehen dadurch wertvolle Informationen verloren. Sinnvoller ist deswegen die Einrichtung eines Filters, mit dem die geschlossenen Problembereiche nicht angezeigt werden. Aus dem gleichen Grund sollten auch so viele Notizen wie möglich mit den Problembereichen verknüpft werden, beispielsweise Protokolldateien bei der Fehlersuche, detaillierte Fehlerbeschreibungen, Ursachen für Fehler, etc. Auch oder besonders wenn etwas nicht umgesetzt wird, sollte es auf diese Weise dokumentiert werden.

- **Probleme detailliert darstellen (Reporter)**

Wird die ID eines Problembereichs ausgewählt, so wird dieser detailliert dargestellt. Abb. 5-22 zeigt die Seite, die für Projektteilnehmer mit „*Reporter*“-Status angezeigt wird. Dabei ist es zu beachten, dass die Bedienung dieser Seite keine Änderung der Attribute der Problembereich zulässt. Dies trifft allerdings nur für den „*Reporter*“ zu.

Dem Administrator hingegen stehen weitere Funktionen zur Verfügung, die ihm die Möglichkeit geben das Problem weitgehend zu bearbeiten, wie nächsten Abschnitt detailliert beschrieben wird.

0000002: Fehlermeldung beim Starten von ATLAS - MantisBT - Windows Internet Explorer

http://swr-os111a.grs.de/mantis/mantisbt-1.2.0/view.php?id=2

0000002: Fehlermeldung beim Starten von ATLAS - M...

Logged in as: cef (F. Cester - developer) 2010-05-03 17:09 CEST

Main | My View | View Issues | Report Issue | Change Log | Roadmap | My Account | Logout

Issue # Jump

Recently Visited: [gooooo](#)

View Issue Details [Jump to Notes] [>>] [Issue History] [Print]

ID	Project	Category	View Status	Date Submitted	Last Update
0000002	ATLAS	[All Projects] General	public	2010-04-26 17:37	2010-04-27 15:42

Reporter: cef

Assigned To:

Priority: high Severity: major Reproducibility: always

Status: resolved Resolution: open

Platform: OS OS Version:

Summary: 0000002: Fehlermeldung beim Starten von ATLAS

Description: Ich habe meine PD- und KEY-Files unter einem sehr langen directory-path angelegt. Das erste Mal ist ATLAS korrekt gelaufen. Nachdem ich die ATLAS-Ikone angeklickt und das Programm erneut gestartet habe, bekam ich folgende Fehlermeldung: 'Can not be opened' gefolgt von einem unvollständigen Verzeichnispfad.

Tags: No tags attached.

Attach Tags: (Separate by ",") Existing tags Attach

Attached Files: Monitor Reopen

Relationships:

Users monitoring this issue

User List: There are no users monitoring this issue. Username Add

Notes: There are no notes attached to this issue.

Issue History

Date Modified	Username	Field	Change
2010-04-26 17:37	cef	New Issue	
2010-04-27 09:02	cef	Severity	crash => major
2010-04-27 09:11	cef	Description Updated	View Revisions
2010-04-27 15:38	cef	Status	new => acknowledged
2010-04-27 15:42	cef	Status	acknowledged => resolved

Copyright © 2000 - 2010 MantisBT Group
webmaster@example.com

Vertrauenswürdige Sites 100%

Abb. 5-22 Problembesicht mit „Reporter“-Status

- **Probleme bearbeiten (Administrator)**

Wird die ID des Problems vom Administrator ausgewählt, so können auch spezielle Aktionen durchgeführt werden. Beispielsweise kann über einen zusätzlich verfügbaren Button „Send a reminder“ eine Erinnerung gezielt an bestimmte Nutzer gesendet werden:

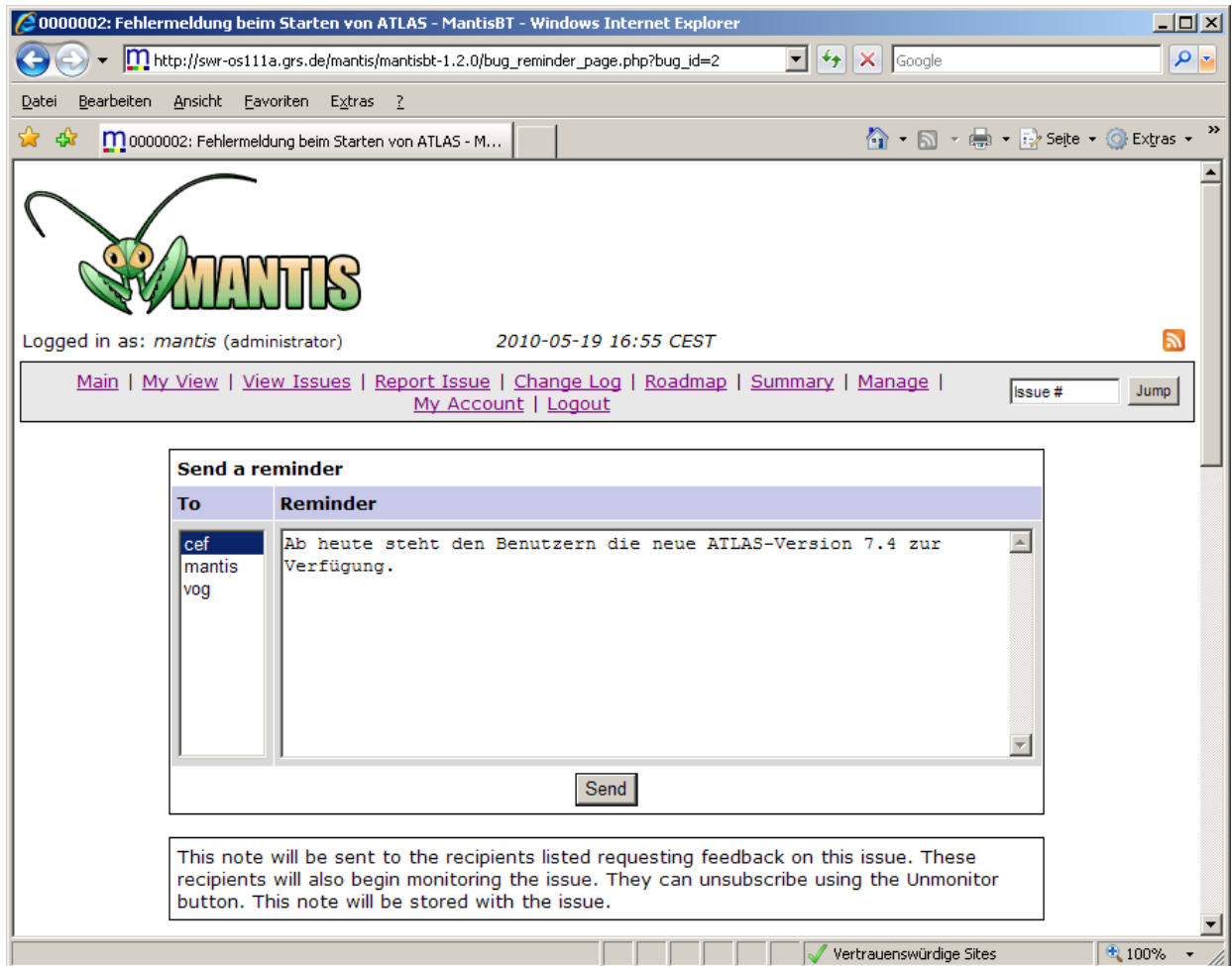


Abb. 5-23 Senden einer Benachrichtigung aus Mantis

Die versendete Nachricht wird dann sowohl als Notiz an den Problembereich gehängt, als auch per Email an die ausgewählten Personen verschickt.

Im unteren Bereich ist es möglich verschiedene Eigenschaften der Problembereich zu editieren bzw. die oben beschriebenen Statuswechsel herbeizuführen. Mit Hilfe der Funktion „Edit“ ist es möglich, alle angezeigten Daten zu ändern. Die Funktion „Assign To:“ weist die Zuständigkeit für die Problembereich an ein auswählbares Teammitglied zu. Um den Status des Problembereichs zu ändern, kann die Funktion „Change Status To:“ verwendet werden. Ein wichtiges und nützliches Feature ist die Beobachtung ei-

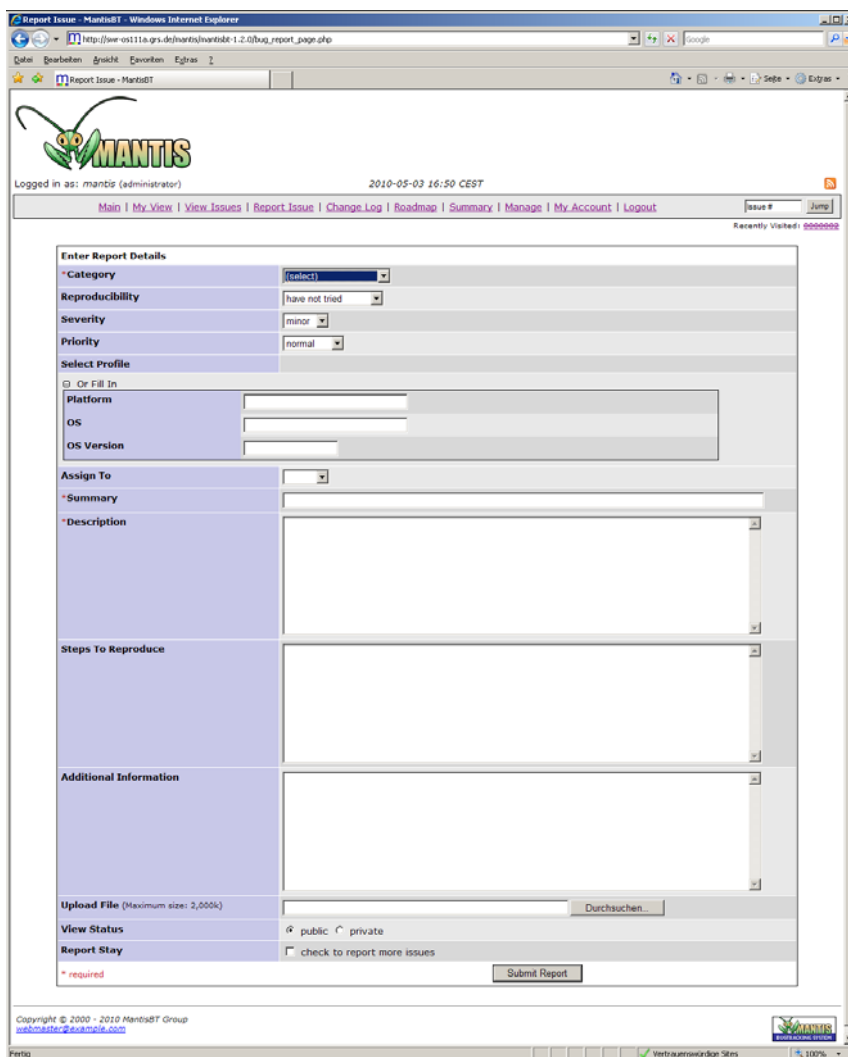
nes Problems mit Hilfe des Schalters „Monitor“.

