

AixOCAT: Open-Source-Bibliothek für Automationssysteme

Markus Hans Schraven, Alexander Kümpel, Marc Axel Baranski, Matthias Mersch, Thomas Storek, Gerrit Bode, Markus Nürnberg, Christian Vering, Dirk Müller

In der heutigen Zeit hat Open-Source-Software einen hohen Stellenwert erreicht. Auch in der Automatisierungstechnik könnten sich die Vorteile von frei verfügbarer und von einer lebendigen Community entwickelten Software nutzen lassen. In diesem Beitrag wird eine Open-Source-Bibliothek vorgestellt, die zunächst für den Anwendungsbereich Gebäudeautomation genutzt werden kann, aber auch die Möglichkeiten für andere Automatisierungsbereiche aufzeigen soll.

Der Gebäudesektor steht heute vor einer zunehmenden Komplexität bei der Planung und dem Betrieb von Gebäudeenergiesystemen. Eine geeignete Implementierung der Gebäudeautomation (GA) für solche Systeme ist daher zeitaufwendig und kostspielig. Um den Implementierungsaufwand zu reduzieren, ist eine strukturierte Steuerungsentwicklung mit wiederverwendbarem Programmcode vorteilhaft. Die kommerziell verfügbaren Steuerungsbibliotheken verursachen jedoch häufig erhebliche Kosten und bieten nur eine begrenzte Interoperabilität mit weiterer Software. Dadurch sind insbesondere kleinere Unternehmen in ihren Möglichkeiten zur Implementierung von GA-Systemen eingeschränkt. Dieser Artikel demonstriert die allgemeine Nutzung der AixOCAT-Bibliothek und den Workflow zur Implementierung von Regelalgorithmen in GA-Systemen am Beispiel eines Regelkreises in einem neu errichteten Gebäude. Die Bibliothek kann somit einen wichtigen Beitrag zum Aufbau intelligenter und energieeffizienter Gebäude leisten.

Hohe Anwendungsvielfalt mit unterschiedlichen Anforderungen

Gebäudeautomationssysteme gelten als notwendiger Bestandteil energieeffizienter und komfortabler Gebäude. In diesem Bereich existiert eine Vielfalt von Anwendungsfällen mit unterschiedlichen Anforderungen [1]. Der übliche Engineering-Prozess von GA-Systemen ist jedoch oft ineffizient und führt zu Energiesystemen, die nicht ihr volles Effizienzpotenzial nutzen [1][2]. Ursächlich für diese Diskrepanz ist vor allem das Fehlen eines allgemeinen

Konzepts für die Erstellung von Steuerungsalgorithmen, was zudem zu einem sehr heterogenen Gebäudebestand führt [1][3]. Aufgrund der großen Unterschiede zwischen einzelnen Gebäuden ist es schwierig, intelligente Algorithmen wie die automatische Fehlererkennung und -diagnose oder modellprädiktive Regelung anzuwenden [3].

Eine frei verfügbare Bibliothek mit objektorientierten, standardisierten Modulen und Best-Practice-Beispielen könnte Ingenieure bei der Implementierung unterstützen, wodurch die Qualität der GA-Systeme deutlich gesteigert würde. In diesem Beitrag wird eine Open-Source-Bibliothek für Automatisierungssoftware nach IEC 61131 vorgestellt, um eine effiziente Entwicklung von Automatisierungscode zu ermöglichen, der sowohl zuverlässig ist als auch die Anwendung fortschrittlicher Regelalgorithmen und Betriebsanalysen erlaubt.

Die AixOCAT-Bibliothek

Zur Erreichung der bereits genannten Effekte wurden die nachfolgenden Ziele definiert, nach welchen die AixOCAT-Bibliothek [4] ausgerichtet ist. Sie bietet sowohl Code-Vorlagen als auch Best-Practice-Beispiele, die diesen Prinzipien folgen. Die Ziele lauten:

1. Eine hohe Wiederverwendbarkeit des Codes und Reduzierung des Engineering-Aufwands.
2. Erhöhung der Zuverlässigkeit der technischen Gebäudeausrüstung (TGA) und Prüfstände.
3. Unterstützung der Erweiterbarkeit von Automatisierungssystemen.
4. Interoperabilität, insbesondere mit cloudbasierten Anwendungen.

