

Steuerung von Analyseautomaten und Labormanagement im analytischen Labor



Reinhold Schäfer

„Wiesbadener Computerintegriertes Labor (WICIL)“ entwickelt Laborintegrationssoftware

Komplexer werdende Abläufe in chemischen Labors, knapper werdende personelle Ressourcen und steigende Analysenzahlen erzwingen zunehmend einen höheren Grad an Automatisierung. Hinzu kommen insbesondere in der Nahrungsmittel- und pharmazeutischen Industrie strenger werdende behördliche Anforderungen insbesondere auch auf dem Gebiet der Softwareunterstützung bei automatisierten Prozessen. Im Wiesbadener Computerintegrierten Labor (WICIL) am Fachbereich Informatik der Fachhochschule Wiesbaden entwickeln nunmehr seit 11 Jahren eine Gruppe von Professoren unter Leitung von Prof. Dr. Reinhold Schäfer mit ihren Studenten Automatisierungssoftware. Gefördert wird dieses Projekt aus Forschungsmitteln des Landes Hessen sowie der Industrie. Die Teilprojekte – meist Diplom- und Projektarbeiten – werden in Kooperation mit namhaften Forschungsinstitutionen und ca. 20 Industrieunternehmen durchgeführt. Hierbei sind besonders zu erwähnen das National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, die US-National Labors in Oak Ridge, TN (ORNL), und Los Alamos, NM (LANL) sowie die Hewlett Packard bzw. Agilent GmbH und die Creon-Labcontrol AG.

Keywords

Integration, Laborautomation, LIMS, Grafische Programmierung, Geräte-Plug-and-Play

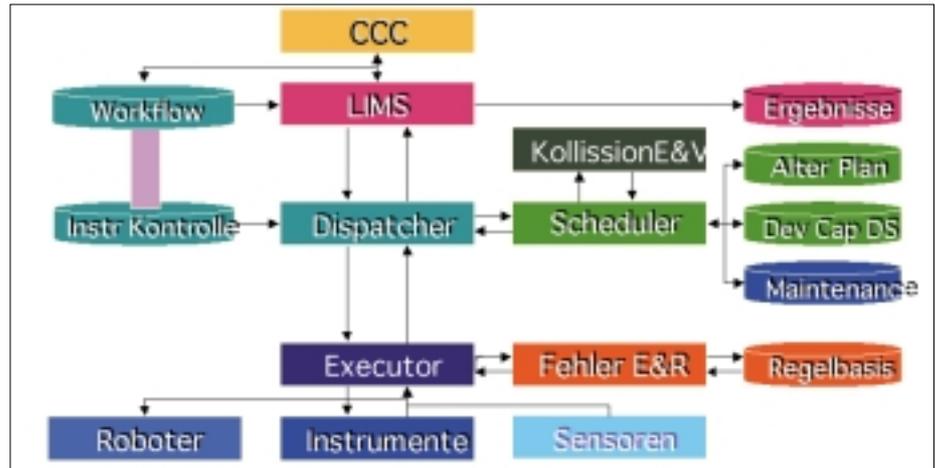


Abb. 1: WICIL Controlling Architektur

Projektüberblick

Ziel des Projektes ist die Steuerung eines Labortisches im unterbrechungsfreien 24-Stundenbetrieb mittels einer einheitlichen System-Architektur (Abb. 1) [1] [4] [10] [16]. Beliebige Laborabläufe werden über einen grafischen Editor erstellt und nach der Zuordnung der zu messenden Proben automatisch bearbeitet. Die Abläufe können dabei in beliebiger Granularität beschrieben werden, von der Durchführung eines gesamten Tests „Content Uniformity“ bis zum feinsten Steuerungs-

detail des Öffnens einer Waagentür oder Reaktionen auf Fehlerzustände. Ein dynamischer Scheduler optimiert die Probenbearbeitung auf dem Labortisch. Er legt dabei eine optimale Bearbeitungsreihenfolge der unterschiedlichen Aktionen fest, so dass möglichst viele Proben parallel abgearbeitet werden ohne sich gegenseitig zu behindern. Zusätzlich werden Ad-Hoc-Analysen und Meldungen bzw. Störungen auf dem Labortisch durch dynamisches Re-Scheduling berücksichtigt [2] [9]. Der weit vorausberechnete Plan kann Kollisionen zwischen Geräten auf