Sobald eine Änderung an dem Problembereich durchgeführt wird, erhält man automatisch eine Email-Nachricht. Um eine ähnlichen Problembereich zu erzeugen, kann die Funktion „Clone“ verwendet werden. Dadurch werden alle gesetzten Eigenschaften in einen neuen Problembereich kopiert.

Problem berichten (Report Issue)

Neben dem Ändern von Problembereichen ist es natürlich auch möglich, neue Einträge zu generieren. Abb. 5-24 zeigt die Seite für die Berichtseingabe. Zwischen existierenden Problembereichen lässt sich mit Hilfe des Links oben rechts umschalten.

Es folgt die detaillierte Beschreibung der Eingabefelder. Dabei ist es zu beachten, dass die Eingabe in die mit einem roten Stern gekennzeichneten Felder obligatorisch ist.



The screenshot shows the 'Report Issue' page in a MantisBT web application. The browser window title is 'Report Issue - MantisBT - Windows Internet Explorer'. The address bar shows the URL 'http://www-ost11a.gps.de/mantis/mantisbt:1.2.0/bug_report_page.php'. The page header includes the Mantis logo, the user 'mantis (administrator)', and the date '2010-05-03 16:50 CEST'. A navigation menu contains links like 'Main', 'My View', 'View Issues', 'Report Issue', 'Change Log', 'Roadmap', 'Summary', 'Manage', 'My Account', and 'Logout'. The main form is titled 'Enter Report Details' and contains several sections:

- Category:** A dropdown menu with '(select)' as the current value.
- Reproducibility:** A dropdown menu with 'have not tried' as the current value.
- Severity:** A dropdown menu with 'minor' as the current value.
- Priority:** A dropdown menu with 'normal' as the current value.
- Select Profile:** A section with a radio button labeled 'Or Fill In' and three input fields for 'Platform', 'OS', and 'OS Version'.
- Assign To:** A dropdown menu.
- Summary:** A text input field.
- Description:** A large text area with a vertical scrollbar.
- Steps To Reproduce:** A text area with a vertical scrollbar.
- Additional Information:** A text area with a vertical scrollbar.
- Upload File:** A file upload field with a maximum size of 2,000k and a 'Durchsuchen...' button.
- View Status:** Radio buttons for 'public' (selected) and 'private'.
- Report Stay:** A checkbox for 'check to report more issues'.
- Submit Report:** A button at the bottom right of the form.

At the bottom of the page, there is a copyright notice: 'Copyright © 2000 - 2010 MantisBT Group' and a logo for 'MantisBT Group'.

Abb. 5-24 Maske zur Generierung eines Problembereichs

Category

Gibt die Kategorie (Zusammenhang) an, in der der Fehler auftritt, kann z. B. Dokumentation, Installationsroutine oder etwas ähnliches sein (vom Benutzer definierbarer Wert).

Reproducibility

Gibt an, ob und wie der Fehler reproduzierbar ist. Die folgenden Werte sind möglich:

Always, sometimes, random, have not tried, unable to reproduce, N/A (not applicable/available)

Severity

Gibt die Schwere des Fehlers an und kann einen der folgenden Werte annehmen:

feature (Änderungswunsch), trivial, text (Fehler im Text), tweak (Unschönheit), minor (kleinerer Fehler), major (schwerer Fehler), crash (Absturz), block (das Programm reagiert nicht mehr bzw. lässt sich nicht weiter bedienen)

Priority

Die Priorität gibt an, wie wichtig eine Bearbeitung des Problems ist. Die folgenden Werte sind möglich:

none, low, normal, high, urgent, immediate

Select Profile

Mit Hilfe des Profils kann das eigene System angegeben werden. Sollten keine Profile vorhanden sein, so ist es möglich, das System in den Einstellungen aufzunehmen oder mit Hilfe der Felder „*Platform*“, „*OS*“ (*Betriebssystem*) und „*OS Version*“ manuelle Angaben zu machen.

Assign to

In diesem Feld kann ein Team-Mitglied ausgewählt werden, welches die Problembereicht bearbeiten soll (nur mit Administratorrechten verfügbar).

Description

In der Beschreibung sollte ausgeführt werden, um was es in dem Problembereicht geht, also welche Änderungen durchgeführt werden sollen. Eine gute Problembeschreibung und viele Informationen über einen Fehler, können die Beseitigung erheblich erleichtern.

Hinweis: Eine Raute gefolgt von der ID einer anderen Anfrage (z. B. #123) erzeugt automatisch einen Link zu dieser Anfrage

Steps To Reproduce

Wenn es sich bei dem Problembereicht um einen Fehler handelt, sollte neben der Beschreibung des Fehlers auch angegeben werden, wie der Fehler reproduziert werden kann. Dies ist wichtig für das Auffinden des Fehlers.

Additional Information

Dieses Feld bietet sich beispielsweise für Fehlermeldungen, bei denen der User mit der Beschreibung eines bestimmten Programmverhaltens einen zusätzlichen Beitrag zur Fehlerfindung leisten möchte.

Upload File

Dem Problembereicht kann auch eine Datei beigefügt werden. Die maximale Größe hängt von der Systemeinstellung ab. Wenn nötig können jedoch weitere Anhänge später über die Bearbeitung der Problembereicht hinzugefügt werden.

View Status

Mit Hilfe des Anzeigestatus kann angegeben werden, ob die Problembereicht öffentlich einsehbar ist oder nur von Team-Mitgliedern.

Report Stay

Wenn die Auswahl markiert ist, dann wird nach dem Absenden der Problembereich („Submit“) eine neue Eingabemaske geöffnet, bei der die Felder *Category*, *Reproducibility*, *Severity*, *Priority* und *Profil* mit den gleichen Werten wie im aktuellen Problembereich vorgelegt sind.

5.6 Persönliche Einstellungen (My Account)

Mit Hilfe der folgenden Seiten kann Mantis individuell auf die eigenen Bedürfnisse angepasst werden.

Die erste Seite (Abb. 5-25) enthält dabei die eigentlichen Konteninformationen zu dem derzeit eingeloggten Benutzer. Hier kann das Passwort, die Email-Adresse, an die Benachrichtigungen geschickt werden sollen und der Name angegeben werden.

Weitere Anpassungen sind über „*Preferences*“ und „*Profiles*“ erreichbar. Bei der ersten handelt es sich um die Möglichkeit, die Einstellungen für die eigenen Mantis-Zugriffe zu ändern. Die zweite Option ermöglicht die Verwaltung des eigenen Profils.

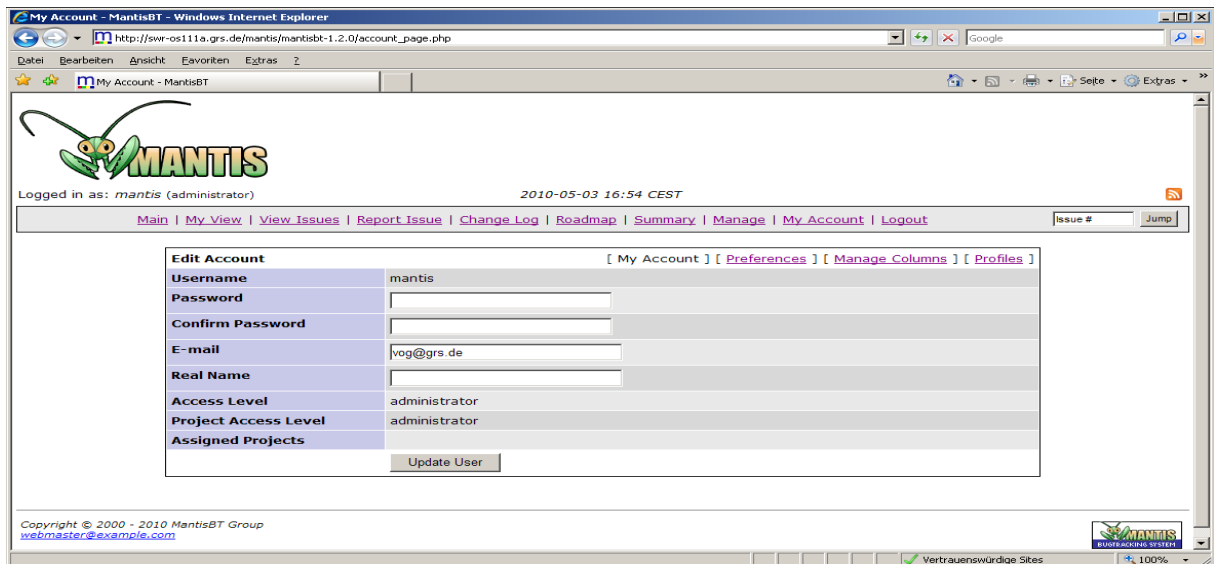


Abb. 5-25 Persönliche Daten in Mantis verwalten

Einstellungen ändern (Preferences)

Mit diesem Dialog (Abb. 5-26) ist es möglich, das Verhalten von Mantis den eigenen Bedürfnissen anzupassen, beispielsweise, wann und in welchen Fällen Mantis Email-

Benachrichtigungen senden soll, ob überhaupt Emails versendet werden sollen und ob standardmäßig ein einfacher oder erweiterter Bericht verwendet werden soll.

