

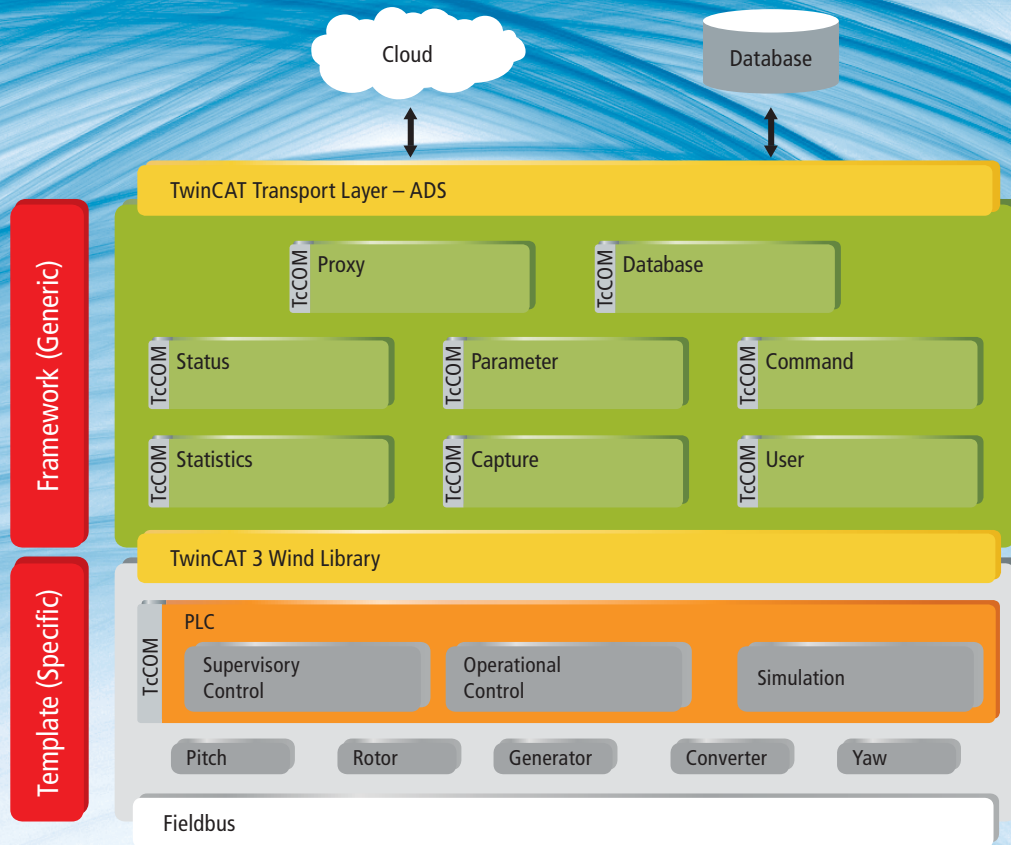


TwinCAT 3 Wind Framework für Windenergieanlagen

1 Framework mit 10 Jahren Know-how aus 40.000 Anlagen

Mit dem neuen TwinCAT 3 Wind Framework sind Hersteller von Windenergieanlagen in der Lage, ihre Anlagen schnell und komfortabel selbst zu programmieren. Alle entscheidenden Funktionen sind in einem Softwarepaket integriert: von der Ereignisverwaltung über die Datenbankanbindung bis hin zu sämtlichen Grundfunktionen, wie Zustandsmaschine oder Hydraulik. Ein vorgefertigtes Applikations-Template reduziert den Aufwand deutlich, denn die Entwickler können sich direkt auf die eigentlichen Anlagenfunktionen konzentrieren. Das Ergebnis: effizientes Engineering, schnellere Time-to-Market und Nutzung der Industrie-4.0-Eigenschaften für die Windenergie.





Applikations-Template und gekapselte Module für modulare Softwarearchitektur mit hoher Funktionalität

Bereits seit 16 Jahren bietet Beckhoff Lösungen für Windenergieanlagen an. So haben sich die TwinCAT-2-Wind-Libraries mittlerweile durch viele erfolgreiche Einsätze im Feld bewährt und bieten Anwendern eine stabile Basis für die Entwicklung der Betriebsführung von Windenergieanlagen.

Durch die immer schneller voranschreitende Entwicklung von immer größeren Windenergieanlagen entstehen neue Anforderungen: Es werden intelligentere Systeme mit zusätzlichen Sensoren und Aktoren eingesetzt, wodurch sich die Komplexität der Anlagen weiter steigert. Dadurch wird die Analyse von Störungen wesentlich umfangreicher und somit ein stabiler Betrieb zunehmend erschwert. Um diese Anforderungen zu erfüllen, wurden die bestehenden Konzepte und Technologien zusammengefasst und das TwinCAT 3 Wind Framework neu entwickelt.

Umfassende Funktionalitäten sind dabei in gekapselten TwinCAT-Modulen umgesetzt, welche sich in die TwinCAT-3-Architektur integrieren. Eine effiziente Softwareentwicklung wird durch eine modulare Architektur im Applikations-Template und mittels erprobter und direkt verwendbarer TwinCAT-Module und -Funktionen gewährleistet. Durch die flexible Konfiguration ist eine Adaption an die spezifische Applikation des Anwenders einfach möglich. Die Anlagen-diagnose wird mittels tiefgreifender Datenhaltung in einer Datenbank sichergestellt. Hieraus ergeben sich eine zukunftssichere Entwicklung, eine effiziente Inbetriebnahme sowie ein optimaler Betrieb der Automatisierungssoftware von Windenergieanlagen.

