



# PEARL-News

Ausgabe 2    November 2005

Mitteilungen  
der GI-Fachgruppe 'real-time'  
Echtzeitsysteme und PEARL

ISSN 1437-5966

# Impressum

Herausgeber	GI-Fachgruppe 'EP' Echtzeitsysteme und PEARL URL: <a href="http://www.real-time.de">http://www.real-time.de</a>
Sprecher	Dr. P. Holleczeck Universität Erlangen-Nürnberg, Regionales Rechenzentrum Martensstraße 1, D-91058 Erlangen Telefon: 09131/85-27817 Telefax: 09131/30 29 41 E-Mail: <a href="mailto:holleczeck@rrze.uni-erlangen.de">holleczeck@rrze.uni-erlangen.de</a>
Stellvertreterin	Prof. Dr.-Ing. B. Vogel-Heuser Bergische Universität, Fachbereich E Lehrstuhl Automatisierungstechnik / Prozeßinformatik Rainer-Gruenther-Str. 21, D-42119 Wuppertal Telefon: 0202/439-1848 Telefax: 0202/429-1944 E-Mail: <a href="mailto:bvogel@uni-wuppertal.de">bvogel@uni-wuppertal.de</a>
Redaktion	Prof. Dr. R. Müller FH Furtwangen, Fachbereich Computer- & Electrical Engineering Robert-Gerwig-Platz 1, 78120 Furtwangen Telefon: 07723/920-2416 Telefax: 07723/920-2610 E-Mail: <a href="mailto:mueller@fh-furtwangen.de">mueller@fh-furtwangen.de</a>  Prof. Dr. R. Müller HTWK Leipzig, Fachbereich Elektrotechnik und Informationstechnik Wächterstraße 13, D-04107 Leipzig E-Mail: <a href="mailto:mueller@fbeit.htwk-leipzig.de">mueller@fbeit.htwk-leipzig.de</a>
ISSN	1437-5966

Redaktionell abgeschlossen am 10. November 2005

## Einreichung von Beiträgen

Diese Zeitschrift soll nicht nur Mitteilungsblatt sein, sondern auch eine Plattform für den Informations- und Meinungsaustausch zwischen allen an den Fragen der Echtzeitprogrammierung Interessierten bilden. Diskussionsstoff bzw. offene Fragen gibt es auf unserem Gebiet reichlich.

Wir möchten Sie, liebe Leserinnen und Leser, daher ausdrücklich ermuntern, auch in Zukunft die PEARL-News durch Ihre Beiträge mit zu gestalten. Für ein ausgewogenes Bild der News sollten Beiträge nicht länger als 5 Seiten sein.

Rainer Müller (Furtwangen)  
Rolf Müller (Leipzig)

## Inhalt

- 1 Reisestipendien
- 2 PEARL-Geschichte
- 3 Abstracts von Abschlussarbeiten

# 1 Reisestipendien

Die GI-Fachgruppe (FG) Echtzeitsysteme & PEARL <http://www.real-time.de> vergibt in diesem Jahr erstmals und in der Folge jährlich ein Reisestipendium. Die Motivation ist eine Mehrfache:

- Die Förderung junger Wissenschaftler mit einschlägigem Hintergrund (Echtzeitsysteme) in Zeiten knapper Mittel.
- Die Stärkung des Nachwuchses in unserer FG.
- Die internationale Vertretung der deutschen Echtzeit-Szene.
- Werbung fuer unsere FG.

Das Stipendium wird an junge Wissenschaftler (fortgeschrittene Studenten, Doktoranden) als Zuschuss für Reisen zu einer anerkannten internationalen Konferenz im Bereich Echtzeitsysteme vergeben. Ausgelobt werden zwei Reisestipendien zu je maximal 500 Euro.

Liegen mehr Bewerbungen als Stipendien vor, entscheidet die FG-Leitung anhand folgender Kriterien:

- Qualität des Beitrags
- Nähe zum Ziel unserer FG

Die Ausschreibung erfolgte aus Zeitgründen in diesem Jahr (nur) an die Mitglieder und Interessenten der FG. Anmeldungen waren bis Freitag 11.11.2005 elektronisch an den Sprecher der Fachgruppe zu richten. Die Entscheidung über die Annahme des Stipendiengesuchs wird spätestens anlässlich des diesjährigen Workshops "Echtzeitaspekte bei der Koordinierung Automomer Systeme" am 1./2. Dezember 2005 in Boppard bekanntgegeben.

Im nächsten Jahr werden die Stipendien über Fachorgane ausgelobt. Von den Bewerbern sind der eingereichte wissenschaftliche Beitrag und die Bestätigung des Programm-Komitees über dessen Annahme vorzulegen. Abgerechnet werden können einzelne Originalbelege in Höhe von bis zu 500 Euro pro Stipendiat. Im Zweifelsfall gelten die Reisekostenregelungen der GI bzw. der jeweiligen Hochschule/Forschungseinrichtung.