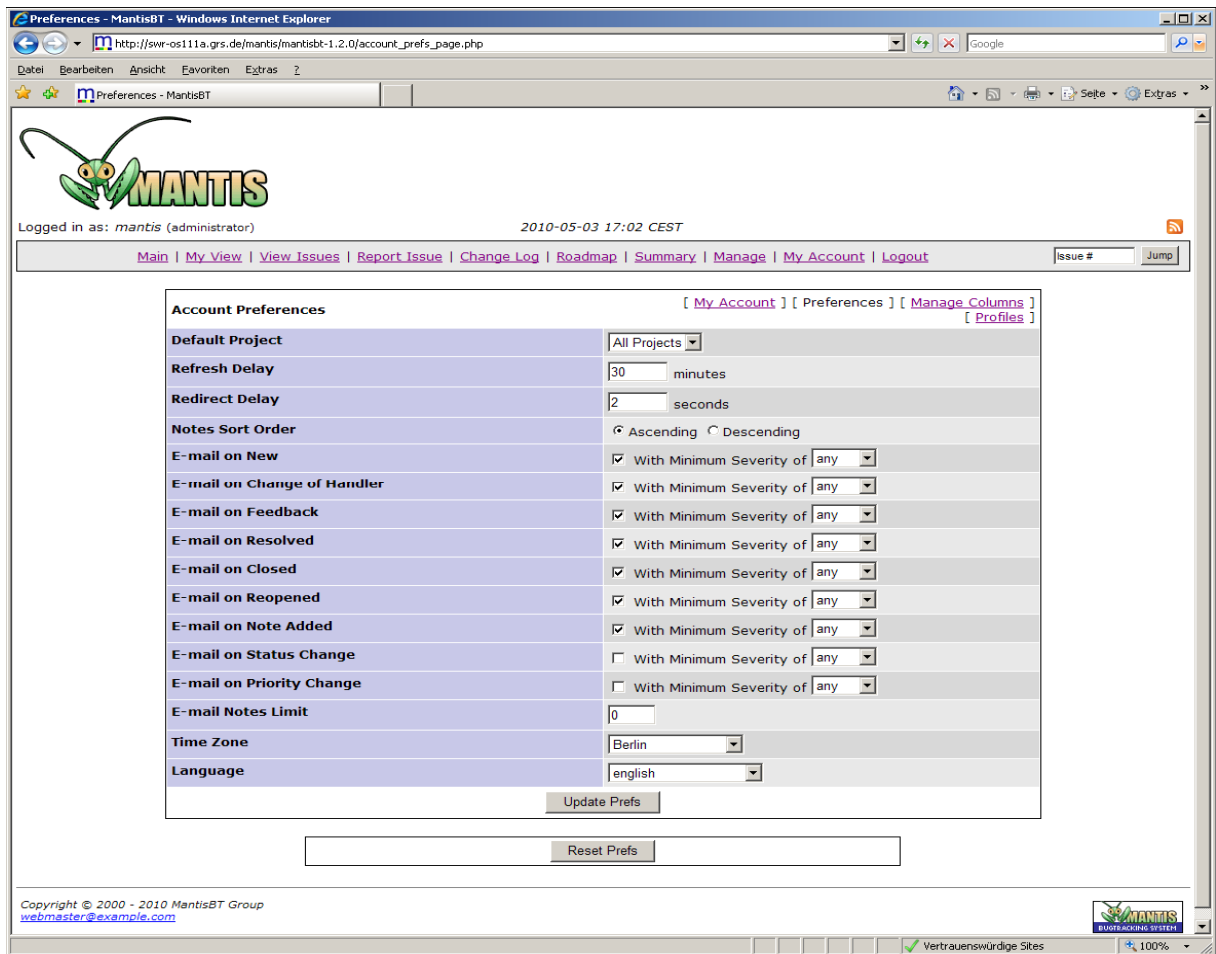


Abb. 5-26 Benachrichtigungen in Mantis verwalten

Profil-Verwaltung (Profiles)

In der Profil-Verwaltung (Abb. 5-27) ist es möglich, Rechner- und Betriebssystem-Konfigurationen zu hinterlegen. Diese werden bei der Aufnahme neuer Problembereiche (siehe Probleme eingeben bzw. Profil wählen) verwendet.

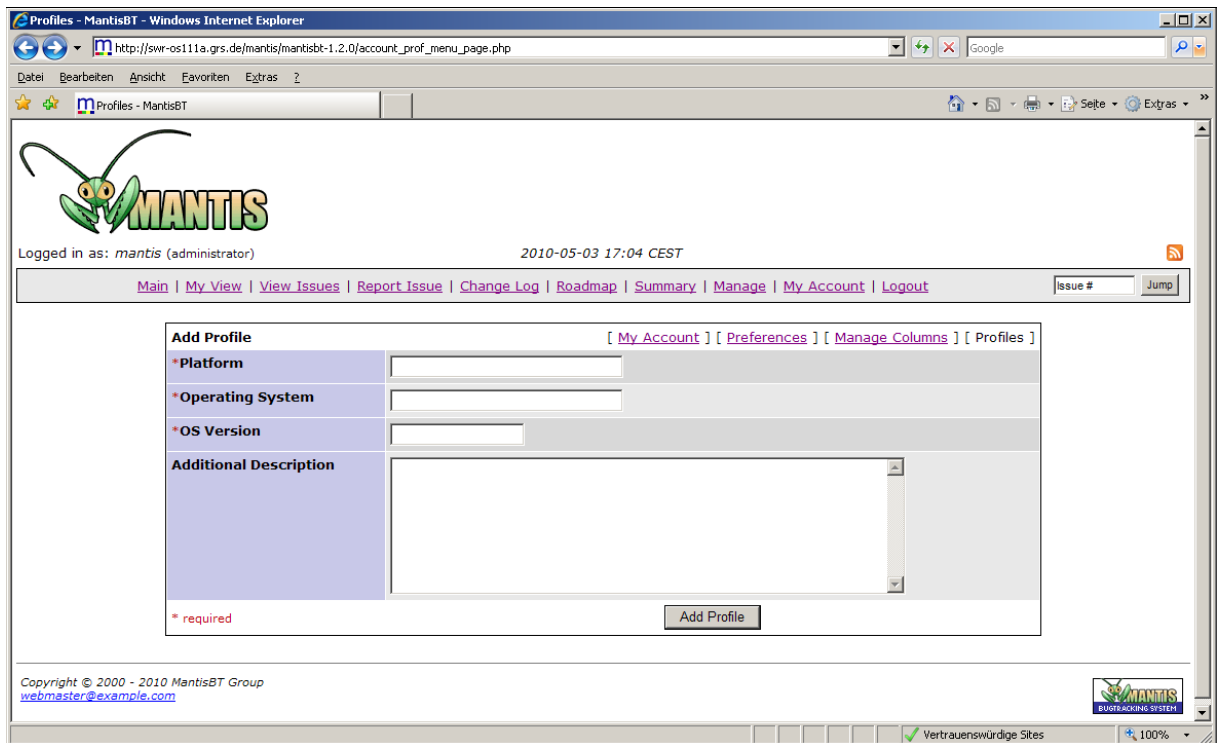


Abb. 5-27 Computer-Profil in Mantis verwalten

Eine Abmeldung vom Mantis System ist jederzeit über den Logout-Reiter möglich. Durch die Abmeldung wird die Session beendet und der Benutzer auf die Anmeldeseite zurückgeleitet.

5.6.1 Installation

Das Installationsprogramm für die ATLAS Software musste vollständig erneuert werden. Dies war aus verschiedenen Gründen notwendig und vorteilhaft. Bisher wurde zu Erzeugung des Installationsprogramms eine ältere Version der Flexera Software „InstallShield“ eingesetzt. Damit war eine Installation unter den 64-bit-Versionen von WINDOWS nicht mehr möglich. Ein Upgrade auf eine aktuelle Version wäre mit hohen Kosten verbunden gewesen. Ebenfalls besteht der Nachteil, dass die Bedienung dieser Software relativ komplex ist und einen hohen Einarbeitungsaufwand erfordert.

Das neue Installationspaket basiert auf dem „Windows Installer“ vom Microsoft, das als der Teil der Entwicklungsumgebung „VisualStudio“ kostenlos verfügbar ist. Es besteht aus einem Windows-Systemdienst, welches Paketdateien im msi-Format (Microsoft Software Installation) interpretieren, entpacken und ausführen kann. Ein Vorteil bei der Verwendung der Windows-Installer-Technologie ist die Reparaturfähigkeit. Der

Windows-Installer-Dienst überwacht beim Starten eines durch ihn installierten Programmes, ob alle zugehörigen Komponenten vorhanden sind. Fehlt eine dieser Komponenten, versucht der Windows Installer, diese nach zu installieren. Dieser Prozess lässt sich auch auslösen, wenn man die Setup-Routine erneut ausführt und gefragt wird, ob man die Anwendung reparieren oder entfernen möchte. Im Allgemeinen ist vor einer Neuinstallation eine Deinstallation von ATLAS über die Standard WINDOWS Softwareverwaltung nötig und wird explizit vom Setup-Programm verlangt.

Für die neue ATLAS Version wurde in Visual Studio eine neues „Setup“-Projekt erzeugt (Abb. 5-28).

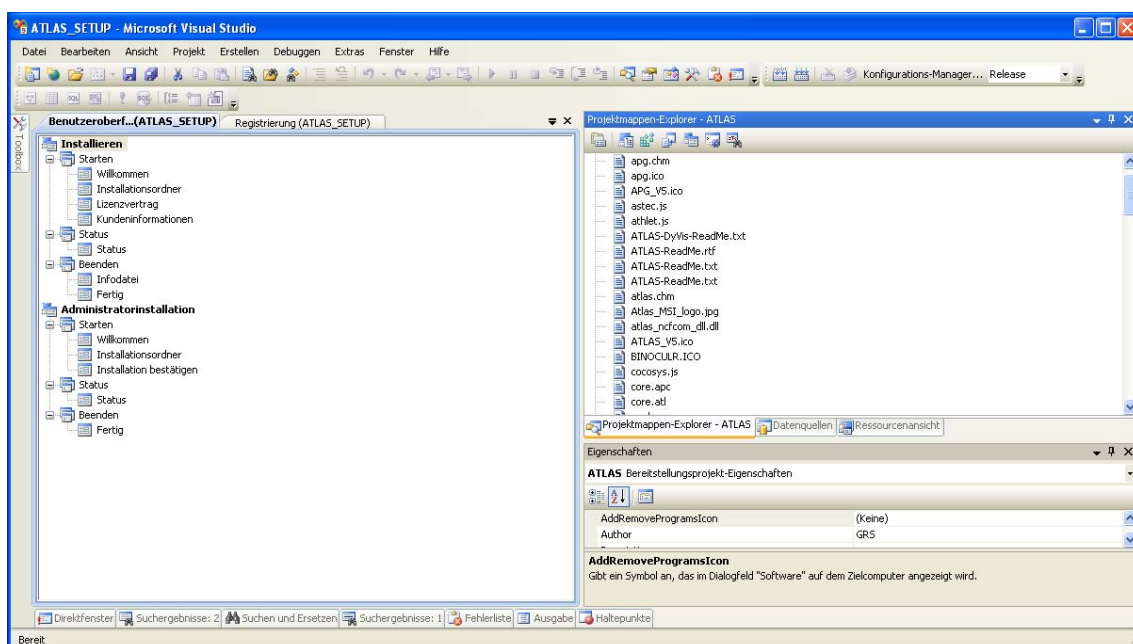


Abb. 5-28 ATLAS Setup-Projekt

In diesem Setup-Projekt werden alle Dateien und Verzeichnisse für eine ATLAS Installation referenziert. Es wurden Dialoge für den Willkommensseite (Abb. 5-29), den Installationspfad, eine Passwordeingabe und eine ATLAS Kurzinformation entwickelt. Aus allen Projektdateien und Einstellungen wurde dann die „ATLAS_SETUP.msi“ - Datei für die Installation generiert. Es wurde in Tests auf verschiedenen PCs nachgewiesen, dass ATLAS mit diesem Installationsprogramm auf allen aktuellen WINDOWS-Versionen problemlos installierbar ist.

