

Engineering Report



Kunde

Hug Engineering AG
Im Geren 14
8352 Elsau
www.hug-engineering.com

Projekt

Die 1983 gegründete Hug Engineering AG mit Sitz im schweizerischen Elsau ist heute einer der führenden Hersteller von Abgasreinigungssystemen für Diesel- und Gasmotoren, die in stationären und mobilen Anwendungen eingesetzt werden.

Hug Engineering bietet effiziente Lösungen und massgeschneiderte Konzepte für die Abgasreinigung an. Als einer der wenigen Hersteller weltweit entwickelt, konstruiert und produziert Hug sämtliche Schlüsselkomponenten selbst.

Folgende Dienstleistungen durfte AVM Engineering AG für Hug Engineering AG umsetzen:

- Konzept HW & SW
- Implementation Basisapplikation
- Einsatz «UP»
- UI Implementation
- Maschinenkonfigurator
- Support

Wie kann eine komplexe Anwendung für verschiedenste Konfigurationsausprägungen und Auslieferungsvarianten komplett modular aufgebaut, implementiert und bei den Kunden in Betrieb genommen werden, wenn auf jeder Anlage aber immer nur der aktive Code vorhanden sein darf? Wie weit kann eine Modularisierung einer Anlage gehen und wo hört die Konfigurierbarkeit auf?

Herkömmlicher Ansatz → nicht erwünscht!

Die Aufgabenstellung ist bekannt. Der Maschinenbauer hat eine Standardanlage zu liefern, die aber für jeden Endkunden spezifisch konfiguriert werden muss. Um dies realisieren zu können, braucht es eine modulare Architektur und klare Schnittstellen, sowohl innerhalb der Software, aber auch bezüglich der entsprechenden Hardwarezugehörigkeit. Man spricht in diesem Zusammenhang gerne von einer 150% Lösung, das heisst, alle möglichen Variationen und Konfigurationen werden ausprogrammiert und für den entsprechenden Kunden der Funktionsumfang auf das notwendige reduziert, indem einzelne Optionen softwaretechnisch ausgeschaltet werden.

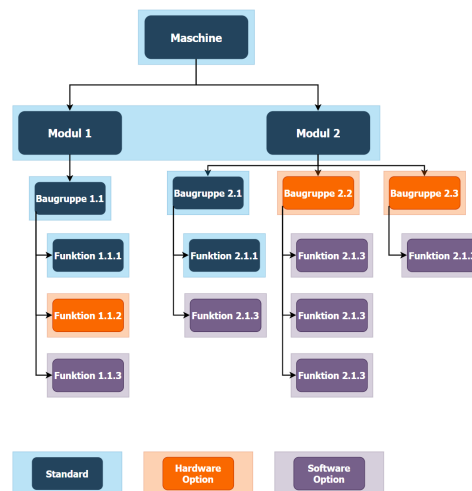


Abb. 1: Softwarearchitektur 150% Variante

Die auf der Maschine installierte Software wird so natürlich überladen mit Funktionen und Modulen, die auf dieser Maschine eventuell gar nie gebraucht, beziehungsweise ausgeführt werden.

Da die Bandbreite von Hug Engineering AG bei ihren Anlagen von einer Einzelanlage mit ei-

nem Dosiersystem bis zum Multidevicesystem mit redundantem Ressourcenmanagement und bis zu 20 unabhängigen Einzelstationen mit gemeinsamem Leitsystem gehen, war der herkömmliche Ansatz nicht erwünscht!

Framework, Architektur, Softwaremodell → Maschinenkonfigurator

Das Aufbrechen von Funktionen, das Verteilen von Aufgaben und die Implementation als wiederverwendbare Komponenten ist natürlich die Grundvoraussetzung, um eine beliebige Maschinenkonstellation komplett generisch zusammenstellen zu lassen. Diese Komponenten müssen clever gebildet und mit klaren Interfaces so gemacht werden, dass die Wiederverwendbarkeit gegeben ist.

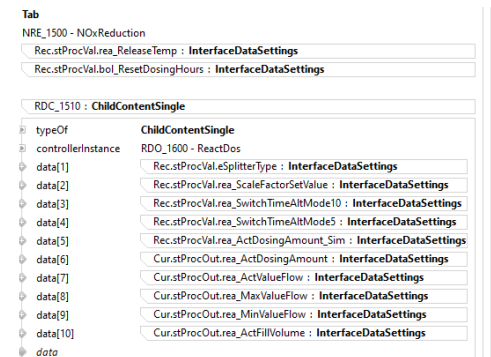


Abb. 2: Konfiguration des UI in «UP»

«UP» unterstützt diesen Prozess, gibt die Architektur vor, überprüft fortlaufend die Durchgängigkeit und bietet ein ganzes Framework an Systemfunktionalitäten, damit der Entwickler sich wirklich nur noch um die Businesslogik und nicht um die systemspezifische Programmierung kümmern muss. Durch die Unterstützung von «UP» und den mächtigen Möglichkeiten mit einer modellbasierten Software als Grundlage hat man einen durchgängigen Maschinenkonfigurator erschaffen, der

sowohl Hardwarekonfiguration, aber auch Elektroschemata, Visualisierung, Bedienungsanleitung, Servicebedienung und Softwareokumentation anlagenspezifisch generiert – und zwar wirklich nur das, was dem Kunden auch verkauft wurde.