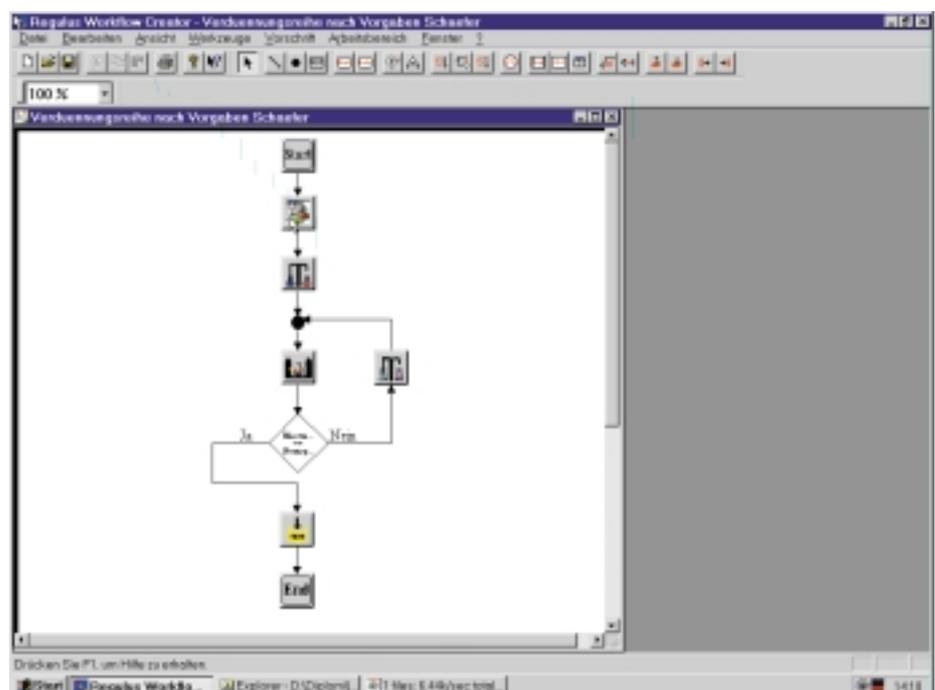


Abb. 2: REGULUS Bildschirm

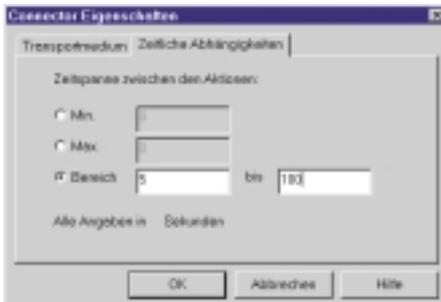


Abb. 3: Connectoreigenschaften

dem Labortisch beinhalten. Diese werden jedoch bereits bei der Planerstellung von einem Kollisionserkennungsmodul aufgezeigt und zur Korrektur der betreffenden Bewegungsabfolgen verwendet. Schließlich wird ein Laborleitstand für das Monitoring und die Simulation ggf. mit generischen Geräten eingesetzt. Meldungen und Fehler werden über eine regelbasierte Ereignis- und Fehlerbehandlung mit grafischer Regeldefinition, Faktenkonditionierung und Simulation bearbeitet.

Grafische Erstellung von Arbeitsabläufen mit Regulus

Arbeitsabläufe dienen im Labor dazu, einer Mitarbeiterin oder einem Mitarbeiter Anweisungen für die exakte Schrittfolge einer Serie von Arbeitsschritten zu geben. Beim Eintreten besonderer Bedingungen muss die Möglichkeit der Durchführung alternativer Schrittfolgen sowie der Ad-Hoc-Prüfung bestimmter Zustände gegeben sein. Im vorliegenden Fall eines automatisierten Labors setzen sich Arbeitsabläufe ebenfalls aus Aktionen auf Geräten inklusive der Steuerung der Gerätefunktionen und dem Übertragen von Ergebnisdaten zusammen. Häufig werden lediglich manuelle Vorgänge auf Automaten (Roboter, Schraubapparate etc.) übertragen. Zunehmend werden die Workflows „maschinengerecht“ adaptiert oder gänzlich neu organisiert. Wichtig ist das Einbeziehen von Sensoren, mit denen die unterschiedlichen Zustände innerhalb eines Arbeitsablaufs auf dem Labortisch oder die Probenorte und -zustände festgestellt, ausgelesen und für Auswahl und Konfiguration der Folgeaktionen verwendet werden können. Außerdem müssen Ergebnisdaten verrechnet und gespeichert werden. Als Operatoren müssen Abfragen, Schleifen und Verzweigungen zur Verfügung stehen. Zur besseren automatisierten Optimierung der Aktionsreihenfolgen müssen schließlich dem System Operatoren für Parallelisierung zur Verfügung stehen (Concurrent Operator).