TwinCAT 3 bietet die Möglichkeit, IEC 61131-3-, C++- und MATLAB®/ Simulink®-Module zu implementieren, diese auf verschiedene CPU-Cores zu laden, in unterschiedlicher Echtzeit ablaufen und trotzdem sicher miteinander interagieren zu lassen. Die Grundlage hierzu ist die TwinCAT-Modulsprache, welche die Eigenschaften der TwinCAT-Module, u. a. bezüglich der Prozessparameter oder der Methoden, beschreibt.

Betriebsführung per Template und Bibliothek programmieren

Die Programmierung einer Betriebsführung unter Verwendung des TwinCAT 3 Wind Frameworks wird durch eine Bibliothek und ein Applikations-Template erleichtert. Die Bibliothek stellt alle Funktionen des Wind Frameworks als SPS-Funktionsblöcke zur Verfügung. In dem Applikations-Template wird eine modulare Architektur für die Betriebsführungssoftware von Windenergieanlagen als SPS-Projekt bereitgestellt. Darin sind die Funktionalitäten der TwinCAT-Module und -Functions bereits vollständig umgesetzt.

Die Architektur des Applikations-Templates wird vereinfacht in Abbildung 1 gezeigt. Jedes Subsystem (wie Pitch, Umrichter, etc.) der Windenergieanlage ist durch ein eigenständiges Objekt dargestellt. Somit können die Subsysteme unabhängig voneinander entwickelt, verwendet und getestet werden. Wie es in der mechanischen Modularisierung von Anlagen bereits üblich ist, sind die Subsysteme nun auch in der Software austauschbar. Die Modularisierung erlaubt eine parallele Entwicklung, und jeder Programmierer kann sich auf die eigentlichen Funktionen und seine spezialisierten Komponenten der Anlage konzentrieren. Damit wird die Qualität, Flexibilität und Wiederverwendbarkeit der Software erhöht und gleichzeitig reduziert sich die Entwicklungszeit und -kosten.

Die verschiedenen Betriebsmodi für das Starten, Stoppen und die übergeordnete Zustandsmaschine der Anlage sind in dem Applikations-Template als Supervisory Control zusammengefasst und bereits durch vereinfachte Implementierungen als SPS-Funktionsblöcke umgesetzt. Daraus entstehen übergeordnete Sollwerte für den Betrieb der Anlage, welche in die Regelung eingehen.

Die allgemeine Regelung der Windenergieanlage, wie die Pitch- und Torque-Regelung, ist in der Software als Operational Control vorbereitet. Für die Regelung ist die Integration weiterer Module vorgesehen, um zum Beispiel die Algorithmen aus der Lastenrechnung zu übernehmen.

So besteht die Möglichkeit, die automatische Generierung eines TwinCAT-Moduls aus MATLAB®/Simulink® heraus zu nutzen oder mittels C/C++ die Algorithmen zur Regelung einzubinden.

Damit wird der gleiche Regler, der für die Lastenrechnung verwendet wurde, auch in der Steuerung genutzt. Der Regler muss nicht erst in eine zweite Programmiersprache übertragen werden, und die fehleranfällige zweite Umsetzung der Algorithmen kann entfallen.

TE1400 – TwinCAT 3 Target for MATLAB®/Simulink®

Das TwinCAT 3 Target for MATLAB®/Simulink® ermöglicht es, echtzeitfähige TwinCAT-Module aus einem MATLAB®/Simulink®-Modell heraus zu erzeugen. Diese können dann in der TwinCAT-3-Runtime ausgeführt, mehrfach instanziiert, parametrisiert und debuggt werden, ohne dass noch ein MATLAB®/Simulink® notwendig ist. Durch die Übernahme des gesamten Blockschaltbildes aus Simulink® in das TwinCAT-Modul wird die Analyse und Optimierung des Reglers möglich. Diese Optimierung kann im Feld und direkt an der Anlage geschehen, da nur noch das TwinCAT-3-Engineering notwendig ist, um den Regler zu parametrieren.

Die Vorgaben aus den Betriebsmodi und der Regelung werden in das Subsystem Control übernommen, wo die Steuerung der individuellen Subsysteme

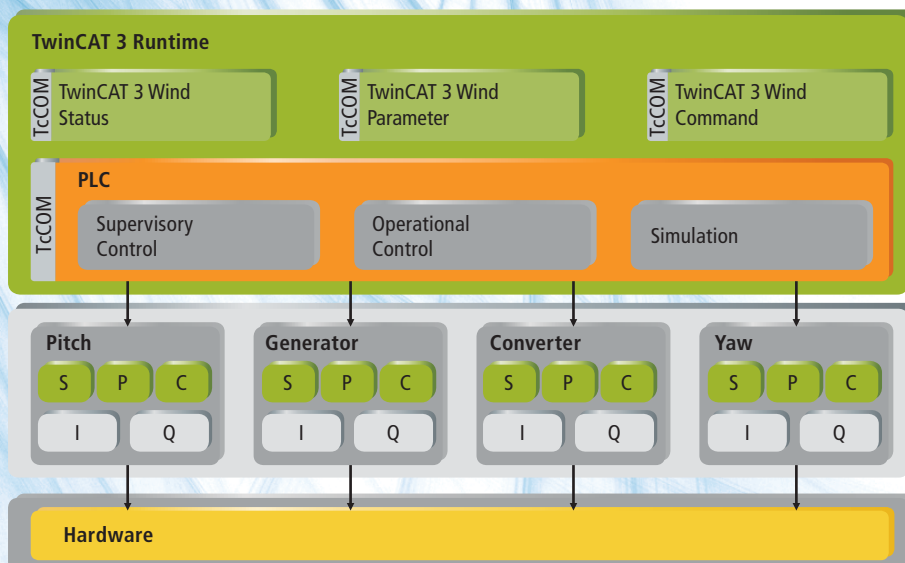


Abb. 1: Die Betriebsführung wird als eigenständiges TwinCAT-Modul und die Subsysteme werden als unabhängige Objekte implementiert.