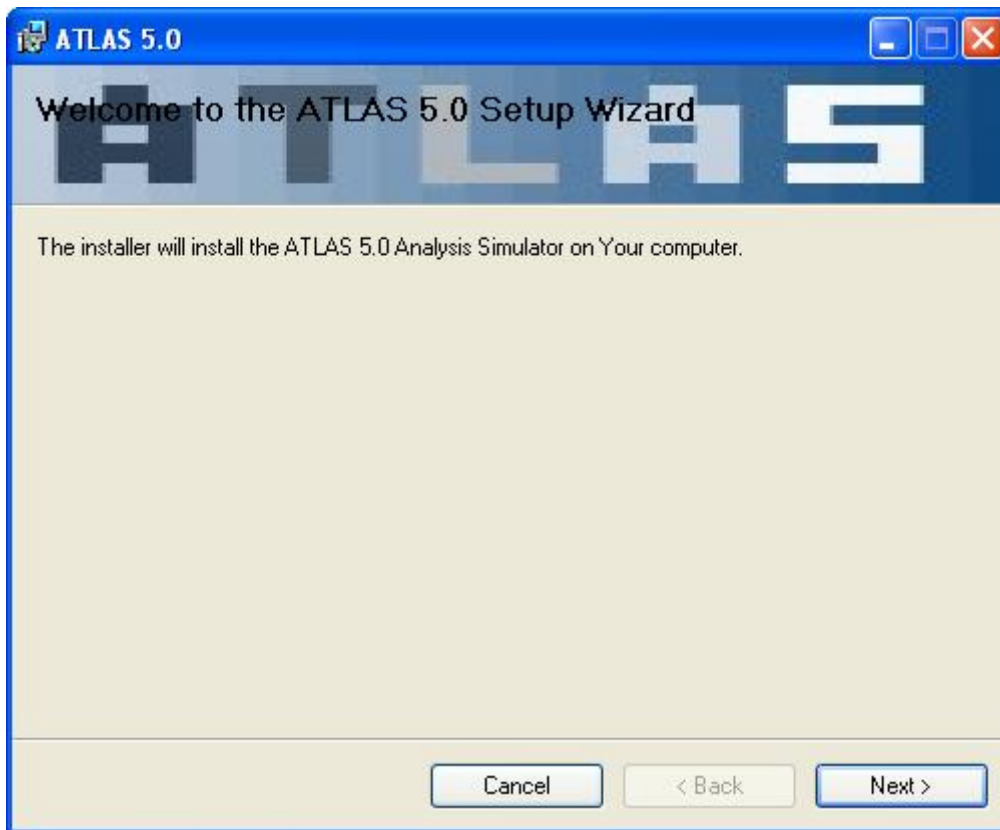


Abb. 5-29 ATLAS Setup Welcome Dialog

6 Zusammenfassung und Ausblick

6.1 ATLAS Übersicht

Der Analysesimulator ATLAS wird als eine interaktive Simulationsumgebung zur Visualisierung und interaktiven Steuerung dynamischer Rechenmodelle entwickelt. Das grundlegende Ziel der ATLAS Entwicklung war es, für diese Modelle eine einheitliche Simulationsumgebung zur Verfügung zu stellen, in der die Ergebnisdaten anschaulich grafisch dargestellt werden und der Simulationsablauf interaktiv beeinflusst werden kann.

ATLAS wird gegenwärtig von Forschungs- und Aufsichtsorganisationen national und international, z. B. in der Störfallanalyse mit Analysesimulatoren, bei der Beurteilung von Bildergonomie, zur Bewertung von Bedienprozeduren und zum Postprozessing von Systemcodes.

Zur Abdeckung des weiten Simulationsspektrums von Stör- und Unfallabläufen im Kühlkreislauf und im Sicherheitsbehälter von KKW sind die GRS Programme ATHLET und COCOSYS in ATLAS gekoppelt. ATHLET enthält Modelle zur Simulation des Kühlkreislaufs, der Reaktorleittechnik, der Neutronendynamik und zur Kernzerstörung, in COCOSYS werden die Thermodynamik des Reaktorcontainments, das Aerosol- und Spaltproduktverhalten, Pyrolyse und Wasserstoff-Deflagration simuliert. Für die Gesamtsimulation schwerer Störfälle können in ATLAS die Integralcodes MELCOR und ASTEC eingesetzt werden. Die Schnittstelle zu ATLAS ist modular aufgebaut, so dass bei Bedarf weitere Modelle integriert werden können.

Mit der fortschreitenden Entwicklung der Simulationscodes in Richtung immer detaillierter Modelle und höherer örtlicher Auflösungen, z. B. mehrdimensionale Simulation des Reaktorkerns, ergaben sich auch für ATLAS erhöhte Anforderungen in der Visualisierung örtlicher Daten und bei der Handhabung des hohen Datenaufkommens.

6.2 Datenmanagement

In ATLAS besteht die grundsätzliche Anforderung, aktuelle oder zurückliegende Simulationsdaten zur Verfügung zu stellen. Zusätzlich müssen interaktive Steuerbefehle an das Simulationsprogramm übertragen werden. Zur Umsetzung wurde ein Datei basier-

tes Verfahren entwickelt, das die Daten in einem standardisierten Format, dem „Network Common Data Form (NetCDF)“ speichert. Die benötigten Informationen sollen in Dateien mit diesem Format bereitgestellt werden. Daher wurden Schnittstellenprogramme entwickelt, die solche Dateien anlegen oder öffnen können und die benötigten Datenstrukturen definieren oder auslesen können. Weitere Programme sind für das Ein- und Auslesen der Beschreibungen für die Variablen („Keywordliste“), die Zeitvektoren der Simulationsergebnisse und die Steuerbefehle in diese Dateien entwickelt worden. Diese Programme können sowohl zur Laufzeit der Simulation als auch durch eine Konversion der bestehenden Ergebnisdateien aus vorliegenden Simulationsläufen verwendet werden. Es ist eine separate Bedienoberfläche für WINDOWS realisiert, mit der bestehende Simulationsergebnisse für alle in ATLAS vorhandenen Modelle in das neue Dateiformat überführt werden können. In Abb. 6-1 werden die wesentlichen ATLAS Module mit dem notwendigen Datenaustausch über die NetCDF-dateischnittstelle gezeigt.

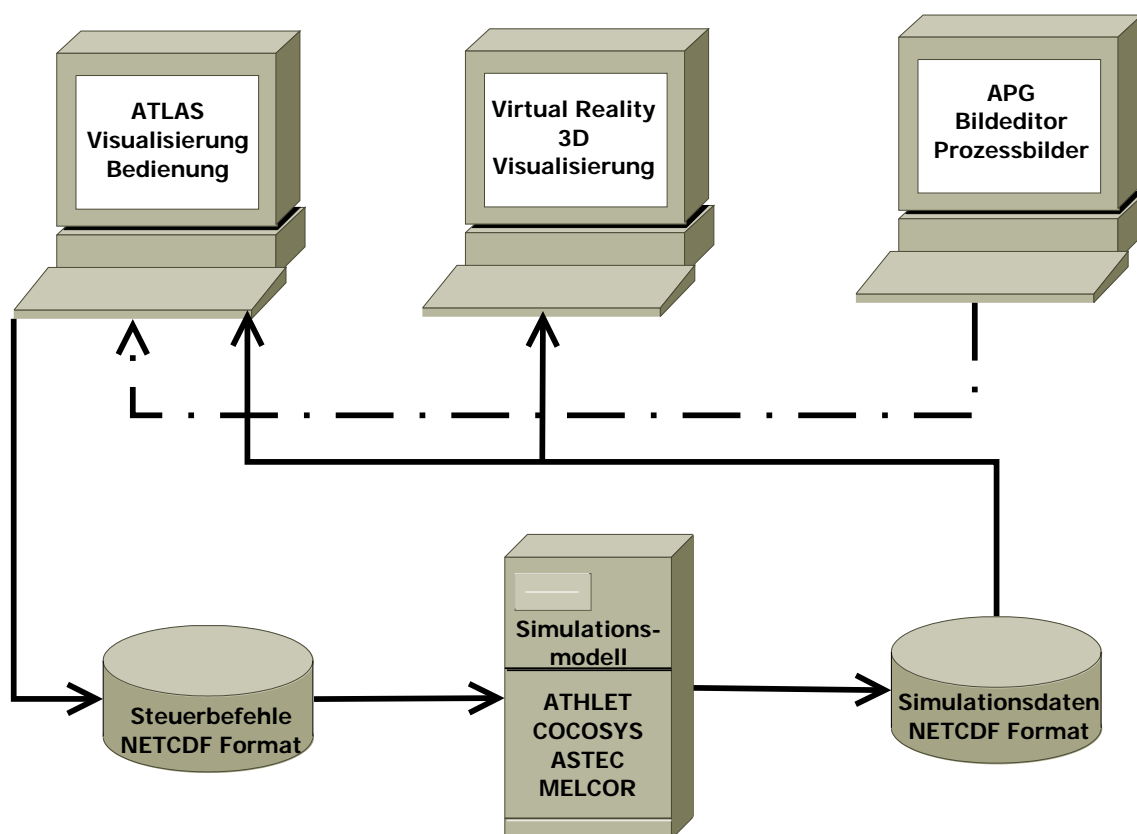


Abb. 6-1 Kopplung der ATLAS Module

Die neue Datenschnittstelle für gespeicherte Daten ist für alle in ATLAS unterstützten Modelle, ATHLET, COCOSYS, ASTEC und MELCOR anwendbar und getestet. Die Ergebnisdarstellung wurde mit dem bisher verwendeten Verfahren verglichen. Gegen-

über der alten Lösung konnte eine Verbesserung bei der Zeitkurvendarstellung, d. h. eine geringere Glättung, erreicht werden. Hinsichtlich der Geschwindigkeit ist die neue Lösung vergleichbar, bei großen Zeitfenstern und vielen Datenpunkten deutlich schneller.

Zur langfristigen Verwaltung von Versuchs- und Simulationsdaten wird Microsoft Office SharePoint Server (MOSS), eine Portal-Softwarelösung von Microsoft, eingesetzt. Mit dieser wurde ein Datenportal für Versuchs- und Simulationsdaten erstellt, in dem Metadaten, wie beispielsweise Anlage, Modellierung, Störfalltyp und Referenzen auf den Speicherort der Nutzdaten, abgelegt werden können. Diese Daten können über speziell entwickelte Eingabeformulare eingetragen oder angezeigt werden. Eine gezielte Volltextsuche mit logischen Begriffsverknüpfungen wird über Suchmasken ebenfalls unterstützt.

Die eigentlichen Nutzdaten, wie Rechenergebnisse, Datensätze, Geometriedaten werden getrennt auf einem speziellen Fileserver des Herstellers NetApp gespeichert, da Sharepoint für sehr große Dateien nicht optimal geeignet ist. Von besonderer Bedeutung ist die Möglichkeit, ein virtuelles Laufwerk unabhängig von physikalischen Gegebenheiten zu definieren. Der Zugriff auf dieses Laufwerk erfolgt über einen virtuellen Hostnamen, dessen freigegebene Verzeichnisse sowohl unter verschiedenen Betriebssystemen als auch über WEB-Interfaces ansprechbar sind. Die Datensicherung erfolgt in regelmäßigen Zeitabständen mit einem Magnetband basierten Backup System und dem "Tivoli Storage Manager" von IBM.

6.3 Visualisierung

Das Visualisierungsmodul von ATLAS ist für das Betriebssystem WINDOWS in den aktuellen Versionen XP, Vista und 7 verfügbar. Das Programmsystem ist als standardisierte Windowsanwendung mit den Entwicklungswerkzeugen von Microsoft realisiert. Die Prozessdiagramme werden mit der Grafikbibliothek OpenGL erzeugt. Die einzelnen Bilder bestehen aus Grundobjekten, beispielsweise Ventile, Pumpen und Leitungen, die wiederum aus einfachen Grafikelementen (Linien, Flächen, Texte) zusammengesetzt sind. Den Elementen können grafische Eigenschaften wie Farbe, Transparenz, Textur, Größe oder Sichtbarkeit zugeordnet werden. Diese Eigenschaften können mit Hilfe spezieller Dynamikdefinitionen mit den Simulationswerten verknüpft und verändert werden. Die Erstellung der Bilder wird durch einen interaktiven

Grafikeditor APG unterstützt. Die Grafikelemente können in einer Werkzeugleiste ausgewählt werden und frei in der Zeichenebene platziert werden. Zur Definition der Eigenschaften und der dynamischen Effekte sind interaktive Werkzeuge integriert.