Peter Holleczek  
Universität Erlangen-Nuernberg

## 2 Anmerkungen zur (Früh-)Geschichte von PEARL

### Vorbemerkungen

Am Rande der PEARL-Tagung im November 2004 hatte der Verfasser Herrn Halang zugesagt, für die Jubiläumsschrift der Fachgruppe aus seiner Erinnerung "etwa eine A4-Seite" über die "Urgeschichte" von PEARL zu schreiben. Dies stellte sich inzwischen als ein sehr leichtsinniges Versprechen heraus. Schon der Umfang war ein Problem. Allein aus den Tagebuchnotizen des Verfassers ließe sich ein ganzes Buch zusammenstellen, vom noch vorhandenen technischen Material ganz zu schweigen. Die jetzige Länge dürfte also ein recht guter Kompromiss zwischen der Vorgabe und der Materialmenge sein.

Dann war es sehr schwierig, einen passenden Stil zu finden. Ein amerikanisches Sprichwort lautet: "Pioniere erkennt man an der Anzahl der Pfeile in ihrem Rücken". Die meisten überleben das bekanntlich nicht. Die aber überleben, haben Stoff für viele "Sagas". In der Wissenschaft ist es jedoch nicht Brauch, solche zu erzählen. Es müßte aber erlaubt sein, Lehren zu ziehen. Welche könnten das im Fall von PEARL sein?

Nach längerem Nachdenken ist der Verfasser zu der Ansicht gekommen, dass die Geschichte von PEARL ein gutes Beispiel für eine "Fallstudie" ist. Sie zeigt einen Erfolg der deutschen Forschungsförderung der 60er und frühen 70er Jahre, warum er damals möglich war, wie er verspielt wurde und warum er heute nicht mehr möglich wäre. Die heutige Forschungspolitik könnte daraus viel lernen - wenn jemand wollte  
.....

## 1967 -1970 Aufbauzeit

Die dem Verfasser bekannte Geschichte von PEARL geht zurück bis ins Jahr 1967. Sie hat auch mehr als einen Ursprung. Es schien also etwas "in der Luft zu liegen". Die Vorarbeiten reichten von Ideenskizzen und Studien (V. Haase, GfK Karlsruhe, P. Elzer, Universität Erlangen-Nürnberg, H. Mittendorf, Siemens, und andere) bis hin zu konkreten Entwicklungsprojekten ("PASI" (= Prozessautomatisierungssprache), G. Koch, Brown Boveri & Cie). Dabei fällt auf, dass sie alle aus der Anwendung kamen (Kerntechnik, Automatisierungstechnik). Die Entwicklung war also "demand driven".

Auch die Förderung durch das damalige Bundesministerium für Bildung und Wissenschaft (BMBW) war bedarfsorientiert. Sie wurde koordiniert durch die "Studiengruppe Nuklearelektronik". Diese befürwortete 1968 das "Projekt NDV13" - die Ausarbeitung eines Sprachvorschlags unter dem Arbeitstitel "EXOS" (=EXperimentOrientierte Sprache).

Daraufhin ging alles sehr schnell. Eine Arbeitsgruppe traf sich zum ersten Mal 1969 in Frankfurt/Main im Institut für Kernphysik. Die Begeisterung war groß und der Einsatz der Mitglieder hervorragend. Im Sommer 1970 erschien die erste Veröffentlichung /1/. Ihre Autoren kamen von den Firmen AEG, BBC und Siemens sowie von den Forschungsstellen GfK Karlsruhe, HMI Berlin und der Universität Erlangen-Nürnberg. Charakteristisch für das Arbeitsklima war, dass der Name PEARL (=Process and Experiment Automation Realtime Language) im Verlauf einer angeregten "Abendsitzung" bei Weihenstephaner Weißbier entstand. Die Kollegen aus der Industrie hatten nämlich darum gebeten, der Entwicklung ein etwas "weniger labororientiertes Image" zu geben. Der Name wurde ursprünglich als "rein provisorisch" betrachtet - bis man einen besseren finden würde .....

Die Veröffentlichung bewirkte einen nächsten Aktivitätsschub. Weitere Forschungsinstitutionen und Industriefirmen beteiligten sich. VDI und VDE unterstützten die Gemeinschaftsaktion. Die Studiengruppe Nuklearelektronik befürwortete die Förderung einer Testimplementation an der Uni Erlangen-Nürnberg (Projekt NDV15).

Im Herbst 1970 geschah dann etwas, was in der heutigen Zeit in Deutschland zur sofortigen Einstellung der Arbeiten geführt hätte: die PEARL-Gruppe erfuhr von einer Aktivität in den USA, dem "Purdue Workshop on Industrial Computer Languages" an der Purdue University in Lafayette, Indiana. Dieses hatte sich zum Ziel gesetzt, Programmiersprachen für Echtzeitanwendungen in der Industrie zu normen oder zur Normreife zu entwickeln. Damals reiste der Verfasser zusammen mit einer Delegation von Industrievertretern dorthin, um sich die Sache erst einmal anzusehen. Es stellte sich heraus, dass die deutsche Entwicklung den US-Ansätzen zeitlich weit voraus und technisch überlegen war. Daraufhin wurde die "amerikanische Herausforderung" angenommen, die Arbeiten intensiviert und in internationalen Normungsgremien sowie bei der EG flankierende Maßnahmen eingeleitet.