Wiederverwendbarkeit und Zuverlässigkeit

Die AixOCAT-Bibliothek enthält Code-Templates, die typische

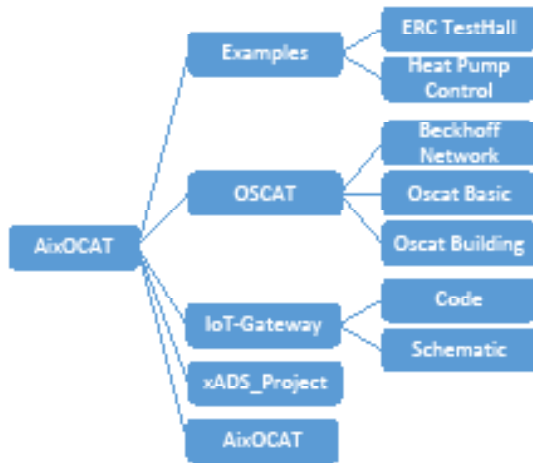


Abbildung 1: Aktuelle Struktur der AixOCAT-Bibliothek

Funktionen zur Automatisierung von TGA und Prüfständen bereitstellen. Als Basis der AixOCAT dient die nicht mehr weiterentwickelte Open Source Community for Automation Technology (OSCAT) [5]. Die OSCAT-Bibliothek enthält viele nützliche Module für die GA, die in einer Vielzahl von Projekten wiederverwendet werden können. Die AixOCAT-Bibliothek hat diese größtenteils integriert und soll ebenfalls open source sein, sodass verschiedene Mitwirkende Codes schreiben und überprüfen, Softwarefehler melden und weitere Entwicklungen anfragen können. Dadurch kann die zur Bibliothek beitragende Automatisierungsgemeinschaft ihr Know-how bündeln und so zuverlässige Automatisierungsprojekte gemäß unserem zweiten Ziel sicherstellen.

Erweiterbarkeit und Interoperabilität

Wir verfolgen einen modularen Designansatz, um Erweiterbarkeit und Interoperabilität zu gewährleisten: Die Bibliothek enthält hauptsächlich Programm-Organisationseinheiten (POUs), die in Strukturtext (ST) nach der IEC 61131-3 implementiert sind. Die Bausteine sind meist Funktionsblöcke (FB), die mehrere Eingänge in mehrere Ausgänge umwandeln. Für einen objektorientierten Programmierstil wurde die FB-Funktionalität gemäß IEC 61131 um Eigenschaften und Methoden erweitert. Die Softwareanwendung TwinCAT [6] ist ein von uns getestetes Beispiel, das es ermöglicht, diese erweiterten FBs zur Programmierung von Echtzeitsteuerungen zu nutzen.

Mit Hilfe von FBs sind die Programme nach physikalischen Subsystemen im Energiesystem gegliedert wie beispielsweise „raumluft-technische Anlage“ (RLT) oder „Betonkerntemperatur“ (BKT). Für die weitere Strukturierung schlagen wir einen betriebsarten-basierten Ansatz vor. Dabei ist eine Betriebsart eine Sammlung von vordefinierten Stellbefehlen oder Sollwerten für Feldgeräte. Die Übergänge von einer Betriebsart in eine andere basieren auf vordefinierten Bedingungen, beispielsweise Fehlerzuständen oder Schwellwerten [7].

Um eine erweiterte Interoperabilität zu unterstützen, kann eine Betriebsart für externe Steueranwendungen vorgesehen werden. So kann eine Verbindung zu Steuerungsalgorithmen ermöglicht werden, die außerhalb des eigentlichen Automatisierungssystems, z. B. auf einer virtuellen Maschine, gehostet werden können, um Steuerbefehle an die Steuerung zu senden. In diesem Fall sind die Stellgliedbefehle und Sollwerte nicht vordefiniert, sondern werden in der externen Anwendung generiert. Bei einem Kommunikationsausfall sorgt ein