Das Programm zur Eingabedarstellung von ASTEC Daten der Thermofluidobjekte wurde mit Optionen zur Anzeige von Brenn- und Steuerstäben erweitert. Zurzeit können Temperatur und Masse mit veränderbaren Farbcodierungen angezeigt werden. Neben der Färbung ist auch die Geometrie der Stabdarstellung veränderlich. Damit kann in den einzelnen axialen Zonen die Ausdehnung bzw. das Abschmelzen der Brennstäbe während der Aufheizphase visualisiert werden.

Die radialen Daten des Reaktorkerns werden mit farbcodierten Querschnitten für die die Kühlkanäle angezeigt. Die zugrundeliegenden Farbskalen können ausgewählt und in den Wertebereichen angepasst werden. Für jeden Kernkanal können der Name des Kanals, der Zeittrend der aktuellen Variablen und die axiale Verteilung dargestellt werden.

Die Anwender von ATHLET und ASTEC können also mit geringem Aufwand, fast automatisch, schematische Darstellungen der simulierten Thermohydraulikobjekte, Wärmeleitobjekte, der Brenn- und Steuerstäbe und des Leittechnikmodells GCSM erzeugen und mit den dynamischen Ergebnisdaten verbinden. Die Analytiker erhalten damit einen sehr guten Überblick über zusammenhängende, wichtige Prozessgrößen.

Die Visualisierung in ATLAS wurde mit einem neuen Modul ergänzt, der realitätsnahe Darstellungen von dynamischen Daten in virtueller 3D-Umgebung ermöglicht.

Ein statisches 3D-Modell für ein SWR-Reaktorgebäude in "Google Sketchup" wurde erstellt. Entsprechend vorliegender Schnitt- und Grundriss-Darstellungen in Papierplänen wurden 3D-Darstellungen für verschiedene Höhenebenen des Reaktorgebäudes erstellt. Wichtige Großkomponenten wie Behälter sind teilweise in den einzelnen Betriebsräumen vereinfacht vorhanden, das Reaktorcontainment mit dem Sicherheitsbehälter und Kondensationskammer und Einbauten wie Reaktordruckgefäß mit den anschließenden Leitungen sind detaillierter ausgeführt. Die 3D-Modelle sind sowohl in einer Gesamtsicht des Reaktorgebäudes als auch in Einzelmodellen der verschiedenen Stockwerke und des Sicherheitsbehälters verfügbar.

Das 3D-Modell einer Ebene des Reaktorgebäudes wurde mit dem Datenaustauschformat COLLADA in das Virtual Reality-Programm Vizard importiert. Die Navigation durch die verschiedenen Einzelräume und die Einstellung von Betrachtungsposition ist mit Tastatur- und Maussteuerung realisiert. Die Daten einer Simulationsrechnung mit dem Containment-Systemcode COCOSYS für einen Leckstörfall wurden als Grundlage für die Visualisierung der dynamischen Daten benutzt. Entsprechend der in ATLAS benutzten Farbcodierung wurden in Vizard Farbskalen für Druck und Temperatur definiert. Damit wurden die Wände ausgewählter Räume im Reaktorgebäude eingefärbt. Mit einer kontinuierlichen Einspielung der Ergebnisdateien in Vizard kann dort die zeitliche Veränderung der Daten visualisiert werden. Die Anzeigegeschwindigkeit der Bildupdates ist zufriedenstellend. Die qualitative Übereinstimmung mit der Visualisierung in ATLAS konnte gezeigt werden.

Mit dieser Art der Darstellung kann der Benutzer die Daten wesentlich leichter im Raum zuordnen und erhält durch die flexiblen Ansichtsmöglichkeiten der VR-Software auch eine bessere Gesamtsicht auf die lokalen Daten in den verschiedenen Räumen.

Der Visualisierungsmodul und der Bildeditor von ATLAS wurden in vielen Details überarbeitet. So werden beispielsweise die Darstellung von Zeitdiagrammen optimiert, alle Benutzereinstellungen an zentraler Stelle gespeichert, die interaktive Kontexthilfe konsequent eingesetzt und die Bilderstellung mit dynamischen Farbänderungen besser unterstützt.

6.4 Qualitätssicherung

Für ATLAS wird nun ein System zur Fehlerverfolgung („Bugtracking System“) verwendet, das dazu dient, Programmfehler zu erfassen und zu dokumentieren. Die Realisierung basiert auf der Open-Source Lösung „Mantis“, einer datenbankgestützten WEB-Anwendung, in dem berechtigte Nutzer beobachtete Programmfehler eintragen können. Im neuen User-Forum für ATLAS ist der Zugriff auf relevante Informationen wie Veröffentlichungen, Benutzerhandbücher und aktuelle Ankündigungen möglich sowie eine Diskussionsplattform zu Problemen und Erfahrungen mit der ATLAS Software vorhanden. Die für das Forum verwendete Basissoftware ist „Lotus Quickr“ von IBM, die auch für die "ATHLET User Area" eingesetzt wird und daher eine ähnliche Benutzeroberfläche bietet. Unabhängig von einem direkten Kontakt zu den Entwicklern können Probleme kommuniziert und dokumentiert werden. Ebenfalls wird der

Erfahrungsaustausch zwischen den verschiedenen Nutzern gefördert. Nicht zuletzt wird der Aufwand zur Verbreitung aktueller Informationen zu ATLAS und die Bereitstellung neuer Versionen spürbar verringert.

Die Benutzerhandbücher für ATLAS sind in konsistenten Online- und Druckversionen als Microsoft Compiled Help Module (CHM) und Adobe Portable Document Format (PDF) in englischer Sprache verfügbar. Es existieren getrennte Manuals zur Bedienung der ATLAS Visualisierung und zum Bildeditor APG.

Das Installationsprogramm wurde vollständig erneuert und basiert auf dem „Windows Installer“ von Microsoft. Die neue ATLAS Version V5 ist mit diesem Installationspaket auf allen aktuellen WINDOWS-Versionen problemlos installierbar.

Als wesentliches Ergebnis der Programmpflege ist die einfachere Handhabung von ATLAS für die Anwender herauszustellen. Einerseits ist die Erstellung von dynamischen Bildern durch die Verbesserungen im Bildeditor APG erleichtert, andererseits wird, besonders fremdsprachigen Nutzern, durch die aktuellen englischen Handbücher eine umfassende Dokumentation zur praktischen Anwendung der ATLAS Software zur Verfügung gestellt.

6.5 Grenzen und neue Anforderungen

ATLAS hat mit Abschluss der im Vorhaben durchgeführten Arbeiten einen Entwicklungsstand erreicht, der im Bereich des Postprozessings der Systemcodes als ausgereift bezeichnet werden kann. Der Bereich der Eingabeerstellung, also das Präprozessing, wird von ATLAS bisher nur in geringem Umfang unterstützt. In den USA wird beispielsweise seit einigen Jahren für wichtige Sicherheitscodes, wie TRACE, MELCOR oder RELAP5 im Auftrag der US-NRC die SNAP-Oberfläche, „Symbolic Nuclear Analysis Package“, entwickelt. Das Werkzeug dient zur Erstellung der Eingabedaten und Ausführung und Auswertung der Analysen. Die wesentlichen Elemente von SNAP sind der Model Editor und Navigator. Im Editor können einzelne Komponenten der Modelle grafisch in einem Netzwerk zu einem Anlagenmodell zusammengesetzt werden und die notwendigen Daten eingegeben werden. Bei den in ATLAS verwendeten Systemcodes ist textbasierte Eingabeerstellung in einem Texteditor noch Standard

Für die Analyse von Problemen, in denen lokale Vorgänge eine große Rolle spielen, wie etwa Deborierungsereignisse, thermische Belastungen des Reaktordruckbehälters, Rekritikalität oder Filmsieden im Reaktorkern, ist die räumliche Auflösung von System-Rechenprogrammen häufig nicht mehr ausreichend. Eine belastbare Beschreibung von Vermischungsvorgängen erfordert mehrdimensionale thermohydraulische Simulationsmodelle, in einigen Fällen auch gekoppelte Analysen mit mehrdimensionaler Neutronendynamik und Systemcodes mit sehr detaillierter Systemmodellierung. Es kommen somit neue Simulationsmodelle wie DORT/TORT, COBRA-TF oder ATHLET/FLUBOX zur Anwendung. Mit diesen komplexen Modellen sind aber auch die Anforderungen an die Anwender entsprechend gestiegen. Es wird immer schwerer, die Vielzahl der notwendigen Eingabedaten für die Modelle und die Flut der berechneten Daten mit angemessenem Aufwand zu beherrschen. Eine angemessene Unterstützung könnte eine gemeinsame interaktive 3D-Simulationsumgebung für grafische Eingabeerstellung und Ergebnisauswertung leisten. Diese sollte CAD („Computer Aided Design“) basierte Werkzeuge zur Vorgabe von Geometrie und Gitterdaten, grafische Verfahren zur Nachbildung thermohydraulischer und leittechnischer Systeme, Prozeduren zur Erzeugung der modellspezifischen Eingabeformate, interaktive Bedienoberflächen zur Durchführung der Simulationsrechnung sowie die grafische Aufbereitung und visuelle Darstellung der Simulationsergebnisse beinhalten.

Die Darstellungen von Simulationsdaten in virtueller Umgebung können bei erkennbarer Akzeptanz und Bedarf der Codeanwender ausgebaut werden. Von besonderem Interesse wären dabei 3D-Darstellungen des Reaktorkerns mit dynamischen Daten. Von Interesse sind dann Visualisierungsmöglichkeiten der 2-Phasen-Stömungsvorgänge und die Prüfung der Eignung von Vizard zur Verarbeitung sehr hoher Datenmengen. Anwendungserfahrung zur Darstellung von wissenschaftlichen Daten liegen bisher nur für die kommerziellen Werkzeuge COVISE von Visenso, AMIRA von Template Graphics und die OpenSource Anwendung ViSTA der RWTH Aachen vor, die speziell auf eine Integration numerischer Ergebnisse in Virtual Reality hin konzipiert wurden.