## 1971 - 1974 Aussicht auf Durchbruch

In der BRD herrschte damals nicht Untergangs- sondern Aufbruchstimmung. Politik, Forschung und Industrie zogen noch weitgehend am gleichen Strang und wollten die internationale Konkurrenzfähigkeit unseres Landes verbessern. Eine Maßnahme in diesem Sinne war die Gründung des Bundesministeriums für Forschung und Technologie (BMFT). Eine seiner ersten Aktionen war die Einleitung intensiver Fördermaßnahmen auf dem Gebiet der Automatisierungstechnik. Dazu wurde 1971 das "Projekt PDV" (= ProzeßDatenVerarbeitung) gegründet. Projektträger war die GfK in Karlsruhe. Mitglieder des PEARL-Kreises spielten eine wesentliche Rolle bei seinem Aufbau. Im Rahmen des Projektes PDV entstanden viele bahnbrechende und bis heute noch nicht überholte Entwicklungen auf dem Gebiet des Einsatzes von Rechnern in der Industrie. Sie auch nur anzudeuten, würde aber den Umfang dieser Darstellung derart ausweiten, dass sie einer gesonderten Veröffentlichung vorbehalten bleiben müssen.

Um die Implementierung von PEARL auf breiter Basis zu ermöglichen, musste eine Definition der Sprache fertig gestellt werden, die auf einem Konsens aller beteiligten Firmen und Forschungsstellen beruhte. Dies war das Ziel einer äußerst intensiven zweiwöchigen Klausur im Frühjahr 1972 in einem Tagungshotel am Bodensee. Es gab dabei Tage, an denen bis 4 Uhr morgens gearbeitet wurde. Der Verfasser erinnert sich noch deutlich an die Härte (und Lautstärke) der Diskussionen, die manchmal den Spott anderer Hotelgäste hervorriefen. Er erinnert sich aber auch noch an den konstruktiven Geist, in dem diese Debatten geführt wurden. Es gelang schließlich dabei, ein wesentliches Problem der damaligen Sprachentwickler zu entschärfen: den "Religionsstreit" zwischen den Anhängern von "PL/I" /2/ einerseits und "ALGOL 68" /3/ andererseits. Die letztere Sprache galt damals vor allem in der neu entstandenen deutschen Informatik

als das Maß aller Dinge.

Nach einer ungeheuren Menge von Detailarbeit konnte dann im April 1973 als erster Forschungsbericht des Projektes PDV die Spezifikation von PEARL veröffentlicht werden /4/. Die Verfasser kamen diesmal von den Firmen AEG-Telefunken, BBC, ESG, Siemens, Unicom und Werum, von den Forschungseinrichtungen HMI Berlin und KfA Jülich sowie von den Universitäten Erlangen-Nürnberg, Heidelberg und Karlsruhe.

Zur Normung von PEARL wurde schließlich im Jahre 1974 der Arbeitskreis DIN/FNI AK 5.8 gegründet. Die Arbeit schien erfolgreich getan. Diejenigen Leser, die sich für weitere Details und Hintergrundinformationen interessieren, seien ermutigt, sich die damaligen ausführlichen Berichte des Verfassers zu beschaffen /5/.

## Eine erste Analyse

Welches sind nun aus heutiger Sicht Faktoren, die zum damaligen Erfolg führten:

- Es handelte sich um ein neues, offenes Forschungs- und Technologiegebiet, in dem noch keine etablierte Wissenschaftsszene bestand. Neue Ideen hatten es deshalb noch leichter, sich durchzusetzen, da sie nicht von etablierten Denkschulen gestoppt werden konnten.
- Zwischen der Industrie und den deutschen Universitäten bestand noch ein Klima, in dem Zusammenarbeit selbstverständlich war. Es war noch nicht durch die Ideologie der 68er vergiftet, gemäß der die "gute Forschung" nicht durch die "böse profitorientierte Industrie" korrumpiert werden dürfe.
- Die Entwicklung von PEARL erfolgte aus den Bedürfnissen der Anwendung heraus, nicht um rein akademische Prinzipien zu formulieren oder zu testen.
- Zwischen Universitäten und Großforschungseinrichtungen bestand noch ein leidliches Gleichgewicht in Bezug auf die Chancen, staatliche Fördermittel zu erhalten.
- Die Universität Erlangen-Nürnberg konnte eine katalytische Rolle wahrnehmen, da sie als "kleine Universität" und damit zunächst von den Großforschungseinrichtungen nicht als Konkurrenz betrachtet wurde.
- In der BRD herrschte noch ein innovationsfreundliches Klima und ein vernünftiger internationaler Selbstbehauptungswille seitens der Regierung.
- Last but not least war die große Expansion der Universitäten noch nicht angelaufen und es gab also eine größere Anzahl (hungriger) junger Forscher ohne Aussicht auf Planstellen, die ihren Leistungswillen deshalb in Entwicklungsprojekte investierten. Einige von ihnen wurden durch ihren Chef auch noch darin bestärkt: "Ich war mit 20 Jahren Leutnant an der Ostfront und verantwortlich für die ganze Elektronik eines Fliegerhorstes. Wieso sollt Ihr jungen Leute von heute also nicht auch für ein großes Projekt selbständig verantwortlich sein!"  
(Anmerkung: Der Verfasser kennt seit 15 Jahren den Universitätsbetrieb von heute und hat sich nie daran gewöhnen können, dass viele (besonders deutsche) Jungakademiker offenbar inzwischen vor neuen Ideen Angst haben.)