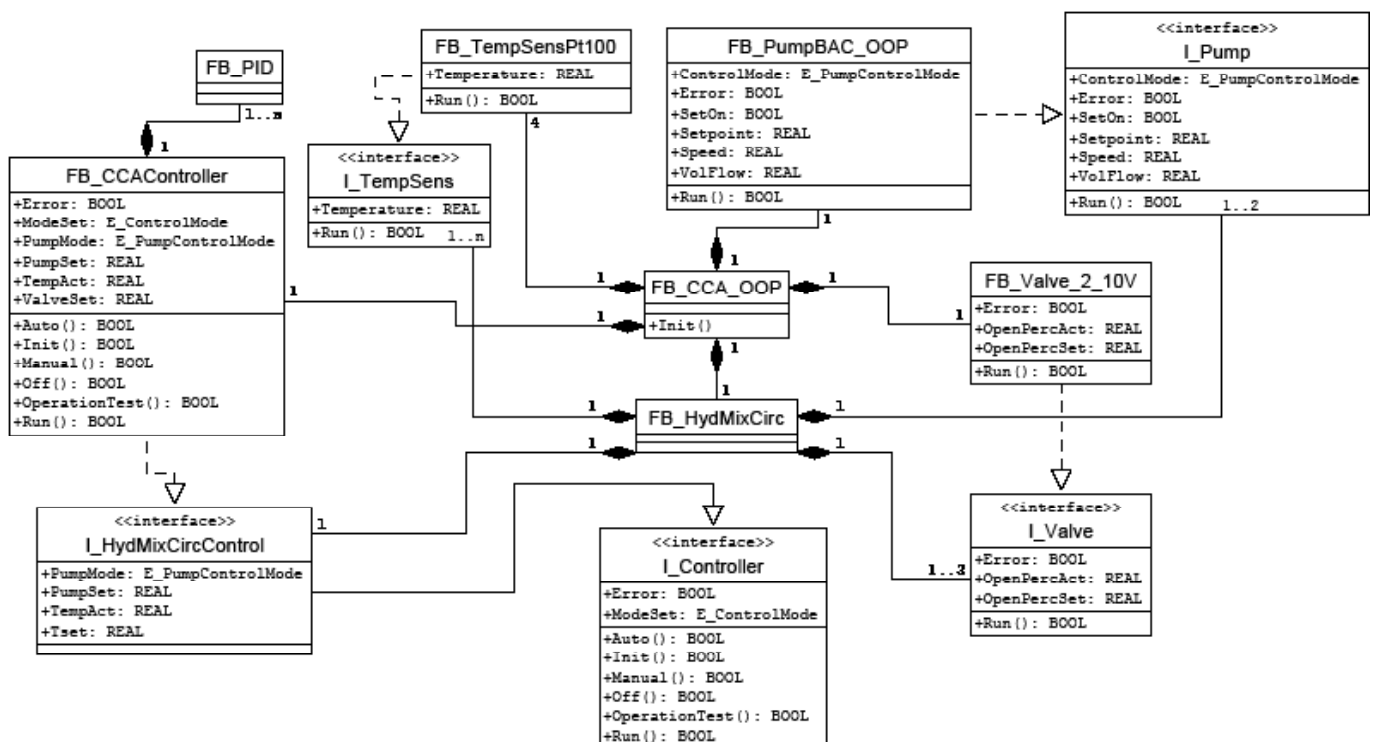


Abbildung 2:

UML-Diagramm des BKT-Automationssystems. Standard-Eigenschaften sowie die get- und set-Methoden sind der Übersicht halber nicht dargestellt.

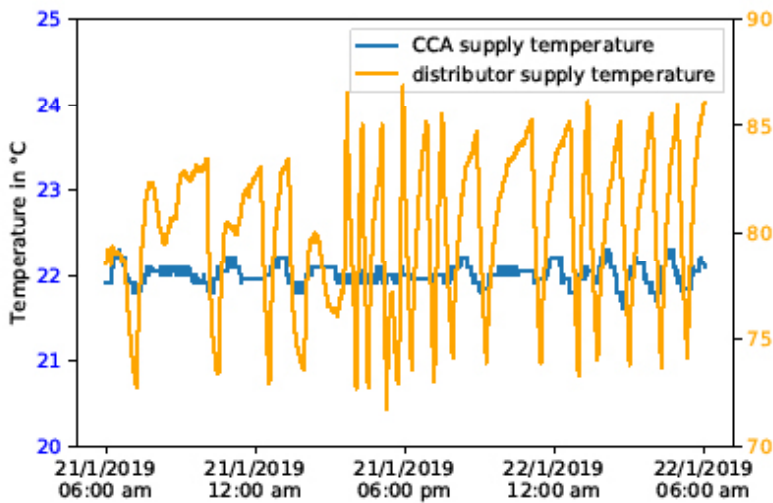


Abbildung 3: Plot der BKT-Vorlauftemperatur und der Verteilertemperatur in der betrachteten Versuchshalle für einen Zeitraum von 24 Stunden. Die ausgeprägte Schwingung der Verteilertemperatur wird fast vollständig durch die PID-Regelung kompensiert.

