

SYNCHRON 2016 – Daten & Informationen

Allgemeine Angaben:

23. Internationale Informatiktagung zu Synchronen Programmiersprachen und ihren Anwendungen

5.12.2016-9.12.2016 in der AULA (ehem. Dominikanerkirche),
Bamberg, Dominikanerstraße 2a.

Veranstalter:

Otto-Friedrich-Universität Bamberg,
Fakultät für Wirtschaftsinformatik und Angewandte Informatik,
Professur für Grundlagen der Informatik,
Prof. Michael Mendler, PhD
An der Weberei 5
96047 Bamberg

<https://www.uni-bamberg.de/gdi/synchron-2016/>

Fachkontext der Tagung:

Damit ein Computer arbeiten kann, muss er programmiert werden. Ohne Software bleibt die Hardware kalt. In einer Programmiersprache gibt der Informatiker dem Computer Rechenvorschriften und Steuerungsanweisungen. Um Verständigungsprobleme durch Missverständnisse zu verhindern besitzen Programmiersprachen eine starre Grammatik und einen stark eingeschränkten Wortschatz. Denn die Wirkungsweise eines Programms muss eindeutig und widerspruchsfrei festgelegt sein. Deshalb sind Programmiersprachen auch nicht annähernd so anpassungsfähig wie unsere natürlichen Sprachen. Kein Wunder also, dass Informatiker immer neue Programmiersprachen erfinden. Sprachen für Fertigungsroboter, Sprachen für Mobiltelefone, Sprachen für Geschäftsprozesse, Sprachen für Internetdienste. Der Zoo der Programmiersprachen ist ebenso reichhaltig wie die Welt der technischen Computeranwendungen.

Die Tagung befasst sich mit den synchronen Programmiersprachen, die den speziellen Erfordernissen sicherheitskritischer technischer Steuerungsanwendungen (Cyber-physisches System) angepasst sind. Mit der zunehmenden Durchdringung und engen elektronischen Vernetzung aller Lebensbereiche durch Computerintelligenz (Internet of Things, Industrie 4.0, Autonome Fahrzeuge, Pflegeroboter, usw.) werden diese immer wichtiger. Bereits heute wäre die Komplexität hoch automatisierter Produktionsanlagen, moderner Automobile oder Flugzeuge ohne die Unterstützung durch komplexe Netzwerke von Steuerungsprozessoren und ihrer Software nicht denkbar.

Der Trend zur Miniaturisierung und Vernetzung der Informationstechnik bietet neue Möglichkeiten wirft aber auch schwierige technische Fragen auf. Wo Computer in einem engen Verbund gemeinsam eine Steuerungsaufgabe lösen, spielt die sichere und reibungslose Koordination von Teilabläufen eine große Rolle. In der Automatikschaltung eines Fahrzeuges muss zuerst das Gas zurückgenommen, dann die Kupplung geöffnet und schließlich erst der Gang umgelegt werden. In der vollautomatischen U-Bahn müssen erst alle Türen vollständig

geschlossen sein bevor der Zug vom Bahnsteig abfährt. So versteckt sich Synchronisationsfehler im Programmablauf einschleichen, so unerwünscht sind ihre Auswirkungen. Diese reichen vom kurzfristigen Ausfall eines Smartphones, dem Ärger eines kaputten Getriebes oder verklemmten Kabriooverdecks über astronomisch hohe Kosten eines verlorenen Satelliten bis hin zum Verlust von Menschenleben.

Die korrekte Synchronisation technischer Prozesse verlangt vom Programmierer besondere Verantwortung und stellt somit auch spezielle Anforderungen an die Programmiersprache. Denn, was Synchronisation betrifft gibt es einen entscheidenden Unterschied zwischen Menschen und Computern. Menschen sind einfallsreich und flexibel. Wo sich Menschen in einer gemeinsamen Sprache spontan aus einer Synchronisationsfalle befreien, bleiben naiv programmierte Computer durch das starre Korsett ihrer Programmiersprache in einer Blockade gefangen. Wer hat nicht schon einmal erlebt, wie schnell er sich mit einem entgegenkommenden Kollegen an einer engen Türe über den Vortritt einigt. Roboter würden das „Bitte, nach Ihnen!“ stupide und unermüdlich wiederholen, bis ihnen – wer sonst? - ein Mensch genervt den Strom abschaltet. Bei technischen Anwendungen, in deren Ablauf Menschen nicht ohne weiteres eingreifen können, kommt es deshalb darauf an, durch geeignete Mechanismen der Programmiersprache mögliche Synchronisationsfehler von vorne herein auszuschließen.

Im Bereich des Entwurfes und der Programmierung eingebetteter Systeme in der Automobil- und Flugzeugindustrie werden deshalb gerne synchrone Programmiersprachen eingesetzt. Diese Sprachen sind darauf ausgerichtet, nebenläufig arbeitende Prozessoren durch einen globalen Steuertakt zu synchronisieren, etwa wie der Steuermann in einem Achterboot die Schläge der Ruderer koordiniert. Synchrone Sprachen nutzen ähnliche Tricks, um in verteilt programmierten Systemen blockadefreie deterministische Abläufe sicherstellen. Sie eignen sich besonders für sicherheitskritische Steuerungssysteme, bei denen es auf verifizierbare Korrektheit ankommt. Aufgrund ihrer Konstruktion ist die Funktionstüchtigkeit im Rahmen eines „Software-TÜVs“ mittels mathematischer Verfahren bedeutend leichter nachweisbar als für andere Programmiersprachen.

Hintergrundinformationen zur Tagungsreihe:

Synchrone Programmiersprachen werden in der Industrie und an Universitäten kontinuierlich weiterentwickelt und neuen Anforderungen angepasst. Jedes Jahr treffen sich international führende Forscher und Ingenieure zur Informatikarbeitsstagung SYNCHRON. Dort werden neueste Forschungsergebnisse zum synchronen Entwurf, der Spezifikation, der Verifikation und dem Test von informationstechnischen Systemen mittels synchroner Programmiersprachen diskutiert.

SYNCHRON 2016 ist die 23. Tagung dieser Reihe, die nach 2007 nun zum zweiten Mal in Bamberg stattfindet. Es werden ca. 60 Teilnehmer aus China, Dänemark, Deutschland, Frankreich, Großbritannien, Indien, Neuseeland und den USA erwartet. Tagungssprache ist Englisch.

Die Tagung wird finanziell unterstützt von der Deutschen Forschungsgemeinschaft, der Technologiegel Allianz Oberfranken und ANSYS (USA).

Prof. Michael Mendler, PhD.