durchgeführt wird. Jedes Subsystem wird als Modul mit fünf Methoden und vier Datenstrukturen als SPS-Funktionsblock umgesetzt. Der einheitliche Aufbau der Subsysteme wird in Abbildung 2 dargestellt. Die Input/Output-Datenstrukturen werden mit den Hardware-Ein- und Ausgängen verknüpft und enthalten die numerischen Werte von Sensoren, Aktoren sowie von Feldbussystemen. In der Methode InputUpdate werden die numerischen Daten aus der Input-Datenstruktur vorverarbeitet und als physikalische Werte in die Inbox-Datenstruktur übertragen. Für die Überwachung und Steuerung des Systems werden die Methoden ActualUpdate, MonitorUpdate und ControlUpdate aufgerufen, welche direkt auf die physikalischen Werte aus der Inbox zurückgreifen. Die neuen Werte zur Steuerung der Aggregate werden als physikalische Werte in die Outbox-Datenstruktur geschrieben. In der Methode OutputUpdate wird die Outbox-Datenstruktur nachbearbeitet und als numerischer Wert bereitgestellt. Eine Anbindung an das übergeordnete Supervisory Control geschieht über die Actual- und Demand-Datenstrukturen. Die Inbox- und Outbox-Datenstrukturen ermöglichen darüber hinaus eine einfache Simulation der Systeme anhand der physikalischen Werte. Jedes Subsystem bringt dadurch seine eigene Simulation mit in das Gesamtsystem ein.

Darüber hinaus ist eine adaptive Simulation von einer 5-MW-Offshore-Windenergieanlage in das Applikations-Template integriert, welche entsprechend der Referenzanlage vom National Renewable Energy Laboratory (NREL) vorkonfiguriert ist. Dies ermöglicht ein Testen der gesamten Betriebsführung in der Entwicklungsumgebung. Dabei ist das Modell der Anlage frei konfigurierbar und auf die jeweilige Anlage anpassbar. Die verwendete Anlagensimulation wird als TwinCAT-Modul bereitgestellt, ist aber genauso wie die Regelung dafür vorbereitet, um bei Bedarf durch ein spezifisches Modell aus MATLAB®/Simulink® oder C/C++ ersetzt zu werden.

Mittels der integrierten Simulationen in das Applikations-Template können die Vorgänge des Gesamtsystems, die Betriebsmodi und auch einzelne Subsysteme abgebildet, nachvollzogen und geprüft werden. Jedes Subsystem kann separat und unabhängig betrieben werden, indem zwischen Simulation und realer Hardware umgeschaltet wird. Dies ermöglicht es zum Beispiel, Teilkomponenten der Gondel in der Werkshalle in Betrieb zu nehmen und zu prüfen. Darüber hinaus

wird der Aufbau von Testständen möglich, um Software-in-the-Loop- oder Hardware-in-the-Loop-Simulationen und sogar Schulungen direkt mit der originalen Applikationssoftware durchzuführen. Diese Echtzeitsimulationen ermöglichen ein Rapid Control Prototyping und virtuelle Inbetriebnahmen mit einer einzigen Version der Software und durch reine Parametrierung.

Die Betriebsführung und deren Subsysteme werden durch die Verwendung der verfügbaren TwinCAT-Module aus dem TwinCAT 3 Wind Framework vervollständigt. Über SPS-Funktionsblöcke aus der SPS-Bibliothek werden Objekte angelegt und konfiguriert. Diese Objekte integrieren sich automatisch in die übergeordneten TwinCAT-Module aus dem Wind Framework, welche die Dienste und Funktionalitäten bereitstellen. Dadurch definiert jedes Subsystem einen individuellen Satz an Objekten, welche die Informationen und Einstellungen in die Betriebsführung einbringen.

Mit der durchgängigen Nutzung der TwinCAT-Module und durch die einheitliche Architektur der Subsysteme entsteht ein Standard in der Applikation. Diese Standardisierung ermöglicht es dem Programmierer, sich schnell in die Applikation und den Source-Code einzufinden, sogar wenn diese von einem anderen Programmierer umgesetzt wurde.

Generische Module für übergeordnete Dienste

Die generischen TwinCAT-Module stellen die übergeordneten Dienste (Services) bereit. Jedes Modul ist direkt verwendbar und wird nur noch in TwinCAT 3 als TCCOM-Modul eingebunden. Die Module können einzeln und unabhängig voneinander verwendet oder miteinander kombiniert werden, um eine Interaktion und den Datenaustausch zu ermöglichen. Die Abbildung 3 stellt die verfügbaren Module dar.

