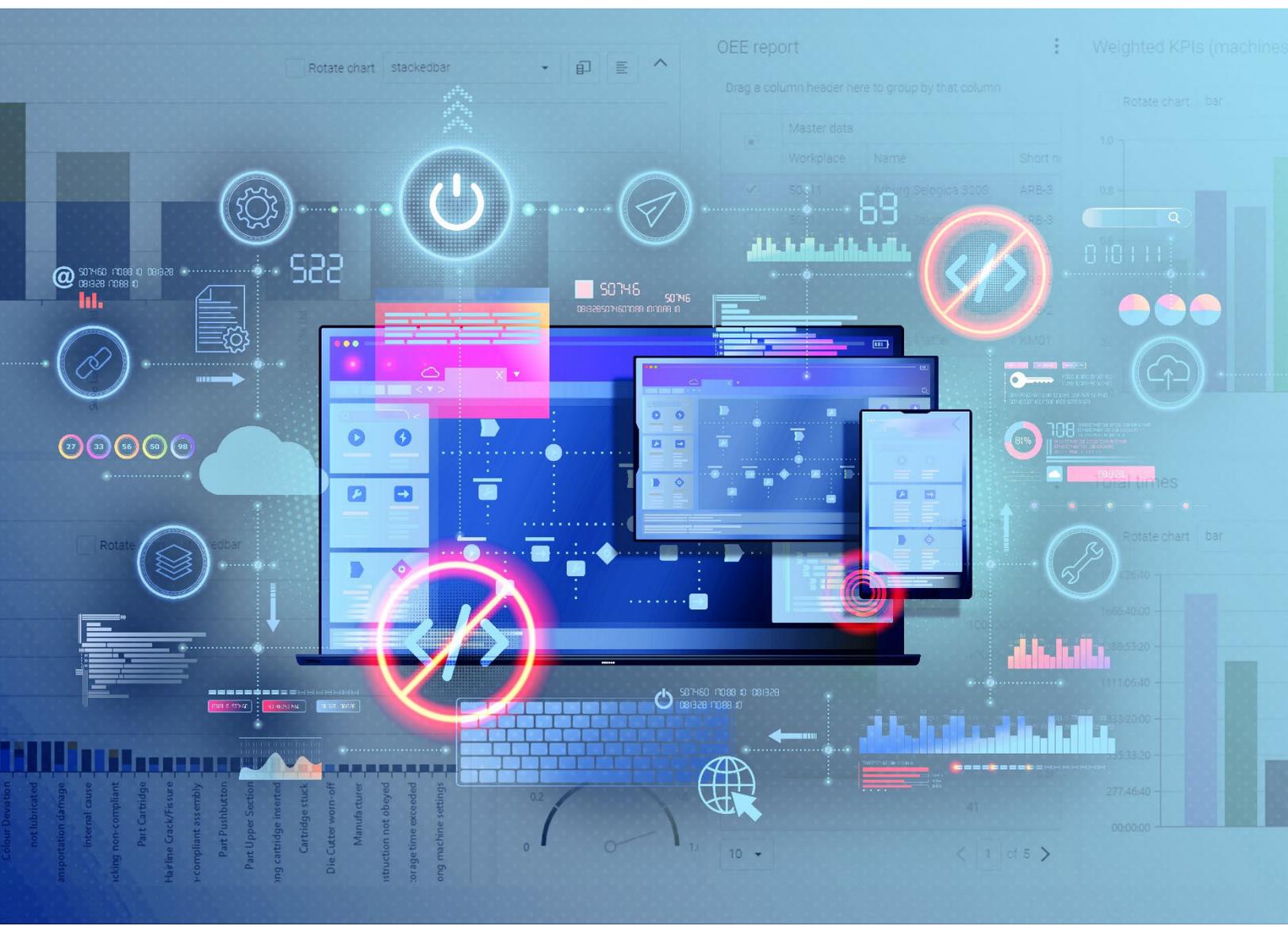


# Modellieren statt Programmieren



## Vorwort

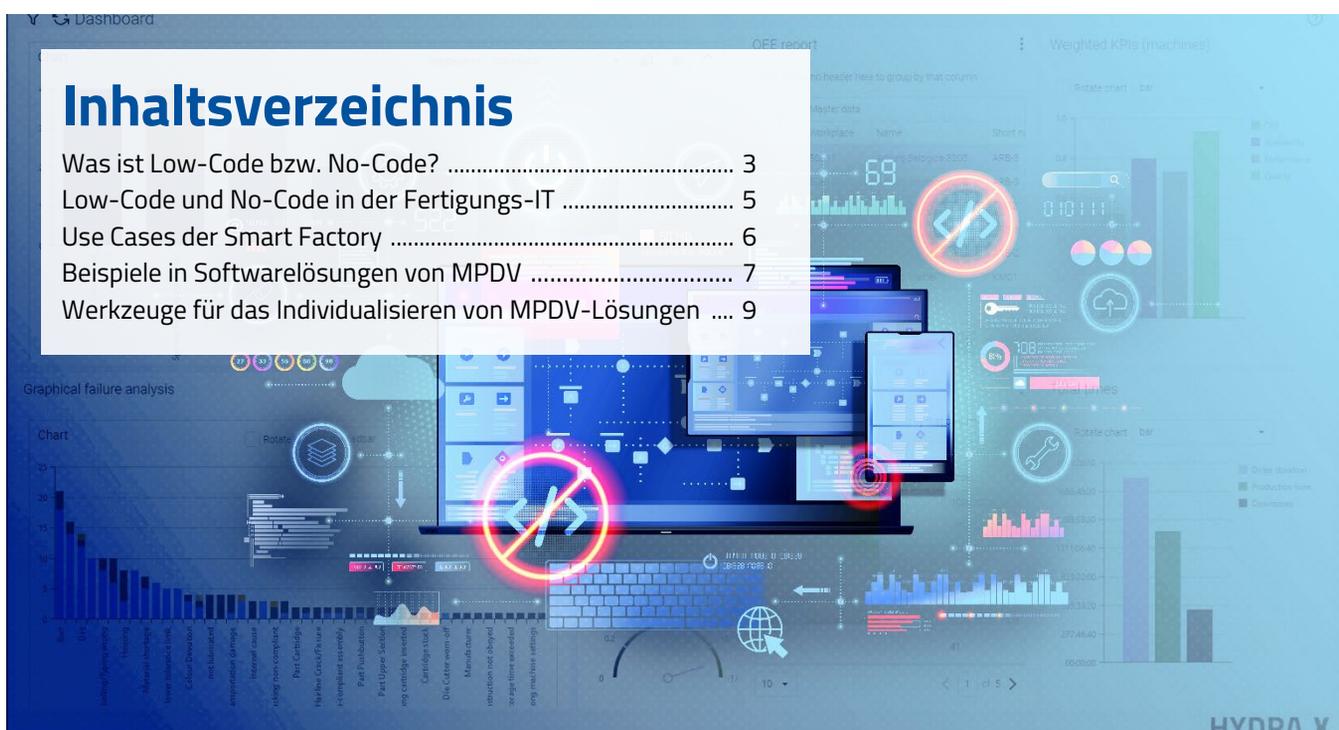
Kennen Sie das auch? Sie benutzen eine Business-Software und bleiben immer wieder an der gleichen Stelle hängen. Dann ärgern Sie sich jedes Mal erneut darüber, dass es einfach nicht zu Ihren Abläufen passt. Oder Sie haben eine klare Vorstellung davon, wie eine Software aussehen und funktionieren müsste, können selbst aber nicht programmieren oder wollen nicht in eine Individualprogrammierung investieren. Wenn Ihnen das nicht bekannt vorkommt, dann fragen Sie mal bei Ihren Mitarbeitenden in der Fertigung nach.