## 1975 - 1976 Unerklärliche Stagnation

Die PEARL Entwicklung schien am Ziel zu sein. Mehrere Compiler waren im Bau. Auf der Basis von PEARL wurden Firmen gegründet ("Spinoffs" nach heutiger Terminologie). Das Netzwerk der flankierenden Maßnahmen auf internationaler Ebene funktionierte. Doch plötzlich tauchten Probleme auf, die man für überwunden gehalten hatte.

Sie hatten sich schon 1973 angedeutet, als der während der "Bodensee-Klausur" erreichte Konsens dadurch in Frage gestellt wurde, dass eine Gruppierung eine nachträgliche Änderung des gemeinsamen Dokumentes durchsetzte mit dem Ziel, das Erscheinungsbild von PEARL doch mehr an PL/I anzugleichen.

Allmählich entstanden verschiedene “Subset-Arbeitskreise“, die in den Fachorganisationen zum Teil heftig konkurrierten. Dabei entstand der Eindruck, dass es meist nicht um technische Probleme ging, sondern um Firmenpositionen, die mit dem Fertigstellungsgrad der jeweiligen Compiler zusammenhingen. Diese Entwicklung führte schließlich dazu, dass der GMD die Meinungsführerschaft übertragen wurde, was zu einer praktisch völligen Neuerfindung von PEARL auf akademischer Basis führte (Zitat: “Wenn sich die Händler streiten, freut sich der Theoretiker.“)

Im April 1975 wechselte der PEARL-Koordinator beim Projekt PDV, der einer der PEARL-Pioniere gewesen war, auf eine Professur an der Fachhochschule Bielefeld. Als der Verfasser beim Leiter des Projekts PDV nach den Gründen fragte, erfuhr er, dass dies nicht freiwillig geschehen war. Die angegebene Begründung möchte er (auch heute noch) lieber nicht zitieren.

Keiner der PEARL-Entwickler der ersten Stunde konnte in der neu gegründeten deutschen Informatik mit seiner Arbeit promovieren. Sie mussten sich entweder anderen (theoretischen) Themen zuwenden oder den Fachbereich wechseln. Eine schöne Bemerkung machte die Runde: ”Das Establishment schließt seine Reihen“. Forschung auf dem Gebiet der Echtzeitdatenverarbeitung fand also wo anders statt. Dass heute von einigen “Kerninformatikern“ eine Verstärkung der Forschung auf dem Gebiet der Echtzeitdatenverarbeitung (= “embedded systems“) gefordert wird, kann unter diesem Gesichtspunkt von den Pionieren als späte Genugtuung betrachtet werden. Aber: “Lieber spät als nie.“

Ein ernster Schlag war schließlich der Rückzug des VDI/VDE. Die zuständigen Ausschüsse beschlossen auf eine Initiative der chemischen Industrie hin, statt PEARL in Zukunft “Prozeß-FORTRAN“ als Standard zu unterstützen. Die besondere Ironie dabei war, daß dieser Sprachdialekt aus den Echtzeitelementen von PEARL bestand. Diese waren vom “FORTRAN-Committee“ im Purdue Workshop in den USA für so gut befunden worden, daß sie zur internationalen Standardisierung vorgeschlagen wurden .....

## Eine zweite Analyse

- Durch die Förderung war plötzlich viel Geld im Spiel. Also wurden die Pioniere durch Menschen ersetzt, die den Umgang mit Macht und Geld gewöhnt waren oder ihn anstrebten. Das ist zunächst vom Prinzip her nicht immer ein falscher Ansatz. Man darf dabei nur nicht zu kritiklos vorgehen oder das “Ersetzen“ dem Zufall überlassen.
- Aus der heutigen Sicht des Verfassers, die sich vor allem auf seine spätere Tätigkeit im Management-Team der Ada-Entwicklung gründet, liegt die Hauptursache aber in einem für Standardisierungsprojekte grundsätzlich falschen Ansatz der deutschen Förderpolitik: der 50% -Förderung. Diese zwingt die geförderten Firmen zu erheblichen Eigeninvestitionen, wofür sie natürlich das Recht erhalten müssen, über die Verwendung der geförderten Entwicklungen frei zu entscheiden. Damit ist das Durchsetzen von Normen praktisch unmöglich.

In den USA wird in solchen Fällen ein anderes Modell verwendet: Die Regierung finanziert die Entwicklung einer Norm (wie z.B. die Programmiersprache Ada) mit allen dazugehörigen Hilfsmitteln (Compiler, Betriebssystem, etc.) vollständig und stellt sie der Industrie zur Verfügung. Im Gegenzug werden alle Firmen, die Regierungsaufträge erhalten wollen, gezwungen, diese Norm zu verwenden. Die Hilfsmittel erhalten sie kostenfrei als “Beistellung“. Wer diese Bedingungen nicht akzeptiert, erhält keine Regierungsaufträge. Der bereits erwähnte Koordinator für die PEARL-Entwicklung hatte damals beim Projekt PDV ein ähnliches Modell durchzusetzen versucht .....