Alle Informationen von den verschiedenen TwinCAT-Modulen werden permanent und in Echtzeit an das Database-Modul übermittelt. Über den TwinCAT 3 Database Server werden diese Daten in einer Datenbank verwaltet: Sie werden

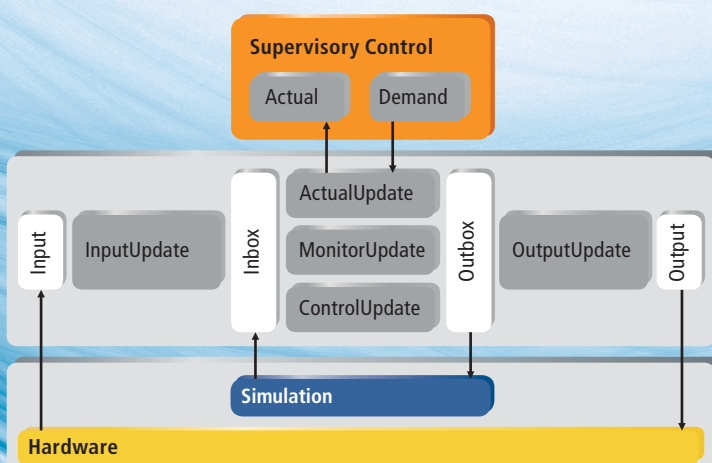


Abb. 2: Die einheitliche Architektur der Subsysteme erzeugt einen Standard in der Applikation und ermöglicht eine schnelle Einarbeitung der Programmierer.

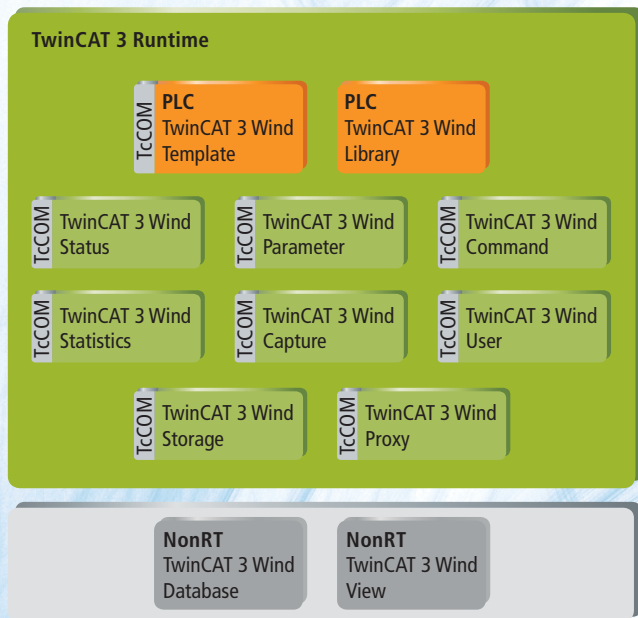


Abb. 3: Das TwinCAT 3 Wind Framework stellt die Funktionalitäten über TwinCAT-Module bereit, sowie eine PLC Bibliothek, die Schnittstelle zur Datenbank und eine Visualisierung.

vom Database-Modul entsprechend vorbereitet und über SQL-Befehle in die Datenbank eingefügt oder abgerufen. Die Anbindung an einen Microsoft SQL Server wird dabei vollständig vom Database-Modul umgesetzt. Das entsprechende Datenbank-Schema, inklusive der Tabellen und Prozeduren für den Microsoft SQL Server, wird vom Wind Framework mitgeliefert. Die weitreichende Verwendung spiegelt sich in den nachfolgend beschriebenen Modulen wieder.

TF6420 – TwinCAT 3 Database Server

Ein Datenaustausch aus der TwinCAT-Echtzeit mit verschiedenen Datenbanken ermöglicht der TwinCAT 3 Database Server. Es können SQL-Befehle wie Insert oder Select sowie Stored Procedures verwendet werden. Dabei werden aktuell elf Datenbanken unterstützt, unter anderem Microsoft SQL, MySQL, PostgreSQL oder Oracle. Zusätzlich werden ein Konfigurator für die visuelle Einstellung der Parameter und eine SPS-Bibliothek mitgeliefert, welche SPS-Funktionsbausteine zum Ausführen der SQL-Befehle bietet.

In der Automatisierung ist es üblich, die Benutzerverwaltung in einer externen Visualisierung oder dem SCADA-System abzudecken. In den Funktionen des TwinCAT 3 Wind Framework ist die Benutzerverwaltung integriert, sodass alle Interaktionen durch einen Benutzer über das User-Modul geprüft, verwaltet und protokolliert werden können. Damit kann für jede Funktion bereits in der Programmierung festgelegt werden, welche Rechte notwendig sind, um diese zu verwenden. Das Überprüfen dieser Benutzerrechte geschieht in der Applikation. So gewährleistet die Betriebsführung eigenständig, unabhängig von einer externen Verwaltung, den korrekten Zugriff durch einen Benutzer.

Ein sicheres Speichern von Accounts wird über kryptografische Funktionen (Hashes) in der Datenbank gewährleistet. Die Authentifizierung geschieht über Name und Passwort, wodurch der hinterlegte Zugriffslevel abgerufen wird.

Anhand von Service-Schaltern an den Schaltschränken kann die örtliche Zugriffsberechtigung ermittelt werden, die angibt, ob der Benutzer aus der Ferne zugreift oder sich vor Ort befindet und somit höhere Rechte erhält.