Als Paradigma zur Workflow-Erstellung wird ein erweiterter Programmab-

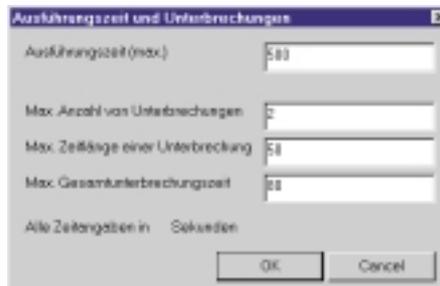


Abb. 4: Bedingungen für Multi-Ressourceninstrumente

laufplan verwendet (Abb. 2) [13] [15]. Mit Symbolen versehene Rechtecke (Service Units) repräsentieren Geräte, Sensoren, Aktoren, Formelprozessoren, Speicheroperatoren sowie Quellen und Senken von Proben, Material und Behältern. Sie werden zur Veranschaulichung zeitlicher Reihenfolgen mit Pfeilen verbunden. In Service Units können Variablen definiert und verarbeitet werden. Während der Erstellung des Arbeitsablaufs können vom Softwarepaket Regulus mit speziellen Dialogen gezielt Parameter vom Benutzer erfragt werden (z.B. Rüstzeiten, Dauer von Aktionen, maximale Delays zu den Folgeaktionen, Dauer und Anzahl von Unterbrechungen der Aktion). Gesamte Workflows können zu Makros zusammengefasst, mit Parameterübergaben versehen und in Bibliotheken zur Wiederverwendung gespeichert werden. Diverse Konfigurationsdialoge vervollständigen schließlich die Palette der Regulus-Funktionalitäten (Abb. 3 und 4) [16].

Optimierendes Scheduling und Re-Scheduling

In der Vorschriftenverwaltung werden neben den Abläufen auch die Steuerparameter der einzelnen Aktivitäten erfasst. Hierzu gehören u. a. deren Rüstzeit, Dauer (Abb. 3) und die zeitlichen Randbedingungen zur Folgeaktivität [2] [9] [13].

Der Schedulingsoftware fällt nun – je nach vorgewählter Optimierungsart – die Aufgabe zu, die einzelnen Aktivitäten so auf die benötigten Ressourcen zu verteilen, dass entweder der Durchsatz der Analysen möglichst hoch wird oder die Einzelanalysen möglichst schnell bearbeitet werden [3] [11].

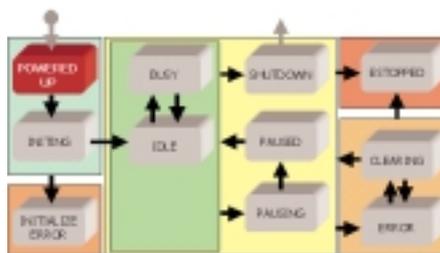


Abb. 6: Standard-Zustandsdiagramm (angelehnt an ASTM E1989-98 Standard)

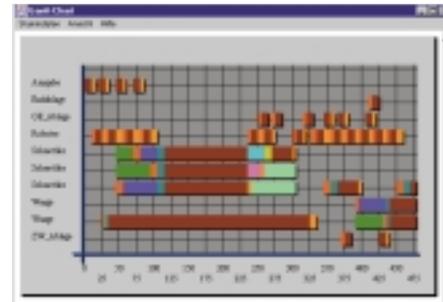


Abb. 5: Gantt-Chart

Beim Scheduling müssen diverse chemische Randbedingungen beachtet werden. So dürfen insbesondere Aktivitäten mit wohldefiniertem Zeitverhalten, z.B. Inkubationen, nicht unterbrochen werden, da sonst die Analysenergebnisse verfälscht oder gar die Proben unbrauchbar werden.