7 Literaturverzeichnis

- /ALL 05/ H.-J. Allelein, et al., 2005, Weiterentwicklung der Rechenprogramme COCOSYS und ASTEC, Abschlussbericht, GRS-A-3266
- /ATH 06/ ATHLET Mod 2.1 Cycle A, 2006, Code Documentation Package, GRS-P-1: Vol. 1: User's Manual, Vol. 2 : Programmer's Manual, Vol. 3 : Validation, Vol. 4 : Models and Methods
- /BAR 08/ M. Barnes, E. Finch, Sony Computer Entertainment Inc., COLLADA – Digital Asset Schema Release 1.4.1, Specification (2nd Edition), March 2008
- /BRO 99/ Frederick P. Brooks, What's Real About Virtual Reality, Department of Computer Science, University of North Carolina at Chapel Hill, 1999
- /BOR 02/ Günter Born, Inside Windows Script Host, Microsoft Press, 2002
- /COH 05/ S. Cohen et al., Scientific Formats for Object-Relational Database Systems: A Study of Suitability and Performance, SIGMOD Vol. 35, No. 2, June 2006
- /COM 09/ ComponentOne, Pittsburgh, USA, Internet flyer,
<http://www.componentone.com/SuperProducts/Chart>
- /DEI 08/ H. Deitenbeck, ATLAS Trendpaket -Programmdokumentation für Entwickler, 22.08.2008, GRS internes Dokument
- /EOS 10/ Eos Systems, Canada, Internet flyer,
<http://www.photomodeler.com/products/photomodeler.htm>
- /FUC 10/ M. Fuchs, M. Olschimke, Mantis Benutzeranleitung,
<http://www.sky.franken.de/mantis-doc/de-docbook>
- /GAU 00/ Gauntt, R. O., R. K. Cole, C. M. Erickson, R. G. Gido, R. D. Gasser, S. B. Rodriguez, M. F. Young, MELCOR Computer Code Manual, Vol. 1: Primer and Users' Guide, Vol. 2: Reference Manuals, Version 1.8.5 May 2000 Sandia National Laboratories, USA, Albuquerque, SAND2000-2417/1+2

/IBM 10.1/ International Business Machines Corp., Ultrium tape technology for backup, restore and archival solutions,

http://www-03.ibm.com/systems/storage/tape/3580/datasheet_ultrium3.html

/IBM 10.2/ International Business Machines Corp., Tivoli Storage Manager,

<http://www-01.ibm.com/software/tivoli/products/storage-mgr/>

/LIN 05/ J. Linxweiler, Ein Prototyp zur immersiven Betrachtung und interaktiven Manipulation räumlicher Objekte, 13. Juli 2005, TU Braunschweig, Diplomarbeit

/NET10/ NetApp Deutschland GmbH,

<http://www.netapp.com/de/products/storage-systems/fas2000/>

/NVI 10/ NVIDIA Corporation, Internet Flyer,

http://www.nvidia.de/page/quadrofx_family.html

/SIE 09/ Johann Sieberer JSPLIT, Internet flyer,

<http://ourworld.compuserve.com/homepages/jsieberer/jsplot.html>

/VOG 06/ T. Voggenberger, et al., 2006, ATLAS XP Erweiterung des Einsatzspektrums für komplexe Analyseanforderungen, Abschlussbericht, GRS-A-3353

/VOG 10.1/ T. Voggenberger, et al., 2010, ATLAS User's Manual, Benutzeranleitung für die ATLAS Anwendung

/VOG 10.2/ T. Voggenberger, et al., 2010, APG User's Manual, Benutzeranleitung für den ATLAS Bildeditor

/WIK 09.1/ Wikipedia, Die freie Enzyklopädie, <http://de.wikipedia.org/wiki/NetCDF>

/WIK 09.2/ Wikipedia, Die freie Enzyklopädie, http://de.wikipedia.org/wiki/Hierarchical_Data_Format

/WIK 09.3/ Wikipedia, Die freie Enzyklopädie, [http:// http://de.wikipedia.org/wiki/Mantis](http://de.wikipedia.org/wiki/Mantis)

/WOR 10/ WorldViz LLC, Vizard Virtual Reality Toolkit, Internet flyer,
http://www.worldviz.com/products/vizard/index_b.html

Anhang 1: Datenmanagement bei den Trendkurven in ATLAS

Zur schnellen Darstellung der Zeitkurven hält das Trendpaket die Ergebniswerte im Speicher (RAM). Da dieser physikalisch begrenzt ist und sehr viele Trends mit einer großen Zahl von Stützpunkten vorhanden sein können, wird die Anzahl der tatsächlich zur Darstellung verwendeten Punkte begrenzt. Dies ist auch deswegen sinnvoll, da die Anzahl der darstellbare Punkte auf einem Bildschirm durch seine Auflösung („Pixel“) beschränkt ist.

In einer Zeitkurve werden maximal 1000 ($mxqntv = maxntv$) Werte dargestellt. Dieser Wert ist ein Parameter und kann angepasst werden. Allerdings ist momentan keine dynamische Speicherverwaltung für die Trenddaten vorhanden. Dieser Wert geht in der Routine „all_trends_from_server“ in die lokalen Arrays $ywertp(maxntv * 1001)$ und $ywertn(maxntv * 1001)$ ein. In den eigentlichen Trendroutinen wird dynamisch Speicher allokiert. Wenn mit der Routine „addchart“ oder „Subroutine ShowVariable“ ein neuer „trend“ allokiert wird, werden $ywert(1000)$ Werte (Modul „Cpage“, type „Ccurve“) für diesen bereitgestellt. Es ist anzunehmen, dass in „Olectra Chart“ ebenfalls entsprechend viele Werte allokiert werden (tatsächlich erhöht sich die Speicherauslastung um 8 KB entsprechend Anzeige im TaskManager). Die Anzahl der benutzen Trendwerte könnte also relativ leicht erhöht werden, wenn man die Programmierung in „all_trends_from_server“ verbessert. Im C++ Teil des Monitors werden die Werte nicht noch einmal gespeichert, sondern aus der FORTRAN-Strukturen abgefragt und an Olectra übergeben (Routine „COlectraView::OlectraDrawPlot“ ruft „gettrendvalues“).

Wenn bei einer laufenden Zeitkurve das Limit von 1000 Werten überschritten wird, wird ein neuer Wert hinzugefügt und ein vorhandener Wert entfernt („subroutine gresar“). Dazu wird ein Zeitschritt gesucht, der kleiner als 10% eines fiktiven, durchschnittlichen Zeitintervalls ist. Dieses Intervall ist das aktuelle Zeitfenster geteilt durch das Wertelimit (aktuell 1000). Wird ein solcher Zeitschritt nicht gefunden, wird der Stützpunkt mit dem kleinsten Zeitschritt entfernt. Frühere Lösungen, bei Limitüberschreitung einfach jeden 2. Wert entfernen oder einen zufälligen Wert entfernen, führen in einigen Fällen zu erheblichen Fehlern in der Anzeige, da in einigen Zeitbereichen nicht mehr genügend Stützstellen vorhanden sind. Im Ergebnis können dann bei sehr langen Zeitfenstern bei allen Ansätzen Zeitbereiche sehr viel weniger Daten enthalten als berechnet wurden.

Insbesondere können auch Extremwerte (Maxima/Minima) fehlen. Ein Konzept, das für alle Trends die Maxima und Minima erhält, ist nicht realisierbar, da alle für alle Trendkurven die Zeitwerte gleich sein müssen. Außerdem wäre der Rechenaufwand zur Ermittlung der Extremwerte sehr hoch, da alle Werte gelesen und analysiert werden müssen.

Anhang 2: Syntax zur Beschreibung der Trendsets in der ATLAS Bilddatei

[trendsets]

In this section trendsets are defined. A trendset consists of one or several diagrams, which are arranged on a page. Every diagram can contain one or several curves. Further down follows a description of the functions to define these pages diagrams and curves. These functions are nearly identically equal to the functions described in the section "Function Reference" of the JSPLLOT manual. The description of every trendset window is started as follows:

=name of the window

The name of the window (max. 39 characters enclosed in single quotation marks) displayed in the window caption

Example

```
'Trend set 1'
```

=window parameters

Optional, if present, the window is displayed, when ATLAS is started

1. Parameter: window priority m. The value m should be between 1 and the number n of initially open windows in ATLAS. A window with a higher m will be in front of a lower m.

The following 4 parameters give the position and the size of the trendset window (in pixel relative to the upper left corner of the main ATLAS window).

2. Parameter: XMIN
3. Parameter: XMAX
4. Parameter: YMIN
5. Parameter: YMAX
6. Parameter: unused
7. Parameter: unused

Example

```
window: 2 0 789 0 658 0 0
```

The description of the trendset shown in the window begins with the statement:

=plt_start

and ends with the statement:

=plt_end

All statements between these two statements have to be closed by a semicolon.

```
Page();  
  Diagram():  
    Header():  
    Xaxis():  
    Y1Axis():  
    Legend():  
    Curve():  
  EndDiagram():  
EndPage();
```

Page

Page(:nbx,:nby)
EndPage()

Page and EndPage enclose a page.

Optional Arguments of Page

Parameter	Beschreibung	Value
:nbx	number of diagrams in X-direction	Integer > 0
:nby	number of diagrams in Y-direction	Integer > 0

Example

```
Page(:nbx 2, :nby 3);  
...  
...  
EndPage();
```

Diagram

Diagram()
EndDiagram()

Between these two statements a diagram is defined. Several diagrams on one page may be defined by repeating a Diagramm() – EndDiagram-block.

Optional Arguments

Parameter	Description	Value
:Distribution	=0 time-value diagrams = 1 for length-value diagrams	<u>0</u> 1

For time-value diagrams (Distribution = 0):

:CLeft ¹	distance between the left page border and the left border of the Chart area	Integer
:CTop ¹	distance between the upper page border and the upper border of the Chart area	Integer
:CWidth ¹	width of the ChartArea	Integer
:CHeight ¹	height of the ChartArea	Integer
:left ¹	distance between the left border of the PlotArea and the left border of the ChartArea	Integer
:top ¹	distance between the upper border of the PlotArea and the upper border of the ChartArea	Integer
:right ¹	distance between the right border of the PlotArea and the right border of the ChartArea	Integer
:bottom ¹	distance between the lower border of the PlotArea and the lower border of the ChartArea	Integer
:backgroundcolor	[red, green, blue] RGB-color for the PlotArea background	0 <= red <= 255 0 <= green <= 255 0 <= blue <= 255
:foregroundcolor	not used	
:IsBoxed	frame around the Plot Area (only used for time-value-diagrams)	T F

1) height, width and distances are given in mm.

Für length-value diagrams:

:ChartType	not used	1
------------	----------	---

Example

```
Diagram(:CLeft 0, :CTop 20, :backgroundcolor [255,255,255]);  
EndDiagram();
```

Header

Header(:text 'text')

Writes a header line on the top of the page area.

Arguments

Parameter	Description	Value
text	header text	Character

Optional Arguments

Keyword	Description	Value
:left	distance to the left border	Integer
:top	distance from the upper border	Integer
:isshowing	visibilitz of the header line	T F
:bordertype ¹	Border Type: 0: None 1: 3dOut 2: 3dIn 3: Shadow 4: Plain 5: EtchedIn 6: Etchedout 7: FrameIn 8: Frameout 9: Bevel	Integer [0:9]
:borderwidth ¹	Border Width	Integer
:backgroundcolor	[red,green,blue] RGB-color for the background of the plot area	0 <= red <= 255 0 <= green <= 255 0 <= blue <= 255
:foregroundcolor	[red,green,blue] RGB-color for the foreground of the plot area	0 <= red <= 255 0 <= green <= 255 0 <= blue <= 255
:fontname	Font name	Character
:size	Font size	Integer
:bold	Font type	T F
:Italic	typeface	T F
:strikethrough	Line through the header line	T F
:underline	underlining the header line	T F

¹⁾ These functions are available only for time value diagrams.