Das Proxy-Modul stellt einen direkten Zugriff auf die Echtzeitdaten aller Module und Objekte bereit. Dieser Zugriff kann direkt über das TwinCAT-ADS-Protokoll erfolgen, um jegliche Eigenschaften eines Objektes oder die protokollierten Daten abzurufen. Auch die sichere und herstellerunabhängige Kommunikation über OPC UA oder die normkonforme Kommunikation aus der IEC 61400-25 wird durch weitere TwinCAT-3-Functions ermöglicht.

TF6100 – TwinCAT 3 OPC UA Server

Der TwinCAT 3 OPC UA Server ermöglicht die Kommunikation nach OPC Unified Architecture (IEC 62541). Als Wegbereiter zu Industrie 4.0 und dem Internet of Things gewährleistet OPC UA den sicheren, zuverlässigen und herstellerunabhängigen Transport von Rohdaten vom Sensor in der Fertigungsebene bis in die IT-Ebene und das ERP-System. Die Steuerung ermöglicht so eine objekt-orientierte Datenkommunikation für aktuelle und historische Daten, Alarmer, aber auch Dienste (Methoden) und stellt diese in einer Serviceorientierten Architektur (SoA) bereit.

TF6510 – TwinCAT 3 IEC 61850/IEC 61400-25

Mit der TwinCAT 3 IEC 61400-25 Function wird der Datenaustausch nach IEC 61850 von den in der IEC 61400-25 spezifizierten Objekten für Windenergieanlagen ermöglicht. Als Protokoll ist die Manufacturing Message Specification (MMS) realisiert, um die hierarchischen Datenobjekte zwischen der Windenergieanlage und einer Leitstelle zu kommunizieren. Der mitgelieferte TwinCAT Telecontrol Configurator unterstützt bei der Konfiguration des Datenmodells und erzeugt den entsprechenden SPS-Code.

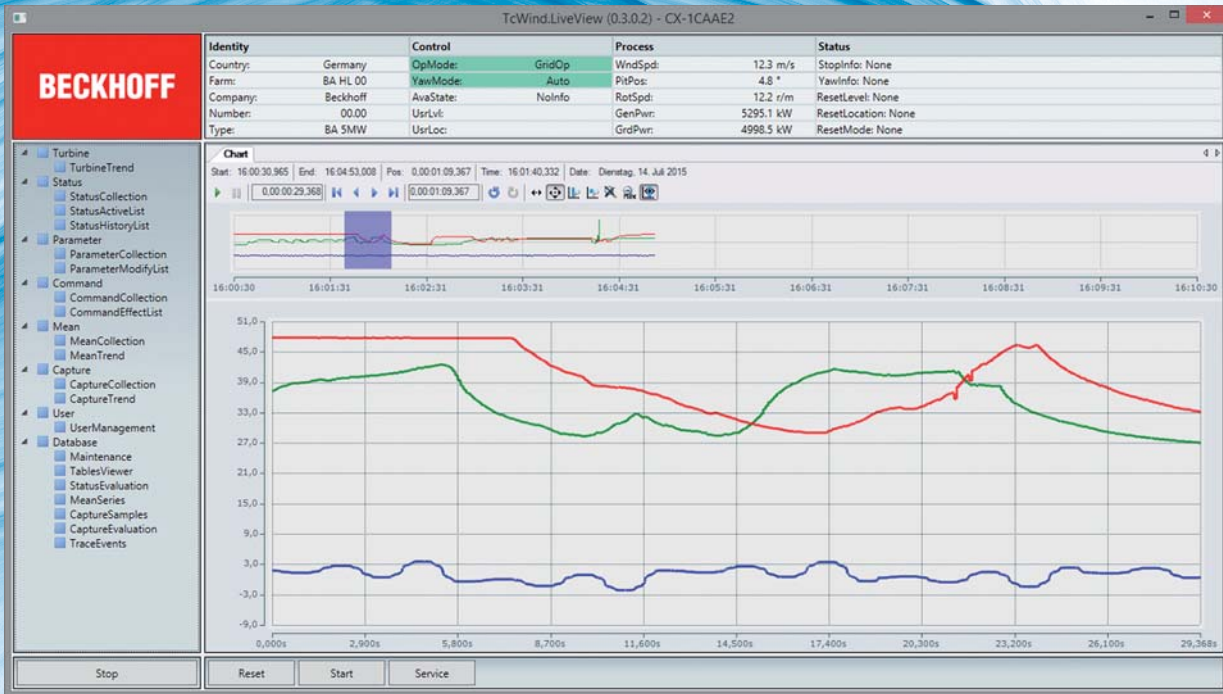


Abb. 4: In der Visualisierung werden aktuelle Zustände, Werte und Einstellungen dargestellt.

Mit dem TwinCAT 3 Wind Framework wird außerdem eine simple Engineering-Visualisierung zur Verfügung gestellt. Diese ruft die Daten vom Proxy-Modul ab und verwendet die integrierte Benutzerverwaltung des User-Moduls. Es lassen sich die aktuellen Zustände, Werte und Einstellungen aller Objekte darstellen. Einfache Auswertungen und Analysen der Datenbank sind möglich, und die Verläufe der aufgezeichneten Daten werden visualisiert. Eine solche Darstellung von aufgezeichneten Daten aus der Datenbank zeigt die Abbildung 4.

Ein weiterer Dienst ist das Status-Modul, welches eine Überwachung aller Komponenten der Windenergieanlage bietet: Es ermöglicht die Fehlererkennung, die Ereignisverwaltung sowie die Fehlerbehandlung und die Berichterstattung. Es werden Status-Objekte angelegt, welche jeweils ein Ereignis darstellen und zum Anzeigen von individuellen Meldungen, Warnungen oder Fehlern genutzt werden. Damit werden beispielsweise der Kabelbruch bei einem Sensor als Warnung oder die Fehlfunktion eines Aggregates als Fehler umgesetzt.