Übergang in eine lokale Betriebsart immer für einen sicheren Systembetrieb.

Eine wichtige Forschungsanwendung ist die Hardware-in-the-Loop-Methode, bei der eine reale Steuerung mit der virtuellen TGA gekoppelt wird, die durch ein Simulationsmodell dargestellt wird. Zu diesem Zweck stellt AixOCAT Schnittstellen für Modelica zur Verfügung, eine objektorientierte Multiphysik-Modellierungssprache. Die Anwendung verwendet die Automation-Device-Specification-Schnittstelle (ADS-Schnittstelle), um Daten mit der SPS auszutauschen.

Kollaborative Entwicklung

Grundsätzlich wird eine flexible, offene Entwicklung der Bibliothek angestrebt, die über GitHub [8] organisiert ist. Die Beteiligten schlagen einen Arbeitsablauf für einen uneingeschränkten, aber effizienten Beitrag zur Bibliothek vor. Dieser Arbeitsablauf definiert grundlegende Anforderungen wie Programmierstil, Dokumentation und Problemverfolgung. Darüber hinaus werden Neuentwicklungen und Code-Erweiterungen in separaten Zweigen durchgeführt, um sicherzustellen, dass der Master-Zweig stabil und für die Community nutzbar bleibt. Eine so genannte „Pull-Request“ erfordert, dass ein anderer den neu hinzugefügten Code überprüft und den überarbeiteten Code in den Master-Zweig einfügt.

Use Case: ERC-Versuchshalle

Um die Nutzung der Bibliothek und ihre Vorteile zu demonstrieren, stellen wir die Automation unserer Versuchshalle als repräsentatives Beispiel für eine Labor- und Maschinenhalle vor. Die Anlage besteht aus zwei Hallenteilen mit einer Fläche von 630 m² bzw. 360 m² für Prüfstände und einem zusätzlichen Bereich mit fünf Büroräumen und einer Gesamtfläche von 90 m². Die Testhalle ist mit einer RLT-Anlage, einer BKT und Deckenstrahlplatten (DSP) sowie Deckeninduktionsgeräten (DIDs) in den Büroräumen ausgestattet. Darü-

ber hinaus enthält das Energiesystem externe Nacherhitzer (NE) und -kühler (NK). Am Beispiel der BKT demonstrieren wir die Funktionalität der AixOCAT-Bibliothek.

Regelstrategie

Das benötigte Warmwasser für RLT, BKT und die dezentralen NE wird über einen Wärmeverteiler bereitgestellt. Wie in den vorherigen Abschnitten beschrieben, unterstützt die Bibliothek insbesondere die SPS-Programmierung durch objektorientierte Programme für GA-Systeme. So umfasst die BKT beispielsweise eine konventionelle Beimischschaltung bestehend aus einer Pumpe, einem Mischventil sowie vier Temperaturfühlern und einer Regelungseinheit. Abbildung 2 zeigt die entwickelte objektorientierte Darstellung der BKT als Klassendiagramm in der Unified Modeling Language (UML).

Die Komponenten sind als Schnittstellen konzipiert, die entsprechend den Kommunikationsschnittstellen der physikalischen Komponenten implementiert und abgebildet werden. Durch die Verwendung dieser Struktur wird grundsätzlich jede Beimischschaltung auf die gleiche Weise beschrieben.

Details und Ergebnisse

Wie zuvor beschrieben, sind die Module des Hydraulikkreises Schnittstellen, die auf verschiedene Weise implementiert werden können. Im vorliegenden Beispiel sind sie wie folgt umgesetzt:

1. Die Temperatursensoren als PT100 (inkl. Umwandlung von Integer in Temperatur).
2. Das Ventil als (0)2-10 V-Gerät für Ist- und Sollposition (inkl. Umwandlung zwischen Integer und Öffnung in Prozent).
3. Die Pumpe als BACnet-Gerät.
4. Die Regelung als modusbasierte Steuerung, die die Mischungsvorlauftemperatur über eine PID-Steuerung regelt.

Im Automatikbetrieb der Regeleinheit wird die Solltemperatur in einer Heizkurve ermittelt. Dieser Wert wird mit der Ist-Temperatur verglichen und ein PID-Regler erzeugt das Stellsignal für das Mischventil, während die Pumpe eine konstante Förderhöhe bereitstellt.