Example

```
Header(:text 'Pressure in the Steam Generator',
      :left 60, :top 10, :isshowing T, :bordertype 0,
      :borderwidth 0, :backgroundcolor [255,255,255],
      :foregroundcolor [ 0, 0, 0], :fontname 'Arial', :size 8.2,
      :bold F, :italic T, :strikethrough F, :underline F);
```

Axes

XAxis()

Y1Axis()

Creates and formattes the X- and Y-Axis.

Optional Arguments

Keyword	Description	Value
:text	Label Text	Character
:fontname	fontname of the axis label	Character
:size	fontsize of the axis label	Real
:bold	fonttype of the axis label	T F
:Italic	typeface of the axis label	T F
:strikethrough	strikethrough the axis label	T F
:underline	underlining the axis label	T F
:weight	not used	0 >= Integer < 10000
:TitleRotation ¹	Rotation of the axis label	0 90 270
:IsStyleDefault ²	use Default values for the Tic-Length	T F
:Min ²	Axis minimum	Real
:Max ²	Axis maximum	Real
:IsStyleDefault ²	not used	Logical
:LineStyle ²	Line style of the Axis: 0: None 1: Solid 2: Long Dash 3: Dotted 4: ShortDash 5: LslDash 6: DashDot	Integer
:Ticlength ²	length of the Tic marks	Integer > 0
:IsLogarithmic ²	Logarithmic scaling	T F
:IsReversed ²	minimum of the axis on the right or upper side	T F

:IsShowing ²	visibility of the axis	T F
:MajorGrid_Spacing_IsDefault ²	use default values for the position of the gridlines	T F
:MajorGrid_Style ²	line type of the grid lines: 0: None 1: Solid 2: Long Dash 3: Dotted 4: ShortDash 5: LsIDash 6: DashDot	Integer
:MajorGrid_width ²	width of the grid lines	Integer
:MajorGrid_Color ²	[red,green,blue] RGB-color of the plot area	0 <= red <= 255 0 <= green <= 255 0 <= blue <= 255
:MajorGrid_SpacingValue ²	distance between two gridlines in user units	Real
:Tic_Spacing_IsDefault ²	use default values for tic marks	T F
:Tic_Spacing ²	distance between two tic marks in user units	Real

1) only available for the Y-axis

2) these functions are available only for length value diagrams

Example

```
XAxis (:IsStyleDefault T, :LineStyle 2, :TicLength 2,
       :IsLogarithmic F, :IsReversed F, :IsShowing T,
       :foregroundColor [ 0, 0, 0], :fontname 'Arial', :size 8.2,
       :MajorGrid_Spacing_Isdefault T, :MajorGrid_Style 2,
       :MajorGrid_Width 1, :MajorGrid_Color [192,192,192]);
```

Legend

Legend()

Creates a legend to describe a curve.

Optional Arguments

Keyword	Beschreibung	Wert
:left ²	Distance from the left border of the chartarea	Integer
:top ²	Distance from the upper border of the chart area	Fontname
:text	header over the legend texts	String
:fontname	Font for the legend texts	Character
:size	font size of the legend texts	Float
:bold	font type of the legend texts	T F
:Italic	typeface of the legend texts	T F
:strikethrough	strikethrough the legend texts	T F
:underline	underlining the legend texts	T F
:bordertype	0: None 1: 3dOut frame type 2: 3dIn 3: Shadow 4: Plain 5: EtchedIn 6: Etchedout 7: FrameIn 8: Frameout 9: Bevel	Integer [0:9]
:borderwidth	width of the frame	Integer
:backgroundcolor	[red,green,blue] RGB-color of the legend background	0 <= red <= 255 0 <= green <= 255 0 <= blue <= 255
:foregroundcolor	[red,green,blue] RGB-color of the legend foreground	0 <= red <= 255 0 <= green <= 255 0 <= blue <= 255

:IsShowing	visibility of the legend	T F
:IsShowing3DShading	3D-legend frame	T F
:Orientation	orientation of the legend texts	,horizontal' ,vertical

1) these functions are available only for time value diagrams.

2) all distances in mm

Example;

```
Legend(:left 70, :top 40, :text 'Legende',
       :fontname 'Arial', :size 6.0, :bold F, :italic T,
       :strikethrough F,
       :underline F, :bordertype 4, :borderwidth 1,
       :backgroundcolor [255,255,255], :foregroundcolor [ 0, 0, 0],
       :foregroundcolor [ 0, 0, 0], :fontname 'Arial', :size 8.2,
       :IsShowing T, :IsShowing3DShading F,
       :Orientation 'Vertical');
```

Curve

Curve(data)

Creates a curve as part of a diagram. Several curves may be generated by several curve definitions.

Arguments

Parameter	Beschreibung	Wert
:f <code>typ</code>	type of the data	,athlet' ,Flott' ,astec' !L <code>enval</code> '
:f <code>key</code>	name of the *.key file	Character
:f <code>pd</code>	name of the *.pd file	Character
:n <code>ametrd</code>	String, composed of the 4 key words	Character

Optional Arguments

Parameter	Description	Value
:Component ¹	ASTEC Component path	Character
:SeriesLabels ¹	curve specific legend text	Character
:lineWidth	line width	Integer > 0
:lineColor	line color	[255,0,0], ",red"
:linePattern	line pattern: 0: None 1: Solid 2: Long Dash 3: Dotted 4: ShortDash 5: LsIDash 6: DashDot	Integer
:SymbolSize	marker size	Integer
:SymbolColor	marker color	[255,0,0], ",red"
:SymbolShape	marker tyzpe 0. None 1. Dot 2. Box	Integer

	3. Tri 4. Diamond 5. Star 6. VertLine 7. HorizLine 8. Cross 9. Circle 10. Square 11. Invert Tri 12. DiagCross 13. OpenTri 14. OpenDiamont 15. OpenInvertTri	
:ax ²	Multiplicator for X-transformation	Real
:bx ²	Summand for X-transformation	REAL
:ay ²	Multiplicator for Y-transformation	Real
:by ²	Summand for Y-transformation	REAL

1) these functions are available only for time value diagrams

Example;

```
Curve
(:fkey 'D:\ATHLET\ATHLET22A\User\Sample1\sample1.run0.key',
:fpd 'D:\ATHLET\ATHLET22A\User\Sample1\sample1.run0.pd',
:nametrd 'P1-SG-UT - NODE5 PRESS',
:SeriesLabels 'PRESS(P1-SG-UT,-,5)', :LineWidth 1, :LineColor [ 34,139, 34],
:LinePattern 2,
:SymbolSize 1, :SymbolColor [255, 0, 0], :SymbolShape 1,
:ax 0.10000E+01, :bx 0.00000E+00, :ay 0.10000E+01, :by 0.00000E+00);
```

Anhang 3: Beispiel für den Befehl „MULTISCALE“ in einer APG-Bilddatei

Mittels der Funktion „Multi Color Scale“ wird das APG-Kommando „MULTISCALE“ generiert. Wenn die Inputroutine „NXTCRD“ von ATLAS das Kommando „MULTISCALE“ einliest, dann werden anhand dessen Spezifikationen eine Reihe von internen Befehlen zur Erzeugung der entsprechenden Multifarbskala generiert. Dazu zählen die Befehle zu Generierung der einzelnen Farbskalen, des Multifarbskala-Anzeige und der anklickbaren Liste der Variablen (Auswahlmenü „CHOICE“).

```

MULTISCALE 1\ ← Multi Color Scale Nr.
'VOID'\ ← Initial Variable
0.06000      0.94000\ ← Position
0.16000      0.03000      0.01600\ ← Size und Text Height
0 0\ ← Initial Variable
12\ ← Anzahl der Variablen
'Heat source (W)'  'POWER'  3\ ← Variable DescriptionName
-0.10000E+10  0.000  0.000  1.000\ ← Min. bzw. Max. und Color
0.00000E+00  1.000  1.000  1.000\
0.10000E+10  1.000  0.000  0.000\
'Pressure (Pa)'  'PRESSURE'  2\
0.00000E+00  0.100  0.800  0.800\
0.20000E+08  0.500  0.000  1.000\
'Fluid temperature (°C)'  'FLUIDTEMP'  2\
0.00000E+00  0.000  0.000  1.000\
0.40000E+03  0.800  0.800  0.300\
'Mixture density (kg/m3)'  'DENSITY'  2\
0.00000E+00  0.800  0.800  0.300\
0.10000E+04  0.000  0.000  1.000\
'Volumetric vapor fraction'  'VOID'  6\
0.00000E+00  0.000  0.000  0.800\
0.10000E-02  0.000  0.000  0.800\
0.10000E-02  0.250  0.250  1.000\
0.99900E+00  0.900  0.900  1.000\
0.99900E+00  1.000  1.000  1.000\
0.10000E+01  1.000  1.000  1.000\
'Mass flow (kg/s)'  'MASSFLOW'  3\
-0.20000E+05  1.000  0.000  0.000\
0.00000E+00  1.000  1.000  1.000\
0.20000E+05  0.000  0.000  1.000\
'Liquid mass flow (kg/s)'  'LIQFLOW'  3\
-0.20000E+05  1.000  0.000  0.000\
0.00000E+00  1.000  1.000  1.000\
0.20000E+05  0.000  0.000  1.000\
'Vapor mass flow (kg/s)'  'VAPFLOW'  3\
-0.10000E+04  1.000  0.000  0.000\
0.00000E+00  1.000  1.000  1.000\
0.10000E+04  0.000  0.000  1.000\
'Vapor enthalpy fraction'  'QUALITYH'  2\
-0.50000E+00  0.200  0.600  0.200\
0.15000E+01  0.600  1.000  0.600\
'Vapor mass fraction'  'QUALITYM'  2\
0.00000E+00  0.200  0.600  0.200\

```

0.10000E+01	0.600	1.000	0.600\
'Mixture enthalpy (J/kg)'		'ENTHALPY'	2\
0.00000E+00	0.200	0.600	0.900\
0.30000E+07	0.900	0.600	0.200\
'Total evap./cond. rate (kg/s)'		'GEI'	3\
-0.10000E+03	0.000	0.000	1.000\
0.00000E+00	1.000	1.000	1.000\
0.10000E+03	1.000	0.000	0.000

Anhang 4: Namensgebung und Parameterübergabe im Modul „NetCDF“

Das Modul wird von verschiedenen Anwendungen mit unterschiedlichen Namensanforderungen und Parameterinterpretationen genutzt. Außerdem wird für die Prozessmodelle gegen die statischen Laufzeitbibliotheken gelinkt, ansonsten aber immer gegen die dynamischen Bibliotheken. Es wird im Normalfall eine statische Bibliothek erzeugt, allerdings ist für die interaktive VNet Anwendung „NetCDFUi“ eine DLL nötig, da von DotNet keine statischen Bibliotheken eingebunden werden können. Die Parameterübergabe erfolgt dort nach C-Konventionen. Insbesondere werden bei den Strings keine Längen übergeben, sondern diese müssen mit „0“ terminiert sein.