Jedes Status-Objekt hat verschiedene Eigenschaften, die frei konfigurierbar sind. Zur Identifizierung erhält jedes Objekt eine zugeordnete Gruppe, die der jeweiligen Komponente der Anlage entspricht, und einen Namen im Klartext. Verzögerungen zum Setzen und Rücksetzen des Ereignisses sowie verschiedene Modi für ein automatisches oder manuelles Rücksetzen sind möglich. So kann das Rücksetzen eines Fehlers nur einer autorisierten Personengruppe ermöglicht werden, indem ein benötigtes Zugriffslevel und eine örtliche Zugangsberechtigung eingestellt werden. Damit kann zum Beispiel festgelegt werden, dass das Rücksetzen eines Fehlers durch den Service-Mitarbeiter nur manuell durchge-

führt werden darf und auch nur, wenn dieser auch wirklich vor Ort ist und sich in der Gondel der Anlage befindet.

Ein Stoppen der Anlage kann optional für jedes Ereignis als Anlagenreaktion eingestellt werden, genauso wie das Auslösen einer hochaufgelösten Protokollierung der Anlagendaten oder das Versenden einer Benachrichtigung. Diese flexible Konfiguration von Ereignissen wird vom Status-Modul ausgewertet, und aus dem aktuellen Zustand aller Ereignisse werden die entsprechenden Reaktionen generiert. Ein Fehler im Pitch oder Umrichter kann beispielsweise so ausgewertet werden, dass als Reaktion ein Anlagenstopp erfolgt. Aufgrund der übergeordneten Überwachung aller Ereignisse ist der sichere Betrieb der Windenergieanlage gewährleistet. Eine Liste aller derzeit aktiven sowie eine Historie der zuletzt aufgetretenen Ereignisse werden vom Status-Modul verwaltet und können jederzeit abgerufen werden. Außerdem wird jedes Ereignis in der Datenbank protokolliert und mit Zeitstempeln versehen, wann das Ereignis aufgetreten ist und wann es zurückgesetzt wurde. Damit lassen sich Häufigkeit und Dauer des Auftretens jedes Ereignisses bestimmen, und Aussagen über den Betrieb oder die Verfügbarkeit der Anlagen treffen. Außerdem können die häufigsten Ursachen für etwaige Anlagenstillstände durch eine statistische Auswertung der Ereignisse ermittelt werden und schaffen die Voraussetzung zu einer Optimierung.

Das Parameter- und Command-Modul stellen Dienste zur Konfiguration und Interaktion mit der Applikation zur Verfügung. Ein Parameter-Objekt kann einen beliebigen Wert eines beliebigen Datentyps annehmen. So sind alle Datentypen aus dem IEC 61131-3-Standard vorbereitet, von denen auch Arrays als Vektoren

oder Tabellen eingesetzt werden können. Die Grenzen zur Überwachung einer Temperatur können beispielsweise als zwei Parameter-Objekte umgesetzt werden, deren Werte die minimal und maximal erlaubte Temperatur angeben oder zum Ein- und Ausschalten von Heizung dienen. Der Wert eines Parameters kann durch die Eigenschaften des Parameter-Objektes limitiert und die Möglichkeiten zur Änderung des Wertes können über Zugriffslevel eingeschränkt werden. Darüber hinaus wird ein Default-Wert für jeden Parameter eingestellt, auf den bei Bedarf zurückgesetzt werden kann.

So kann die gesamte Konfiguration der Windenergieanlage über Parameter abgebildet und geändert werden. Ein Protokollieren jeder Parameter-Änderung und ein persistentes Speichern und Laden von Konfigurationen wird über die Datenbankanbindung durch das Database-Modul ermöglicht. Damit ist die gesamte Konfiguration der Windenergieanlage in der Datenbank abgelegt und kann mit den Konfigurationen anderer Anlagen abgeglichen werden.

Mittels der Command-Objekte lassen sich Aktionen in der Applikation auslösen oder aktivieren. Jede Interaktion, zum Beispiel über einen Schalter an der Schaltschranktür oder einen Button aus der Visualisierung, kann über ein solches Command-Objekt umgesetzt werden. Hier werden verschiedene Modi für die Umsetzung als Druckschalter, Kippschalter oder Totmannschalter angeboten, deren Bedienung ebenfalls eine Autorisierung benötigt und vollständig in der Datenbank protokolliert wird. Zur besseren Diagnose und Visualisierung wird von jedem Command-Objekt eine Bestätigung und Rückmeldung ausgewertet. Mittels des Protokolls in der Datenbank lässt sich jederzeit sagen, welcher Mitarbeiter zu welcher Zeit einen manuellen Stopp der Anlage oder ein Rücksetzen der Ereignisse durchgeführt hat, genauso wie die Änderung von etwaigen Parametern.