Mehrfachnutzung von Inhalten - zentrale Verwaltung

Wie oft werden Daten aufgrund von Medienbrüchen oder unterschiedlichen Formatvorgaben mehrfach erfasst und die Übersichtlichkeit geht verloren, weil die Alarmursache und die dazu gehörende Fehlermeldung von zwei unterschiedlichen Entwicklern erschaffen wurde und diese nicht die gleiche Datengrundlage zur Verfügung hatten. So entstehen Fehlinterpretationen, redundante Datenhaltung und vor allem ineffiziente Arbeitsabläufe. Auch diesem Umstand begegnen wir mit «UP». Die zentrale Erfassung aller relevanten Maschinendaten und die einfache Adaption und automatische Formatumwandlung automatisieren viele Arbeiten und bringen eine deutliche Effizienzsteigerung. Ebenfalls wird es erst durch die Verknüpfung existierender Daten und der Verknüpfung mit einer generischen Software möglich, komplett individuelle Maschinensoftware, sogar hinunter bis auf den einzelnen Benutzer, zu ermöglichen.

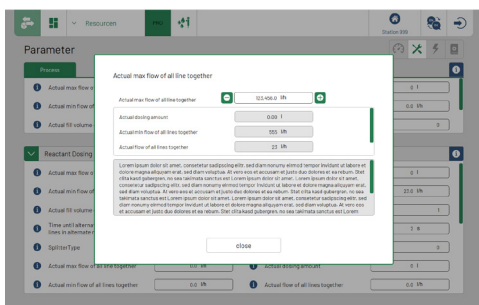


Abb. 3: individuelle Parameteransicht

Individuelle Lösungen werden generiert

Mit der effizienten Unterstützung von «UP» wurde ein Visualisierungskonzept realisiert, das bei manueller Ausprogrammierung aufgrund der anfallenden Entwicklungsarbeit sicher nicht umgesetzt worden wäre. So gibt es zum Beispiel für Parameter eine Möglichkeit bei der Konfiguration zu definieren, ob der Pa-

rameter als einfache Werteeingabe am UI angezeigt oder ob der Wert auf Wunsch in einem Dialogfenster mit zusätzlichen Informationen und interessanten Istwerten angereichert werden soll.

Diese Möglichkeiten sind sowohl auf Basis der Maschinenkonfiguration anwendbar, auf dem UI aber auch konsequent für jeden einzelnen Benutzer anpassbar. So kann sich der Operator sein Dashboard mit einigen wenigen Werten einfach und übersichtlich gestalten, während dem sich der Servicetechniker spezifische Kennzahlen und Detailinformationen anzeigen lassen möchte. Dass sich sowohl die Sprache, aber auch die Einheiten und je nach Kunde sogar das Design des UI dem Benutzer anpasst, versteht sich natürlich von selbst.

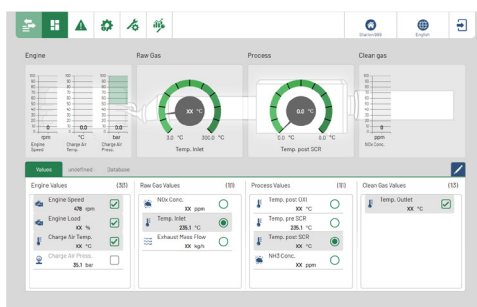


Abb. 4: userabhängiges Dashboard

Umsetzung mit «UP», mapView und B&R

Das UI ist mit mapView von B&R umgesetzt. Wobei neben den herkömmlich gelösten Standard-Bedienungssseiten wie Alarmierung, System Configuration oder UserHandling, ein generischer Ansatz für die Darstellung der Parameter gewählt wurde. Bei den generischen UI Seiten wird viel auf CompoundWidgets gesetzt, bei welchen sich mehrere Basis Widgets in der Darstellung mit erweitertem Funktionsumfang koppeln lassen. Dadurch wird die Lesbarkeit des Codes stark vereinfacht und zukünftige Änderungen können zentral für das gesamte UI implementiert werden. Ein weiterer Vorteil der generischen Lösung ist, dass nur das Generator Template angepasst werden muss, falls Design Anpassungen nötig sind. Dadurch kann einfach ein durchgängiges Design gewährleistet, verwaltet und weiterentwickelt werden.

Die Daten werden mit OPC UA zwischen UI und PLC ausgetauscht, wobei für die Parameter folgende Funktionalitäten des Standards genutzt werden: Einheiten, Limiten und User-role-Berechtigungen. Diese Eigenschaften werden zentral in «UP» verwaltet. Die OPC UA Konfiguration wird neben den sichtbaren Contents und damit einhergehende Binding Files ebenfalls generiert. Dadurch lassen sich in «UP» die Parameter-Seiten des UI mit wenigen Klicks editieren, wofür es herkömmlich mehrere Arbeitsschritte benötigen würde, welche an verschiedenen Orten der Software durchgeführt werden müssen und dadurch entsprechend fehleranfällig sind.

Beim Multidevicesystem mit bis zu 20 Einzelstationen, welche alle ein eigenständiges UI haben, wird mittels einer Kopfsteuerung zwischen den UIs der Stationen navigiert. Dabei ist das UI der Kopfsteuerung als Frame-Visu ausgeführt und über das WebViewer Widget lassen sich die WebVisu der Stationen laden. Neben dieser Navigation werden die wesentlichen Parameter sowie die aktiven Meldungen in einem Station Overview dargestellt. Die Anbindung der Daten von den Stationen lässt sich komplett mit mapView über OPC UA realisieren, ohne eine separate Querkommunikation zu implementieren.

Fazit

Durch die modellbasierte Softwareentwicklung, die «UP» ermöglicht, konnten die Anforderungen an eine höchst modulare, konfigurier- und wartbare Softwarelösung nicht nur erreicht, sondern übertroffen werden. Erweiterungen und neue Funktionalitäten können problemlos in die Software integriert und deren Auswirkungen mit automatisierten Tests überprüft werden. So entsteht eine Softwarebasis für die Anlagen von Hug Engineering AG, die über mehrere Jahre leben und weiterwachsen kann.

AVM Engineering AG
Patrick Schmidt