## 1977 - 1978 Das Kaninchen und die Schlange

Nachdem das Verteidigungsministerium der USA angekündigt hatte, eine allgemeingültige genormte Programmiersprache für Echtzeitanwendungen zu schaffen, entstand in allen Gremien Unruhe. Obwohl die Arbeiten auf europäischer Ebene weiter gediehen waren, wurde von der europäischen Informatik, Regierungsvertretern und vielen Industrieleuten sofort angenommen, dass die Entwicklung aus den USA eigene Arbeiten überflüssig machen würden.

In Frankreich und England wurde diese US-Initiative zwar offiziell unterstützt und begleitet, die existierenden nationalen Projekte wurden aber als “backup“ beibehalten. Im Gegensatz dazu kündigten die deutschen offiziellen Stellen an, die Förderung von PEARL ganz einzustellen. Die Verhandlungstaktik von offizieller deutscher Seite in den internationalen Gremien war defensiv.

Im Jahr 1978 wurde dann der Verfasser als Beobachter dieser Sprachentwicklung in die USA entsandt. Die Förderung der Entwicklungsgruppe in Erlangen wurde eingestellt.

## Eine dritte Analyse

- Der Wind in der Politik hatte sich gedreht. Der schlagartig gewachsene Wohlstand ließ Dinge wie internationale Konkurrenzfähigkeit der deutschen Volkswirtschaft, Effizienzsteigerung, technologische Innovationsfähigkeit etc. plötzlich zweitrangig erscheinen. “Humanisierung der Arbeitswelt“, “Arbeitsplatzerhaltung“, “Selbstverwirklichung“ wurden zu den Forderungen der Stunde. Die Folgen erleben wir heute.
- Die Freizeitgesellschaft warf ihre Schatten voraus. So äußerte einmal ein höherer “Offizieller“ dem Verfasser gegenüber, als der sich etwas über die lückenhafte Präsenz deutscher Vertreter in der internationalen Standardisierungsszene und bei der EG-Kommission beklagte: “meine Freizeit als zweiter Cellist im Stadtorchester von xxxx ist mir wichtiger als die ewigen Meetings“ .....
- Selbst in der Industrie hatten sich eigenartige Vorstellungen von leicht verdientem Geld herausgebildet. So äußerte z.B. einmal ein hochrangiger Vertreter eines Chemiekonzerns während einer Sitzung eines VDI-Gremiums: “Wir verdienen viel mehr Geld mit Anlagenbau. Das bisschen Informationstechnik kaufen wir aus den USA zu.“  
Heute weist die BRD Rückstände in einigen wesentlichen Schlüsseltechnologien auf .....

## 1979 - 1980 Ein erneuter Versuch

Als der Verfasser aus den USA zurückkehrt, hat er einige wesentliche Dinge dazugelernt:

- Wie organisiert man derartige Projekte straff und effizient und bindet trotzdem alle “interessierten Parteien“ ein (siehe oben).
- Wie fördert und nutzt man die Kreativität von Entwicklern - und belohnt sie angemessen, anstatt sie als “Störenfriede“ zu entsorgen.
- “Auch der große Bruder kocht nur mit Wasser“: die Entwicklung von Ada führte zu einem ganz anderen Ergebnis als dem ursprünglich geplanten und wurde trotzdem für ihre Zwecke ein Erfolg /6/.

Konkret heißt dies, dass Ada nicht die universelle “Systemprogrammiersprache“ (wie z.B. “C“) ist, als die sie ursprünglich geplant worden war, sondern eine Echtzeitsprache, die ein eigenes Betriebssystem benötigt - wie eben auch PEARL.

PEARL war also voll konkurrenzfähig - nur eben einfacher, besser verständlich und preiswerter.

Auf der Basis dieser Überlegung engagierte sich der Verfasser im Vorstand des PEARL-Vereins, der versuchte, diese Tatsachen der Fachöffentlichkeit klarzumachen und damit vielleicht die Untergangsstimmung der offiziellen Seiten zu kompensieren.

Damals maßgebliche Personen der deutschen Informatik waren aber anderer Meinung - einer davon wollte nämlich den ersten Ada-Compiler der Welt bauen. Das BMFT stellte die Förderung von PEARL ein. Die EG “investierte“ viele Millionen ECU an Fördergeldern in “universelle Ada-Programmierungsumgebungen“, z.B. das “Portable Common Tool Environment“ (PCTE), die in der Praxis nie zum Einsatz kamen. Die entsprechende Abteilung derjenigen Weltfirma, die diese Entwicklung betrieb, wurde schon vor über 15 Jahren aufgelöst .....

## Noch eine Analyse

- Die BRD verdiente damals mit ihren “klassischen“ Industrien so gut, dass sich keiner der “Verantwortlichen“ klar machte, dass sie bereits ihr Kapital verzehrten - nämlich die Innovationen aus den 20er Jahren des 20. Jahrhunderts. Die heute maßgeblichen Technologien wurden als “Spielzeug“ und kommerzielle Randerscheinungen betrachtet. Die Folgen dieses Denkens erleben wir heute.