Das Erfassen von Signalen und eine statistische Auswertung werden vom Capture- und dem Mean-Modul bereitgestellt. Rohdaten werden flexibel über Capture-Objekte aufgezeichnet. Die Art des Signals, ob digital oder analog, sowie die Abtastrate zur Aufzeichnung werden individuell für jedes Capture-Objekt eingestellt. Erste Auswertungen für die spätere Diagnose werden bereits in Echtzeit durchgeführt. Von einem digitalen Signal werden die Anzahl der Änderungen sowie die Dauer des aktiven Zustandes ausgewertet. So kann beispielsweise ein Aggregat und dessen Verhalten überwacht und die Schalthäufigkeit sowie Betriebsdauer einfach ausgelesen werden. Beim Tausch des Aggregates kann der Service-Mitarbeiter diese Statistik manuell zurücksetzen. Außerdem ermöglicht die optionale Integration eines analogen Signals die Berechnung von Werten wie Durchfluss, Energieerzeugung oder -verbrauch.

Würden solche Auswertungen außerhalb der Echtzeit durchgeführt, entstünden durch das nicht deterministische Aufzeichnen und Auswerten der Informationen große Abweichungen zum realen Wert. Die ermittelten Daten werden permanent in der Datenbank aufgezeichnet, lassen sich aber auch als momentane Werte abrufen und weiterverwenden.

Ein stetiges Ermitteln von gleitenden Mittelwerten wird vom Mean-Modul ermöglicht. Aus beliebigen analogen Signalen können mittels der Mean-Objekte die Mittelwerte über frei wählbare Zeitintervalle berechnet werden. Arithmetischer Mittelwert, Effektivwert, Standardabweichung und die Mittelwertberechnung anhand der Windrichtung werden angeboten, genauso wie das Ermitteln

von Minima, Maxima oder der Turbulenzintensität. In welchem Zeitintervall der gemittelte Wert in der Datenbank aufgezeichnet wird lässt sich frei konfigurieren. So kann man die typischen 30-Sekunden- oder 10-Minuten-Mittelwerte berechnen und archivieren. Statistiken, wie die Leistungskurve, Wind-Rose oder eine Capture-Matrix, lassen sich dadurch für beliebige Zeiträume und auf Anforderung aus der Datenbank generieren.

Bei bedeutenden Ereignissen, die durch ein Status-Objekt beschrieben werden, kann vom Trace-Modul das Schreiben eines hochaufgelösten Protokolls der Anlagendaten ausgelöst werden. Die Anlagendaten werden anhand der Capture-Objekte bereitgestellt und vom Trace-Modul in Zykluszeit aufgezeichnet. Dieses Protokoll beinhaltet außerdem Daten von konfigurierbaren Zeiträumen vor und nach dem Auslösen des Ereignisses. Damit sind die Daten von mehreren Sekunden vor dem Eintritt des Ereignisses verfügbar, wodurch die Ursache einfach ermittelt werden kann. Außerdem kann kontrolliert werden, wie die Anlage auf das Ereignis reagiert, da auch Daten von mehreren Sekunden nach dem Auftreten bereitstehen.

Jeder ermittelte Wert und jedes Ereignis werden mit einem Zeitstempel versehen. Dieser wird vom lokalen System verwendet, kann aber optional auch aus einer synchronisierten Zeitquelle bezogen werden. Dafür steht das Time-Modul zur Verfügung, welches die EtherCAT-Distributed-Clocks nutzt, um zum Beispiel über IEEE 1588 bzw. PTPv2 eine global synchronisierte Zeitquelle zu verwenden. Die Zyklen zum Aufzeichnen von Signalen mittels der Capture- oder Mean-Objekte werden ebenfalls anhand dieser Zeit synchronisiert. Es werden 10-Minuten-Mittelwerte aufgezeichnet – nicht von einem beliebigen Zeitpunkt ausgehend, sondern in einem festen 10-Minuten-Intervall um 11:00 Uhr, 11:10 Uhr usw.. Nur so sind die Daten von verschiedenen Quellen und Anlagen wirklich miteinander vergleichbar, da alle Werte zu einem identischen Zeitpunkt, mit identischen Zeitabständen ermittelt wurden.

Datenbankanbindung für detaillierte Analysen

Die Anbindung über das Database-Modul und den TwinCAT 3 Database Server an die SQL-Datenbank bietet eine effiziente und kompakte Datenhaltung über ein einheitliches und bekanntes Format. Das Aufzeichnen aller Ereignisse und Signale sowie das Speichern und Laden der gesamten Konfiguration aller Objekte ermöglichen detaillierte Analysen. Nötige Vorverarbeitungen geschehen durch die TwinCAT-Module in Echtzeit. So werden die Mittelwerte vom Mean-Modul durchgängig in jedem Zyklus der Applikation berechnet und jeder Wert aus jedem Zyklus geht in die Mittelung ein. Genauso werden das Abtasten von Werten und die Integrationen durch das Capture-Modul in jedem Zyklus ausgewertet, um die höchstmögliche Genauigkeit der Berechnung zu erreichen.

Durch das Erfassen und die Vorverarbeitung aller Daten in Echtzeit, welche dann zuverlässig in die Datenbank übertragen werden, können Auswertungen auf Anfrage und außerhalb der Echtzeitumgebung geschehen. Mit Hilfe dieser historischen Daten können Zustandsänderungen erkannt, die Ursachen von Störungen ermittelt, detaillierte Statistiken berechnet und Optimierungen der Anlage durchgeführt werden.