Wesentliche Zeitersparnisse können erreicht werden, wenn die Eigenschaft von Mehrpositionsinstrumenten genutzt werden kann. So wird es z. B. möglich, mehrere Proben mit unterschiedlichen Bearbeitungszeiten bei identischem Schüttelverhalten gleichzeitig auf einem Schüttler zu bearbeiten. Der Schüttelvorgang muss dann lediglich zu den vorgegebenen Zeiten angehalten und Proben vom Schüttler genommen werden (Abb. 4). Diese Eigenschaft darf jedoch nur eingesetzt werden, wenn sie vom Bediener bei der Probenaufgabe explizit erlaubt wurde.

Im Falle eines Fehlers auf dem Labortisch – u. a. durch Ausfall eines Instrumentes – oder des Einschubes einer Ad-Hoc-Probe in den Arbeitsablauf, errechnet der Scheduler einen neuen optimalen Plan unter Berücksichtigung der noch in Bearbeitung befindlichen Proben sowie der Rechenzeit für den Plan. Dabei muss natürlich einerseits sichergestellt werden, dass aktuell auf einem Gerät befindliche Proben möglichst problemlos weiter bearbeitet werden, andererseits Maximalzeiten im Workflow und nicht unterbrechbare Sequenzen eingehalten werden.

Zur Kontrolle können die Plandaten des Scheduler mittels Gantt-Chart visualisiert werden (Abb. 5). Standardmäßig werden die errechneten Pläne an den Executor weitergereicht, der sie gerätespezifisch im – durch CORBA unterstützten – verteilten System ausführt [3].

Der Scheduler greift auf die Beschreibung der Geräte, deren Geometrie und Abhängigkeiten sowie weiterer Ressourcen im System Capability Dataset (SCD) zu [6] [7] [8] [11]. Außerdem muss er notwendige vorbeugende Wartungszeiten mit einplanen, um einem Systemausfall vorzubeugen. In diesem Fall müssen Aktivitäten auf gleichartige Geräte

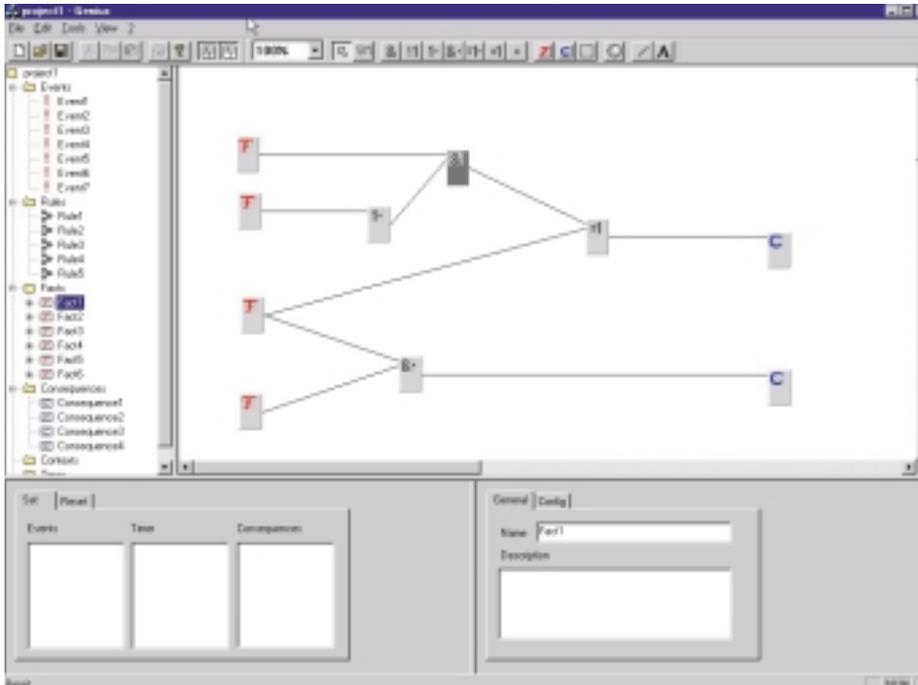


Abb. 7: Grafische Definition von Regeln in Genius

ausweichen, oder die entsprechenden Proben müssen bis nach Abschluss der Wartungsarbeiten „geparkt“ werden.

Der Scheduler steht mit einem noch zu integrierenden Bahnplanungsmodul in Verbindung, das Kollisionen mit stationären und beweglichen Hindernissen im voraus erkennen und einen alternativen Polygonzug für den eingesetzten Roboter errechnet.

Zusammen mit dem Dispatcher, der alle Aktionen automatisch monitort, und dem Bahnplaner gehört der Scheduler zu den wichtigsten Komponenten in der WICIL Controlling Architecture (WCA) [4].