- Die deutsche Informatik verstand sich immer als Ableger der Mathematik und nicht als die hochkreative Denkwerkstatt und Experimentalwissenschaft, die sie in den angelsächsischen Ländern ist. Durch Nachprüfen, Systematisieren und Reproduzieren anderswo gemachter Erfindungen kann man aber keine Industrie am Laufen halten.
- Wenn man sich - wie der Verfasser - intensiv mit Geschichte befasst, sieht man, dass der Glaube, dass alles Heil aus dem Ausland komme, in Deutschland eine lange Tradition hat. Es gibt natürlich zwischendurch Aufwallungen von "Nationalismus". Diese gründen sich aber nicht auf ein gelassenes Selbstbewusstsein, sondern haben eher den Charakter von Verzweiflungstaten und sind deswegen dann meist destruktiv.

## 1981 "Noch ein Aussteiger"

Der Verfasser legte auch seine letzten Ämter nieder, weil er sich nicht mehr von Personen herumkommandieren lassen wollte, die zur PEARL Entwicklung wenig beigetragen, aber erheblich davon profitiert hatten und dann auch noch als die Erfinder galten. Er machte jedoch den Grund nicht öffentlich, um der Sache nicht zu schaden - und weil es ihm damals sowieso keiner geglaubt hätte. Sicher kommt es vor, dass Ereignisse, die zunächst als Unglücke aussehen, sich nach Jahren als Glücksfälle herausstellen. In diesem Fall kann der Verfasser den Umständen, die ihn zu dieser Entscheidung zwangen, bis heute wenig Positives abgewinnen. Das mag vielleicht daran liegen, dass er noch nicht den rechten Grad an Weisheit des Alters erreicht hat.

## 1982 - 2005 Totgesagte leben länger

Darüber müssen die berichten, die die gute Sache am Leben erhalten haben.

## Danksagung

Der Verfasser dankt Peter Holleczeck ganz herzlich für die sorgfältige Durchsicht dieses Beitrages und die vielen Verbesserungsvorschläge. Er bittet ihn aber auch vielmals um Verständnis dafür, dass er nicht alle befolgt hat.

## Literaturhinweise

- /1/ J. Brandes, S. Eichertopf, P. Elzer, L. Frevert, V. Haase, H. Mittendorf, G. Müller, P. Rieder: PEARL, The Concept of a Process and Experiment Oriented Programming Language; elektronische datenverarbeitung, 10 (1970), S.429-442
- /2/ PL/I-language-specifications, IBM C28/65-71
- /3/ Wijngarden, Mailloux, Peck, Koster: Report on the Algorithmic Language ALGOL 68; Num. Math. 14 (1969), S. 79-218
- /4/ K.H. Timmesfeld, B. Schürlein, P. Rieder, K. Pfeiffer, G. Müller, K. Kreuter, P. Holleczeck, V. Haase, L. Frevert, P. Elzer, S. Eichertopf, B. Eichenauer, J. Brandes: PEARL - Vorschlag für eine Prozess- und Experimentautomatisierungssprache. Gesellschaft für Kernforschung mbH, Karlsruhe, PDV-Bericht KfK-PDV 1, April 1973
- /5/ P. Elzer: Abschlußbericht NDV 13 und Zwischenbericht NDV 15 vom 3.5.1974
- /6/ P. Elzer: Ergebnisse der Mitarbeit am Projekt Programmiersprache Ada; Gesellschaft für Kernforschung mbH, Karlsruhe, PFT-Bericht KfK-PFT 19, 1982

P. Elzer  
 Paradiesstr. 4  
 80538 München  
[elzer@ipp.tu-clausthal.de](mailto:elzer@ipp.tu-clausthal.de)

### 3 Abstracts von Abschlussarbeiten

Auf der FGL-Sitzung am 11.5.2005 wurde die Idee geboren, Abstracts von Thesarbeiten und Dissertationen aus den Einrichtungen unserer Fachgruppenmitglieder mit starkem Bezug zu PEARL oder Echtzeitproblemen als festen Bestandteil aufzunehmen. Damit verfolgen wir das Ziel, die Breite der aktuellen Forschungsarbeiten in diesem spannenden Gebiet darzustellen.

Der Umfang der Abstracts sollte ca 15 Textzeilen umfassen. Die Einreichung soll direkt an die Redaktion per E-Mail mit den Angaben über Verfasser, Hochschule, Kontaktadresse (EMail) und evtl. eine URL für weitere Informationen erfolgen. Als Textformat wird seitens der Redaktion  $\LaTeX$  bevorzugt.

Die Leser der PEARL-News werden gebeten, auf geeignete Arbeiten zu achten und die betreffenden Abstracts mit den ergänzenden Daten an die Redaktion zu übermitteln.