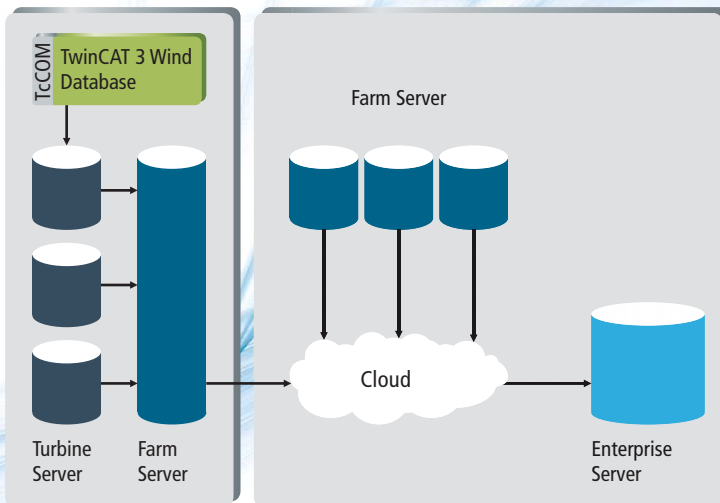


Abb. 5: Die Anlagendaten können auf einem zentralen Server zusammengefasst werden und ermöglichen übergeordnete Analysen.

Das Schema der Datenbank ist so vorbereitet, dass die Daten von einer einzelnen oder von multiplen Anlagen in einer Datenbank gesammelt und verwaltet werden können. So lassen sich über vorbereitete Prozeduren die Daten einfach zusammenführen, um eine übergeordnete Analyse und Vergleiche durchzuführen. Das Zusammenführen der Daten wird schematisch in Abbildung 5 gezeigt.

Werden die Daten aller Anlagen auf einem zentralen Unternehmensserver oder in der Cloud zu einem Data-Warehouse zusammengefasst, ist es möglich, die Daten dauerhaft und über die komplette Lebenszeit der Anlagen vorzuhalten. Solche Datenmengen von beliebig vielen Systemen, die in Echtzeit entstehen und auf zentralen Servern gesammelt werden, können allgemein als Big Data bezeichnet werden. Big Data stellt einen weiteren Baustein zu Industrie 4.0 dar und wird unterstützt durch die Möglichkeit, zusätzliche Daten aus dem Windpark-Management oder von Monitoring- und Messsystemen einzubinden (siehe Abbildung rechts).

Einheitlich zugänglich, ermöglichen diese Daten weitreichende und auch automatisierte Auswertungen. Es lassen sich Störungen oder Unregelmäßigkeiten leicht erkennen, Statistiken berechnen und Optimierungen der Betriebsführung durchführen, bis hin zur zustandsbasierten Überwachung und vorausschauenden Wartung von Anlagen. Mittels Data-Mining lassen sich ganz neue Kenntnisse über den Betrieb der Anlagen gewinnen. Es könnten beispielsweise Zusammenhänge zwischen dem Verschleiß von Aggregaten mit deren Schalthäufigkeit und Betriebsdauer ausgewertet werden, um zukünftig das Aggregat zu tauschen, bevor es ausfällt.

Fazit

Moderne Konzepte aus dem Software-Engineering und einer Serviceorientierten Architektur (SoA) wurden im TwinCAT 3 Wind Framework umgesetzt. Damit wird das breite Produktportfolio von Beckhoff um eine TwinCAT 3 Funktion für Windenergieanlagen erweitert. Mit der modularen Architektur, den Kommunikationsschnittstellen, der Datenbankbindung und der Möglichkeit, alle Daten zentral zu speichern sind die Voraussetzungen für Industrie 4.0 für Windenergieanlagen gegeben.

Es werden umfassende Dienste (Services) durch TwinCAT-Module bereitgestellt, in denen essentielle Aufgaben der Betriebsführung sowie die Benutzerverwaltung, die gesamte Protokollierung und Datenhaltung bereits umgesetzt sind. Ein Programmierer kann diese Funktionen über SPS-Funktionsblöcke nutzen und sich auf seine eigentlichen Aufgaben konzentrieren.

Durch das Applikations-Template ist bereits eine modulare Softwarearchitektur vorgefertigt. Die Applikation ist für Erweiterungen und Anpassungen an die spezifische Anlage vorbereitet. Eine Integration von Reglern oder Simulationen aus MATLAB®/Simulink® oder C/C++ ist vorgesehen und durch die flexible TwinCAT-3-Architektur einfach realisierbar.

Alle Informationen der Anlagen werden in Datenbanken bereitgestellt, und es wird die Voraussetzung geschaffen, um alle Daten in zentralen Datenbanken zusammenzufassen. Jede einzelne sowie die zentralen Datenbanken ermöglichen die weitreichende Überwachung und Auswertung des Zustandes bzw. der Betriebsmodi der Windenergieanlagen.

weitere Infos unter:

www.beckhoff.de/TwinCAT-Wind

Big Data

Optimierte Aufbereitung, Verfügbarkeit und Verarbeitung aller relevanten Daten in Echtzeit für Anlagenbetreiber und -hersteller

- Datensammlung und Data-Warehouse
- Datenauswertung und Data-Mining
- Power- und Condition Monitoring

Betreiber

Cloud
Big Data

Hersteller



Engineering

Durchgängiges und integratives Engineering über den gesamten Anlagenlebenszyklus

- IEC 61131-3, C/C++, MATLAB®/Simulink®
- Objektorientierung, Modularisierung
- Datenaustausch zwischen Engineering-Werkzeugen
- automatisiertes Engineering

Kommunikation

Sichere vertikale und horizontale Kommunikation

- Unterstützung aller gängigen Bussysteme (EtherCAT, Ethernet, PROFIBUS ...)
- durchgängige(s) Messaging/Connectivity (ADS, OPC UA, Live-Diagnose, ...)