Zunächst werden die Routinen aus dem Modul im Monitor über die Fortran-Utilities oder direkt von C++ angesprochen. Diese benutzen „LowerCase“ und „No-Underscore“ als External Names, um C-Routinen einbinden zu können. Dies ist **nicht** der Standard von Intel Fortran (Standard ist „UpperCase“ und „No-Underscore“) und führt daher bei der Verwendung in den Prozessmodellen (ATHLET,...) zu Problemen. Da alle Routinennamen in Modulen normalerweise von Intel mit dem Modulnamen dekoriert werden, sind im Modul bereits „ALIAS“ Anweisungen vorhanden, die alle Namenskonventionen im Monitor berücksichtigen, z. B.:

```
!DEC$ ATTRIBUTES ALIAS:'_nc_get_trend_values' :: nc_get_trend_values
```

Der Aufruf von FORTRAN kann in beliebiger Schreibweise erfolgen, d. h. Groß- und Kleinschreibung spielt keine Rolle. Das gilt nur, falls der Name in einem Interface Block definiert wurde bzw. ein entsprechendes „Use“ - Statement vorhanden ist. Andernfalls werden Großbuchstaben gesucht! Wenn man also die Benutzung von ATLAS-Moduln (z. B. use ATLAS_NCFCOM) in ATHLET vermeiden will, muss man Routinen aus der „process_source.lib“ aufrufen, da diese mit Großbuchstaben generiert werden. Für die Benutzung von Großbuchstaben für die Routinen in „ATLAS_NCFCOM_stat.lib“, sollte bei Bedarf der Routine ein ALIAS mit Großbuchstaben definiert werden und dieser mit `#ifdef _uppercase` einbinden werden wie z. B.

```
!DEC$ ATTRIBUTES ALIAS:'_NC_TEST' :: nc_test
```

Alternativ kann man auch die rufende Routine mit der Option „External Procedures > Name Case Interpretation > As-Is“ übersetzen. Das geht aber nur, falls alle anderen

gerufenen Routinen und Symbole mit Großbuchstaben verwendet sind. Das ist im Einzelfall möglicherweise nicht garantiert.

Die NetCDF-Basisbibliotheken müssen dagegen unter Beachtung der Schreibweise erzeugt werden. Die Einstellung erfolgt in der Datei „**cfortran.h**“ (bei der Originaldistribution war Kleinschreibung eingestellt, wenn `#define pgiFortran`). Die zu ändernde Stelle ist mit

#define UPPER (bzw. #ifdef UPPER)

gekennzeichnet. Die statische Bibliothek „NetCDF_stat.lib“ muss mit Großbuchstaben erzeugt werden.

Die Parameterübergabe von Arrays und Character Strings variabler Dimension erfordert (`dimension (:)` oder `len=*`) im rufenden Programm unbedingt einen INTERFACE Block wie:

```
interface
  subroutine nc_test (itest)
    !DEC$ ATTRIBUTES ALIAS: '_NC_TEST' :: nc_test
    integer, intent(in) , dimension (:):: itest
  end subroutine nc_test
end interface
```

Ohne diesen Block kommt es zu Fehlern bei der Parameterübergabe.

Anhang 5: Schnittstellenprogramme in den Prozessmodellen (z. B. ATHLET) und Monitor für Datenverwaltung und Datenaustausch

A-20

Routine	Funktion / Erklärung	Parameter	Erklärung
Basic Routines			
nc_set_maxexpected_keys	Set the variable "max_expected_keys" (Only necessary, if keys are generated at runtime)	mekeys	Maximum number of expected keywords
nc_get_numrecs	Get the number of records on NetCDF file	ncfno numrecs status	Number of the NetCDF file in an application Number of records on NetCDF file Error state
nc_get_numvars		ncfno numvars status	Number of the NetCDF file in an application Number of variables on NetCDF file Error state
nc_set_filestate	Set the statusmes (status_varid) of a NetCDF file	ncfno statusmes	Number of the NetCDF file in an application Current state of NCF-file: <i>"No_Data"</i> <i>"Keywords_completed"</i> <i>"Data_Extended"</i> <i>"Completed"</i> <i>"New_Messages"</i>

Routine	Funktion / Erklärung	Parameter	Erklärung
		status	"Locked" "Unknown" Error state
nc_get_filestate	Get the statusmes (status_varid) of a NetCDF file	ncfno statusmes status	Number of the NetCDF file in an application Current state of NCF-file: Error state
nc_close_for	Close of a NetCDF file	ncfno status	Number of the NetCDF file in an application Error state
nc_get_inittime	Get the first time (rec == 1) on NetCDF file	ncfno inittime status	Number of the NetCDF file in an application First time on NetCDF file Error state
nc_get_lasttime	Get the last time (rec == numrecs) on NetCDF file	ncfno lasttime status	Number of the NetCDF file in an application Last time on NetCDF file Error state
nc_update_recordtimes	Update the array "time_of_recs"	ncfno status	Number of the NetCDF file in an application Error state
nc_connect_file	Connect to an existing NETDCF file (open it) and read some definitions	Ncfile ncfno status	Name of the NetCDF file to be read Number of the NetCDF file in an application Error state
nc_connect_file_cint	C-Interface to nc_connect_file (has to be used if called by C)	Ncfile ncfno status	Name of the NetCDF file to be read Number of the NetCDF file in an application Error state

Routine	Funktion / Erklärung	Parameter	Erklärung
Keyword Data Routines			
nc_put_keywords	Create a new NETDCF file with keyword names ("keys") inserted as data, if ncfno=0 Modify a NETDCF file with changed keyword names, if ncfno>0	keys units& keytype klen& ulen numkeys ncfile ncfno status totkeys startindex	Array of keyword names Array of physical units Array of keyword types String length of keys String length of units Number of elements of keys Name of the NetCDF file to be written Identifier of the NetCDF file to be written, if <=0, then it will be generated Error state If given, the written keys are extended with dummy names Index of first keyword array element (default == 1)
nc_get_keywords	Connect to an existing NETDCF file and read keyword names and units	keys units& keytype klen& ulen numkeys ncfile ncfno status startindex	Array of keyword names Array of physical units Array of keyword types String length of keys String length of units No. of requested keys, if < 0, all keys will be returned Name of the NetCDF file to be read Identifier of the NetCDF file to be read, if <0, then it will be opened by the ncfno parameter Error state Index of first keyword array element

Routine	Funktion / Erklärung	Parameter	Erklärung
			(default == 1)
nc_read_keywordfile	Read an existing ASCII-keywordfile (*.key) and store the data (Names und units) into memory	iokeyunit numval_gvector ncfno status keyfile	Unit nr. of keyword file (unused) (unused) Error state if present, the keywords are read from "keyfile" (otherwise from "iokeyunit")
Graphics data Routines			
nc_put_timepoints	Put the available timepoints(i.e. record times) on NetCDF file	ncfno timevec np status	Number of the NetCDF file in an application Vector of time points on a NetCDF file Number of time points Error state
nc_get_timepoints	Get the available timepoints(i.e. record times) on NetCDF file	ncfno timevec np status	Number of the NetCDF file in an application Vector of time points on a NetCDF file Number of time points Error state
nc_send_complete_gvector	Stores a full result vector into a NetCDF file	numval_gvector ncfno time gvector status	Number of elements of gvector Number of the NetCDF file in an application Problem time related to gvector Array of results (values to be plotted) to be written Error state
nc_get_complete_gvector	Get a full result vector from a NetCDF file	ncfno	Number of the NetCDF file in an ap-

Routine	Funktion / Erklärung	Parameter	Erklärung
	(this routine is currently unused)	time_desired numval_gvector^ time_ret gvector vec_found status nextstep samestep previousstep	plication Problem time required Number of elements of gvector Problem time related to gvector Array of results (values to be plotted) returned ==0 no values returned ==1: valid values found Error state ==1 get the next time (rec+1) ==1 get the same time (rec) ==1 get the previous time (rec-1)
nc_get_gvalues	Get a bunch of results (gvector by addresses_gvector) from a NetCDF file	ncfn0 time_desired numval_gvalues addresses_gvalues time_ret gvalues vec_found status nextstep samestep previousstep	Number of the NetCDF file in an application Problem time required Number of elements of gvalues Addresses of required elements Problem time related to gvalues Array of results (values to be plotted) returned ==0 no values returned ==1: valid values found Error state ==1 get the next time (rec+1) ==1 get the same time (rec) ==1 get the previous time (rec-1)
nc_get_trend_values	Get a bunch of trends (tvalues by addresses_tvalues) and the time points related from a NetCDF file	ncfn0 time_start time_end	Number of the NetCDF file in an application Start time of trend time window^ End time of trend time window

Routine	Funktion / Erklärung	Parameter	Erklärung
		maxval_required num_addresses addresses_tvalues numval_returned time_values trend_values status	Maximum of values to be returned for a single trend If more are available, they are reduced If ≤ 0 , the values time_start and time_end are interpreted as record numbers Number of addresses (trends) requested Addresses of required values Number of values returned for a single trend (\leq maxval_required) Array of time values related to trend values (numval_returned) Array of results (trend values, trend vector by trend vector, i.e. num_addresses x numval_returned) Error state
Message Routines			
nc_create_mesfile	Creates a message file and the necessary variables	mesfile_name mesfile_found status	Problem specific name of message file (including the complete UNC-path) , the name "messages.ncc" will be appended ==0 not connected ==1: connected Error state
nc_connect_mesfile	Tries to open a message file and get the variable id's in the file	mesfile_name	Problem specific name of message file (including the complete UNC-path) , the name "messages.ncc" will be appended

Routine	Funktion / Erklärung	Parameter	Erklärung
		mesfile_found status readonly	==0 not connected ==1: connected Error state ==1; open the file for read
nc_send_message	Stores a message string into a NetCDF message file (including a real time stamp)	message status	Message string (maximum 255 bytes) Error state
nc_get_message	Gets a message string from a NetCDF message file	message mes_found status	Message string (maximum 255 bytes) ==0 no message returned ==1: new message found Error state

Verteiler

		Exemplare: gedruckte Form:	CD-ROM:	Version als pdf:
BMWi				
Referat III B 4		1 x		
GRS-PT/B				
Internationale Verteilung	(FIZ)	40 x	2 x	
Projektbegleiter	(bri)	3 x	1 x	1 x
GRS				
Geschäftsführung	(wfp, stj)	--		je 1 x
Bereichsleiter	(erv, paa, prg, rot, stc, ver, zir)	--		je 1 x
Abteilungsleiter	(luw, poi, som, vek)	--		je 1 x
Projektleiter/Autor	(vog)	2 x		1 x
Projektbetreuung	(wal)	--		1 x
Informationsverarbeitung	(nit)	--		1 x
Bibliothek	(Köln)	1 x		
Gesamtauflage		47	3	

**Gesellschaft für Anlagen-
und Reaktorsicherheit
(GRS) mbH**

Schwertnergasse 1
50667 Köln
Telefon +49 221 2068-0
Telefax +49 221 2068-888

Forschungszentrum
85748 Garching b. München
Telefon +49 89 32004-0
Telefax +49 89 32004-300

Kurfürstendamm 200
10719 Berlin
Telefon +49 30 88589-0
Telefax +49 30 88589-111

Theodor-Heuss-Straße 4
38122 Braunschweig
Telefon +49 531 8012-0
Telefax +49 531 8012-200

www.grs.de