**Investigation of the IEEE 802.11e Standard and its Performance  
in an industrial Environment based on a Wireless LAN Access Point  
with Embedded Linux**

Henning Trsek

Fachhochschule Lippe und Höxter

Fachbereich Elektrotechnik und Informationstechnik

{henning.trsek@fh-luh.de }

<http://www.network-lab.de>

Die Verwendung von IEEE 802.11 *Wireless Local Area Networks* in unterschiedlichen Umfeldern wie z. B. im Small Office/Home Office (SOHO) Bereich, in Büroanwendungen und in der Industrie bietet erhebliche Vorteile. Diese beinhalten eine größere Flexibilität, reduzierte Installations- und Wartungskosten sowie eine gesteigerte Mobilität. In dem Bereich der industriellen Automatisierung ergeben sich jedoch harte Anforderungen bezüglich eines deterministischen Austauschs von Informationen, um Steuerungsaufgaben in Echtzeit ausführen zu können. WLANs die auf IEEE 802.11 basieren sind wegen ihres Kanalzugriffs nicht deterministisch und können deshalb nicht garantieren, dass ein bestimmter Frame sein Ziel in der gewünschten Zeit erreicht. Als Lösungsansatz für drahtlose Echtzeit-Anwendungen bietet der IEEE 802.11e Standard die Möglichkeit eines priorisierten Kanalzugangs mit der *Enhanced Distributed Coordination Function* (EDCF), um zeitkritischen Anwendungen den Zugang zum Medium zu ermöglichen, wenn sie in benötigen. In dieser Arbeit wird die Leistungsfähigkeit der EDCF in industriellen Anwendungen mit zeitkritischem Datenverkehr hinsichtlich der auftretenden Verzögerungen sowohl in einem statischen Aufbau als auch in einer realistischen Roboter Zelle bewertet. Hierfür wurde ein Access Point (AP) für industrielle Umgebungen und Anwendungen entwickelt und aufgebaut, der auf embedded Linux basiert. Der AP implementiert die EDCF vollständig und bietet somit die Möglichkeit Datenverkehr zu priorisieren. Für die Bewertung wurde er in ein Testaufbau integriert, der aus einem SPS System mit verteilten I/O Knoten bestand und auf PROFINET basierte und diente als Ersatz für eine drahtgebundene Verbindung.

**A generic UML-Modeling approach for Ethernet control networks  
and formal verification of their real-time capability**

Daniel Witsch

Bergische Universität Wuppertal

Lehrstuhl für Automatisierungstechnik

Fachbereich Elektrotechnik, Informationstechnik und Medientechnik

{witsch@uni-wuppertal.de}

Growth in Internet use across all sectors of the economy will inevitably lead to controlling distributed systems by means of communication devices that make use of the IP protocol the Ethernet standard.

The physical architecture and the behavior of the networks obtained are quite different of those of a specific fieldbus. The use of IP protocols and Ethernet implies to add active components in the network such as switches. Moreover, IP-communication is based on client-server relationships in contrast to a specific fieldbus where often master-slave structures are used.

So, first of all common architectural structures and behavior of Ethernet-networks (Modbus, Profinet) used in the field of automation need to be investigated.

These reusable artifacts of knowledge should be collected and structured with regard to an easy and flexible application in architecture modeling with the UML. The pattern mechanism needs to be evaluated for this purpose.

Using these techniques concept for generic modeling of industrial Ethernet architectures with the UML should be elaborated. A model based on this concept must be able to serve as a flexible extendable basis for various kinds of analysis and generation of different target data.

Moreover, the IP protocol however has not been specifically intended for industrial control applications. A number of studies and investigations have thus become necessary to evaluate the performance of the control system in order to check if it meets the requirements of industrial control applications. The second part of this master's thesis evaluates the implementation of a formal verification of the real-time capabilities for a subset of the possible network-architectures using a timed model-checker, e.g. UPPAAL, which has to be interfaced by the UML-tool or the model respectively.

This modelling and analyzing approach should be validated using a lab model as a case study.

**Entwurf und Implementation eines fehlertoleranten Reglers nach IEC 61131-3  
zur verlässlichen Regelung eines chemischen Prozesses.**

Jan F. Westerkamp  
Universität Bremen  
Fachbereich 1 / Physik und Elektrotechnik  
Institut für Automatisierungstechnik  
{wes@biba.uni-bremen.de }

Diese Diplomarbeit ist Teil des Projekts *Automatisierung chemischer Reaktoren*. Ziel des Projekts ist die fehlertolerante Regelung eines Durchflussrührkesselreaktors (continuous stirred tank reactor, CSTR). Dazu werden zertifizierbar korrekte fehlertoleranzgerichtete Komponenten benötigt.

Ziel dieser Diplomarbeit ist es, einen hierarchisch aufgebauten, fehlertoleranten Regler um weitere fehlertoleranzgerichtete Funktionsblöcke zur Kommunikation mit Sensoren und Aktoren zu erweitern. Weiterhin wird für die Fehlererkennung und -lokalisierung eine 1-Beobachter-Lösung entworfen. Implementiert werden die Funktionsblöcke in den Sprachen FBD (Funktionsblockdiagramm) und ST (strukturiertes Text) des Standards [IEC 61131-3].

