



Nachgefragt bei

Dipl.-Ing. Bernd  
Hildebrandt, Vertriebs-  
leiter von Sigmatek



Sigmatek setzt bereits seit dem Jahr 2000 mit seinem Engineering Tool »LASAL« auf die objektorientierte Programmierung mit grafischer Darstellung und Client-Server-Kommunikation. Und war damit einer der ersten Hersteller am Automatisierungsmarkt, der die Vorteile der Objektorientierung erkannte und für seine Entwicklungsumgebung zu nutzen wusste. Heute wird die Software als ausgereiftes »All-in-one«-Tool beworben, das modular aufgebaut alle Funktionalitäten zur Lösung von Automatisierungsaufgaben bietet. Wie das im Detail aussieht, fragte Austromatisierung bei Dipl.-Ing. Bernd Hildebrandt, Vertriebsleiter von Sigmatek Österreich, nach.

**Austromatisierung:** Herr Dipl.-Ing. Hildebrandt, was unterscheidet »LASAL« von Programmier- bzw. Engineeringssystemen anderer Hersteller?

**Dipl.-Ing. Bernd Hildebrandt:** Sehr vieles! Die Idee, ein einziges Software-Tool zu schaffen, das modular als Baukastensystem angelegt sämtliche Engineeringaufgaben der Automatisierungstechnik vereint – von der Steuerungsprogrammierung über die Projektierung der Visualisierung, der Antriebstechnik und der Safety-Komponenten bis hin zur Realisierung von Servicefunktionen wie Fernwartung und Diagnose, hatte man im Hause Sigmatek schon sehr früh. Bereits vor über zehn Jahren wurde »LASAL« am Markt eingeführt und verwendete dabei von Anfang an die objektorientierte Programmierung – damit war Sigmatek der erste Hersteller in der Automatisierungstechnik und setzte neue Maßstäbe für Modularität und Wiederverwend-

## „All-in-one ist bei uns nicht nur ein Slogan“

barkeit. Wir stellen dem Maschinenbauer eine durchgängige Entwicklungsumgebung nach IEC 61131-3 – erweitert um die Objektorientierung – zur Verfügung, die ihm einerseits größtmögliche Flexibilität beim Umsetzen individueller Kundenwünsche bietet, und andererseits die Engineeringzeiten und somit -kosten senkt. Während andere gerade erst beginnen, ähnliche Sys-

„Sigmatek setzt seit über zehn Jahren auf die objektorientierte Programmierung.“

teme am Markt einzuführen, ist »LASAL« bereits ausgereift und seit mehr als zehn Jahren markterprobt. Die Unterschiede stecken also sowohl im historischen Ansatz als auch in einer Vielzahl von praktischen Details – in der Software selbst ebenso wie drum herum.

**Austromatisierung:** Wodurch genau senkt die objektorientierte Programmierung die Engineering-Zeiten?

**Dipl.-Ing. Hildebrandt:** Ich würde sogar von objektorientierter Projektierung sprechen! Der wesentliche Punkt ist die Wiederverwendbarkeit von einmal gemachter Arbeit – spricht der Ent-

wickler kann einmal erstellte Klassen immer wieder verwenden, auch um sie zu kopieren, zu vererben etc. Maschinenkomponenten werden durch Softwarekomponenten – sprich Objekte – nachgebildet. Hinter jedem Objekt steht eine Klasse, die der Bauplan für ein Objekt ist und den Programmcode und die Datenelemente definiert. Eine solche Klasse übernimmt eine bestimmte Aufgabe, wie beispielsweise die Steuerung eines Antriebes oder die Messung und Auswertung der Temperatur. Durch die grafische Darstellung wird die Komplexität des Programms gekapselt – das heißt, der Programmcode selbst ist auf den ersten Blick nicht ersichtlich. Dargestellt wird lediglich die Beziehung von Programmteilen zueinander – das macht die Projektstruktur sehr übersichtlich und der Entwickler bzw. Anwender kann sich schneller einen Überblick verschaffen. Möchte er ins Detail gehen, klickt er auf die Klasse und bekommt den Programmcode angezeigt.

**Austromatisierung:** Was sind das für praktische Details von »LASAL«, die den Unterschied zu anderen Systemen ausmachen?