Der gesamte Code für die beschriebenen Module der Beimischschaltung befindet sich in der Bibliothek. Abbildung 3 zeigt eine Darstellung der Vorlauftemperatur für die BKT sowie die Temperatur am Wärmeverteiler. Die Vorlauftemperatur wurde auf 22 °C eingestellt. Die Darstellung zeigt, dass die stärker oszillierende Verteilertemperatur durch den Beimischkreis gut kompensiert werden kann.

Neben der BKT werden auch die DIDs, der Vorerhitzer, der Nacherhitzer sowie der Kühler der RLT durch Beimischschaltungen realisiert, so dass der oben beschriebene Code für die jeweiligen Hydraulikkreisläufe nur noch mit deren physikalischen Sensoren, Pumpen und Ventilen verbunden werden muss.

Fazit

Es wurde eine Open-Source-Bibliothek eingeführt, die der Unterstützung der Projektierung von Gebäudeautomationsystemen dient. Insbesondere bietet die AixOCAT-Bibliothek Code-Templates und Best-Practices zur Implementierung von Modulen in Automationssoftware. In diesem Beitrag haben wir die Bibliotheksstruktur vorgestellt und die Anwendung anhand eines realen Systems demonstriert.

Da viele Mängel im Gebäudebestand auf das Fehlen eines allgemeinen, strukturierten Konzepts für die Erstellung von Gebäudeautomationssoftware zurückzuführen sind, folgt die Bibliothek den daran abgeleiteten Prinzipien:

- » Wiederverwendbarkeit und Zuverlässigkeit,
- » Erweiterbarkeit und Interoperabilität sowie
- » Kollaborative Entwicklung.

Die Bibliothek basiert auf diesen Ideen und trägt durch die Open-Source-Entwicklung somit auch zur Standardisierung des Engineering-Prozesses in der Gebäudeautomation bei. Die Bibliothek wird ständig verbessert und ist auf GitHub [3] verfügbar.

Danksagung

Wir danken dem Europäischen Fonds für regionale Entwicklung (EFRE, Förderkennzeichen EFRE-0800537) und dem Bundesministerium für Wirtschaft, Mittelstand und Energie (BMWi, Förderkennzeichen 03ET1485A) für die finanzielle Unterstützung.

Referenzen

- [1] Domingues, P., Carreira, P., Vieira, R., Kastner, W. (2016): Building automation systems: Concepts and technology review. In Computer Standards & Interfaces 45 (pp. 1–12). DOI: 10.1016/j.csi.2015.11.005.
- [2] Royapoor, M., Antony, A., & Roskilly, T. (2018). A review of building climate and plant controls, and a survey of industry perspectives. In Energy and Buildings, 158, (pp. 453–465). DOI: 10.1016/j.enbuild.2017.10.022.
- [3] Gunay, H. B., Shen, W., & Newsham, G. (2019). Data analytics to improve building performance: A critical review. Automation in Construction, 97, (pp. 96–109). DOI: 10.1016/j.autcon.2018.10.020.
- [4] GitHub Inc. (2019). RWTH-EBC / AixOCAT. Abgerufen von: <https://github.com/RWTH-EBC/AixOCAT>.
- [5] Open Source Community for Automation Technology (OSCAT). (2015). Willkommen bei OScat. Abgerufen von: <http://www.oscat.de/>.
- [6] Beckhoff Automation GmbH & Co. KG. (2019). TwinCAT 3 eXtended Automation (XA). Abgerufen von: <https://www.beckhoff.de/twincat3/>.
- [7] Schild, T. P., Baranski, M. A., Müller, D. (2018): Modus-basierte Steuerungsentwicklung für modulare Energieversorgungssysteme. In: Tagungsband Automation 2018, Baden-Baden, (pp. 1009–1021), VDI-

Verlag, VDI-Berichte 2330.

- [8] GitHub Inc. (2019). Built for developers. Abgerufen von: <https://github.com>.

M.Sc. Markus Hans Schraven

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen
Tel. +49 241 80 49780
mschraven@eonerc.rwth-aachen.de

M.Sc. Alexander Kümpel

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen

M.Sc. Marc Axel Baranski

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen

M.Sc. Matthias Mersch

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen

M.Sc. Thomas Storek

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen

M.Sc. Gerrit Bode

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen

M.Sc. Markus Nürnberg

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen

M.Sc. Christian Vering

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen

Univ.-Prof. Dr.-Ing. Dirk Müller

RWTH Aachen, E.ON Energieforschungszentrum, Lehrstuhl für Gebäude- und Raumklimatechnik
52074 Aachen