Aufbauend auf der Arbeit von [Laarz, 2002] wird der dort portierte Regler für einen Durchflussrührkesselreaktor um die noch fehlenden Kommunikationsfunktionsblöcke erweitert. Bei der Portierung wurde die Hardwareanbindung der Sensoren und Aktoren zunächst nicht berücksichtigt, da sich diese bei den beiden speicherprogrammierbaren Steuerungen (SPS) nur auf sehr unterschiedliche Weise ansprechen und verwenden lassen. Außerdem werden von der PEARL-SPS, und der neuen  $\mu$ -SPS unterschiedliche CAN-Bus-Module verwendet, deren Kommunikationsprotokolle verschiedenen Standards folgen. Eine einfache Portierung der Funktionsblöcke ist deshalb nicht möglich.

Die Detektoren des Grundregelkreises entsprechen den Ausgängen der Sensor-Kommunikationsfunktionsblöcke. Diese liefern nicht nur den eigentlichen Messwert sondern signalisieren außerdem noch den Status des Sensors. Der Zustand der Aktoren wird zur Zeit nicht überwacht.

Nach dem Überblick über die Zustände, in denen der Prozess gefahren werden kann, und möglichen Fehlertoleranzmaßnahmen werden zwei Ansätze für die Kommunikationsfunktionsblöcke diskutiert. Der erste Ansatz folgt dem Standard [IEC 61131-3] und kann wegen Einschränkungen durch die  $\mu$ -SPS und fehlender Bibliotheken für die Entwicklungsumgebung CoDeSys nur andeuten, wie die Kommunikation mit *direkt dargestellten Variablen* ablaufen könnte. Der zweite Ansatz benutzt Funktionsblöcke, die dem Standard [CiA DS-405] angelehnt sind, um problemorientierte Funktionsblöcke zum Ansprechen von DA- und AD-Wandlern zu implementieren.

Die Funktionsblöcke aus dem zweiten Ansatz werden dann zusammen mit einem Beobachter, der zwei Messgrößen verwendet, um die Temperatur und Konzentration zu rekonstruieren, in den Regelkreis aus [Laarz, 2002] integriert.

Abschließend wird gezeigt, dass durch diese Ergänzungen nun ein fehlertoleranter Regler inklusive der nötigen Prozessperipherie zur Verfügung steht, der auch beim Ausfall von Sensoren oder deren analytischen Redundanz in der Lage ist, den Prozess sicher zu führen.

**Design and validation of divergence detection and handling  
for an observer as dependability component for the control of a chemical process  
using CoDeSys-Platform**

S. B. Kaja

Universität Bremen  
Fachbereich 1 / Physik und Elektrotechnik  
Institut für Automatisierungstechnik

The topic of this Master Thesis originates from the interdisciplinary project “Automation of Chemical Reactors“, guided by the Institute of Automation (FB1, Prof. G.Thiele) and the Institute of Applied and Physical Chemistry (FB2, Prof. G.Schulz-Ekloff). The Master Thesis contributes to real-time software development for the control of automation systems using state-observer as analytic redundancy to increase process-availability in case of sensor faults. As is often the case in the control of chemical processes, the observation-model is nonlinear and the design of the correction feedback is eventually based, as in this thesis, on a linearization of this model. If no proof of observer-stability is available, possible divergence of the observer has to be supervised, in order to guarantee, at least, the safety of the controlled process, by avoiding the use of non valid reconstructions as “analytic redundancy“. Furthermore, in addition to safety, graceful degradation should be the aim in order to keep the process-availability as long as possible.

Although modern real-time languages, based on appropriate real-time operating systems, comprise constructs for the detection and handling of arithmetic overflows, this form of detection is, in general, often too late to initiate successfully means of graceful degradation. The consequence of this is, therefore, shut down of the controlled process, so that it is of importance to predict a possible divergence early enough, possibly accepting some wrong decisions, if this leads only to minor decrease of reliability, but to a significant increase of availability.

The main contribution of the thesis is the simulative analysis of suitable strategies based on the parameterization of a given divergence-detection logic, using the magnitudes of reconstruction errors and their (approximated) derivatives, respectively, taking into account possible sensor faults (using a re-configurable observer [1]) in the case-study of a Fuzzy-controlled decomposition of hydrogen-peroxide. Of special interest is the technique to use a form of “forced reproducible divergence” [2] in pre-defined scenarios where a divergence-tendency will be assumed. On the basis of the different analyzed scenarios, appropriate for the validation of process-dependability, rules of logic-parameterization will be deduced for practical application.

- [1] G. Thiele, Th. Laarz, R. Neimeier, G. Schulz-Ekloff, L. Renner, E. Wendland (2003). Development of verifiable-by-inspection real-time control software for a dependable chemical process. In M. Staroswiecki, N. Wu (Eds.): Fault Detection, Supervision and Safety of Technical Processes. Proc. SAFEPROCESS 2003 (IFAC), Washington, June 9-11, 2003, Vol. 3, pp. 897-902. Elsevier, ISBN 0-08-044011-8.
- [2] G. Selvaganapathy (2004). Design of observer-divergence detection and removal as dependability function for fault-tolerant control of a chemical process. Master Thesis, Universität Bremen, FB 1.