**Dipl.-Ing. Hildebrandt:** Der Anwender profitiert von der Objektorientierung, ist dabei aber soweit wie möglich von der Komplexität einer



## »LASAL Motor Calculation« ergänzt Engineering Tool

Neu in Sigmateks All-in-one Engineering Tool »LASAL« bietet das Software-Modul »LASAL Motor Calculation« Unterstützung bei der Dimensionierung und energieeffizienten Auslegung von Antrieben, Motoren und Getrieben. Basierend auf benutzerdefinierbaren Geschwindigkeitsprofilen und mechanischen Daten lässt sich der optimale Antrieb für die jeweilige Applikation einfach bestimmen. Die nahtlose Integration des Antriebsstrangs in die Steuerungsplattform sorgt für eine schnelle und durchgängige Softwareentwicklung. In der mitgelieferten Bibliothek gibt es eine große Standard-Auswahl an Motion-Control-Funktionen. Für alle Sigmatek-Motoren sind bereits Parametersätze vorhanden, die einfach in die Applikation eingebunden und auch nachträglich verändert werden können. Da die Parameter in der Steuerung gespeichert werden, verfügt der Antrieb immer über die korrekten Daten. Somit ist ein Austausch des Servoverstärkers ohne zusätzlichen Aufwand und ohne Software-Tool möglich.

INFOLINK: [www.sigmatek-automation.com](http://www.sigmatek-automation.com)

objektorientierten Syntax befreit. Er definiert die Klassen mit den Schnittstellen und Datenelementen rein über die Oberfläche und »LASAL CLASS« erledigt das Einfügen des notwendigen Deklarationscodes. Dem Anwender bleibt nur mehr das Implementieren der einzelnen Methoden. Beim grafischen Ansatz, wie er in »LASAL« von jeher verfolgt wird, werden die Objekte in sogenannten Netzwerken dargestellt. Der große Vorteil dabei ist, dass die Maschine in der Software grafisch nachgebildet wird. Mit anderen Worten: Der Entwickler sieht auf den ersten Blick die wichtigsten Eigenschaften eines Maschinenteils sowie die Kommunikation mit anderen Objekten, sprich Maschinenteilen. Mit der Objektorientierung lassen sich – ausgehend vom Gesamtprojekt – die einzelnen Funktionen wie in einem Baukastensystem hierarchisch gruppieren und verbinden. Das ist ein großer Vorteil, da in der Praxis oftmals kaum einmal zwei Maschinen wirklich gleich sind. Der Maschinenbauer ist so in der

Erstellung unterstützt. Ein praktisches Feature ist die Möglichkeit des Programm-Updates via USB-File: Also-USB Stick anstecken, ausschalten, einschalten und schon ist das neue Programm geladen ohne, dass der Bediener ein Programmiergerät oder Programmierkenntnisse dafür benötigt.

**Austromatisierung:** *Ist das Feature mit der Datenübertragung via USB-Stick aus Security-Sicht nicht relativ riskant? Der Trojaner »Stuxnet« schlich sich ja bekanntlich über einen USB-Stick ein...*

**Dipl.-Ing. Hildebrandt:** Das ist ein guter Einwand – und ein Stichwort zugleich. Da Sigmatek sein eigenes Betriebssystem speziell für die hohen Sicherheitsansprüche unserer Maschinenbaukunden entwickelt hat und daher nicht auf »Windows« aufsetzt, gibt es die Probleme mit Viren und Hackerangriffen definitiv nicht.

**Austromatisierung:** *Wie machen sich die Unterschiede beim angesprochenen »Drumherum« bemerkbar?*

**Dipl.-Ing. Hildebrandt:** Wir verkaufen prinzipiell Lösungen – und nicht bloß Hardwarekomponenten oder Softwaretools. Das gilt für uns auch bei der Applikationsunterstützung – das ist für uns selbstverständlich, dass wir hier unserem Kunden zur Seite stehen. Bei Sigmatek bekommt man also grundsätzlich eine Gesamtlösung mit allem Drumherum angeboten und eine funktionierende Lösung geliefert. Und wir gehen sehr flexibel auf spezifische Kundenwünsche ein.

**Austromatisierung:** *Danke für das Gespräch.*

*Gesprächspartner von Dipl.-Ing. Bernd Hildebrandt war Austromatisierung-ChR. Ing. Thomas Reznicek.*

### „Die Unterschiede stecken in einer Vielzahl praktischer Details.“

Lage rasch und flexibel auf individuelle Anforderungen und Ausstattungsoptionen zu reagieren. Einmal erstellte Objekte werden in Bibliotheken abgelegt. Sie lassen sich in unterschiedlichen Projekten oder Systemteilen einfach einsetzen und zu hochkomplexen Programmstrukturen zusammenfügen, ohne in den Programmcode der untersten Ebene eingreifen zu müssen. Ein wichtiger Aspekt ist hier natürlich die Erweiterung bzw. Abänderung durch Vererbung. All dies reduziert die Engineeringzeiten und -kosten erheblich, zudem wird das Fehlrisiko reduziert. Zudem wird der Entwickler mit komfortablen Werkzeugen wie Online-Debugger, Echtzeit-Datenaufzeichnung und automatische Projektbearbeitung über Scripts und Bootstrik-