Geräteansteuerung, Rohdaten- und Ergebnisbearbeitung

Die Ansteuerung der Geräte wird durch den Executor vorgenommen, der als Eingabe vom Scheduler den sog. Execution Plan erhält. Dieser fasst zeitbezogen die auf den einzelnen Geräten durchzuführenden nicht-unterbrechbaren Sequenzen (NIS) von Aktivitäten und die Einzel-Aktivitäten zusammen. Als wichtige Funktion fällt dem Executor die exakte Terminierung der Aktivitäten und die gerätebezogene Überwachung der Zeitvorgaben zu. Wird eine Zeitvorgabe nicht eingehalten, muss dies der Executor dem System mitteilen, so dass jede betroffene Komponente entsprechend reagieren kann.

Prinzipiell ist die Wicil Controlling Architecture (WCA) so ausgelegt, dass nur im äußersten Notfall das System angehalten werden muss. Vorhersehbare Ausnahmezustände sollen automatisiert behoben werden, mit dem Ziel, möglichst

keine Proben unbrauchbar zu machen oder Rohdaten bzw. Ergebnisse zu verlieren (siehe dazu das Modul Genius).

Die Ansteuerung der Geräte erfolgt – wie bereits erwähnt – über CORBA. Dadurch kann u. a. erreicht werden, dass auch Geräte an anderen Rechnern als dem Executor-Rechner zum Betrieb des Labortisches einbezogen werden können.

Die Geräte werden nach einem einheitlichen Mechanismus angesteuert, der weitgehend an den ASTM-Standard Lecis (Laboratory Equipment Communication Interface Standard, ASTM E1989-98) angelehnt ist (Abb. 6) [5] [11] [12] [14]. Zu jedem Zeitpunkt befindet sich ein Gerät in einem wohl definierten Zustand, von dem aus nur bestimmte Übergänge in andere Zustände erlaubt sind (Pfeile in der Abb. 6).

Nach dem Einschalten (Powered up) und Initialisieren (Initing) läuft das Gerät in einen Wartezustand (IDLE), aus dem heraus der nächste Befehl im Execution Plan ausgeführt werden kann (BUSY). Die für ein bestimmtes Gerät verfügbaren Befehle wurden bereits bei der Erstellung der Vorschrift für den aktuellen Probenotyp aus dem System Capability Dataset (SCD) entnommen und in den Workflow eingefügt. Jetzt werden sie zur Ausführung gebracht.

Neben den „normalen“ Befehlen gibt es Ausnahmebehandlungen. So können aus allen grün unterlegten Zuständen das Pausieren und ein Shutdown eingeleitet werden. Aus allen Zuständen (gelb unterlegt und Error) kann ein Nothalt (Estop) erfolgen, aus dem jedoch nur manuell wieder angefahren werden kann.

Automatisierte Behandlung von Ereignissen und Ausnahmezuständen (Genius)

Ausnahmezustände werden heute hart kodiert im Programmcode hinterlegt. Geräte können bei mitgelieferten Steuer- und Auswertprogrammen die Fehlerbehandlung „mitbringen“. Ausnahmebehandlungen, die von der Interaktion zwischen Labortischkomponenten veranlasst werden, sind nicht fest programmierbar, weil Geräte von unterschiedlichen Herstellern kommen können. Die Ausnahmebearbeitung erfolgt in der WICIL Controlling Architecture ausgelagert über das Expertensystemmodul Genius (Generic Events Network Improves Unattended Systems) [17].

Die Ausnahmesituationen werden durch Events repräsentiert, die allen registrierten Systemkomponenten zur Verfügung stehen. Kombinationen von Events bilden Regeln, die „feuern“, so fern alle notwendigen Fakten an deren Eingang erfüllt sind. Beliebige Systemfunktionalitäten, die Events produzieren, können mit dem WICIL-Paket Genius bearbeitet werden [20] [22].

Die Regeln werden grafisch erstellt (Abb. 7). Mehrere Events können zu Fakten zusammen gefasst, mit Zeitbedingungen versehen und über boolesche Operatoren verknüpft werden. Die Fakten können durch Events oder Konsequenzen – auch Konsequenzen anderer Regeln – zurück gesetzt werden. Die Konsequenzen sind ebenfalls Events, mit denen u. a. Funktionen gestartet, Dialoge angestoßen werden können. Somit können z.B. im Fehlerfall geeignete Maßnahmen ausgelöst werden.

Zusammenfassung und Ausblick

Prinzipiell konnte inzwischen gezeigt werden, dass ein unterbrechungsfreier automatisierter Betrieb des Labortisches realisierbar ist. Alle beschriebenen Module sind lauffähig, allerdings noch nicht komplett integriert. Eine Ursache liegt dabei in der noch nicht vollständig umgesetzten CORBA-Spezifikation. Besonders wichtig ist die Funktionsfähigkeit des System Capability Dataset, mit dem erstmals eine Standardisierung von Laborgeräteschnittstellen möglich wird, ohne die Funktionsvielfalt unterschiedlicher Geräte einzuschränken. Durch den Metabeschreibungsansatz, d. h. dem Festlegen der Notationsbeschreibung und nicht der Notation selbst, wird dies gewährleistet. Dabei wird eine Standardisierung angestrebt. Wünschenswert wäre für das Wiesbadener Computeringegrierte Labor eine weitergehende

Unterstützung von Geräteherstellern und Anwendern. Nur so werden wir die notwendige umfassende Verifizierung der Ideen erreichen können.

Schließlich sei an dieser Stelle auch für die materielle und ideelle Unterstützung durch das Land Hessen sowie insbesondere der Hewlett Packard Deutschland GmbH und der Creon-Labcontrol AG gedankt. Sie ermöglichten erst die Durchführung der beschriebenen Arbeiten.

Literatur

- [1] R. SCHÄFER, LIMS als Komponente der Informationsarchitektur eines Unternehmens, CHEManager 4 (1996) 24-26
- [2] R. SCHÄFER, B. STIEHL, Expert System for Dynamic Scheduling Supporting Sample Preparation Constraints, Poster, Int'l. LabAutomation Conference, San Diego, California, USA, 1997
- [3] R. SCHÄFER, M. HAGEMEIER, M. HOFMANN, K. REUBERT, A Highly Distributed Laboratory Automation System Using the CORBA Standard as Integration Platform, Int'l LabAutomation98 Conference, San Diego, CA, USA, 1998
- [4] R. SCHÄFER, T. J. BEUGELSDIJK, J. ELLING, Architectural Issues for Total Lab Automation, Int'l LabAutomation98 Conference, San Diego, CA, USA, 1998
- [5] B. SCHÄFER, T. STAAB, R. SCHÄFER, J. ELLING, Standardized control of laboratory equipment with the American Society for Testing and Materials (ASTM) Laboratory Equipment Control Interface Specification (LECIS), Int'l LabAutomation98 Conference, San Diego, CA, USA, 1998
- [6] C. PIOTROWSKI, T. RICHTER, R. SCHÄFER AND G. W. KRAMER, System Capability Dataset for Laboratory Automation System Integration, Journal of Analytical Laboratory Automation, 12 (1998)
- [7] O. BORCHERT, G. W. Kramer and R. Schäfer, Dealing with Result Data Automatically - What Does a Capability Dataset Need to Contain?, Int'l LabAutomation99 Conference, San Diego, CA, USA, 1999
- [8] T. TAUBER, G. W. Kramer and R. Schäfer, Device Integration Using the Device Capability Dataset Concept", Int'l LabAutomation99 Conference, San Diego, CA, USA, 1999
- [9] R. SCHÄFER, Advanced Scheduling and its Impact on Laboratory Integration, Int'l LabAutomation99 Conference, San Diego, CA, USA, 1999
- [10]– [22] Literaturangaben beim Autor

Der Autor

Prof. Dr. Reinhold Schäfer

Dr. rer. nat., Dipl.-Physiker Studium an der Universität Mainz, Industrietätigkeiten in der Pharmazeutischen Industrie, dort zuletzt Gruppenleiter für wissenschaftliche Informationsverarbeitung in einem 24.000 Mitarbeiter-Unternehmen, Realisierung diverser Automatisierungsprojekte. Seit 1989, Professor für Datenbanken, Künstliche Intelligenz und Automatisierung an der Fachhochschule Wiesbaden

Fachhochschule Wiesbaden
University of Applied Sciences
Fachbereich Informatik,
Kurt-Schumacher-Ring 18
65197 Wiesbaden
Fax 0611 / 9495-210
schaefer@informatik.fh-wiesbaden.de