Heutzutage gibt es Möglichkeiten, dass Sie ganz einfach selbst in die Gestaltung und Funktionsweise von Software eingreifen können, ohne programmieren zu lernen. Auch das Erstellen von eigenen Anwendungen ist mittlerweile ohne Kenntnis einer Programmiersprache möglich: Die Zauberworte heißen **Modellieren** beziehungsweise **Low-Code** und **No-Code** – genau darum geht es in diesem Whitepaper. Als Anbieter von Softwarelösungen für die Smart Factory fokussieren wir uns auf Anwendungsfälle in Ihrer Produktion und sprechen über Beispiele aus der Fertigungs-IT.

Außerdem gehen wir darauf ein, wie Low-Code und No-Code mit unseren Lösungen funktioniert. Wenn Sie mehr über die Werkzeuge erfahren wollen, die wir zum Individualisieren unserer Lösungen anbieten, dann lesen Sie das Whitepaper Smart Factory Development Suite.

Viel Spaß beim Lesen.

Thorsten Strebel  
Geschäftsführer Products & Services bei MPDV



## Was ist Low-Code bzw. No-Code?

---

Im Zusammenhang mit Low-Code bzw. No-Code ist häufig die Rede von einer Low-Code Plattform bzw. No-Code Plattform. Das liegt daran, dass es einige Anbieter gibt, die entsprechende Werkzeuge in Form einer Plattform anbieten. Dieses Whitepaper behandelt allerdings keine Plattform, sondern die eigentlichen Methoden Low-Code und No-Code.

Low-Code bedeutet, dass für die Entwicklung bzw. Anpassung einer Anwendung deutlich weniger programmiert werden muss als bisher – einfacher gesagt, es muss weniger Quellcode erzeugt bzw. geändert werden. Meist kommen alternativ grafische Editoren oder anschauliche Beschreibungssprachen zum Einsatz. No-Code braucht gar keinen Code mehr. In beiden Fällen erleichtern Prinzipien wie Drag-and-drop das Zusammenstellen von Bedienoberflächen bzw. Abläufen und Zusammenhängen. Kurzum: es wird deutlich einfacher. Bildlich spricht man daher auch von "Modellieren statt Programmieren".

### Warum braucht es Modellierung überhaupt?

Die Frage nach dem Warum ist bei Low-Code und No-Code einfach zu beantworten: Der Mangel an Software-Entwicklern macht es einfach notwendig, dass auch Fachexperten Hand an die Software legen können. Und selbst wenn Software-Entwickler zur Verfügung stehen, hilft es, den Entwicklungsprozess zu beschleunigen, zum Beispiel durch Modellieren. Gleichzeitig fällt dadurch eine Barriere weg, die sich zwischen der IT-Abteilung und den Kollegen in der Fertigung bilden kann. Modellierung hilft dabei, Prozesse schneller und einfach in der digitalen Welt abzubilden. Dafür muss der Prozessowner wenig IT-Kenntnisse haben bzw. der IT-Kollege weniger vom Prozess verstehen. Low-Code ist also auch eine Brücke zwischen den Welten – sie verbindet Fachexperten mit der IT.

### Software-Entwicklung nicht nur für Entwickler

Damit wird die Entwicklung und Individualisierung von Applikationen ein Handlungsfeld, das nicht mehr nur den eigentlichen Software-Entwicklern vorbehalten ist. Jeder, der sich in seiner Domäne auskennt und sich auf einfache Gestaltungswerkzeuge einlässt, kann das heute tun. Soll zum Beispiel die Reihenfolge von drei Eingabefeldern geändert werden oder bei der Eingabe geprüft werden, ob der Zahlenwert größer fünf ist, so kann das mittels Low-Code erledigt werden. Ein Programmierer wird dafür nicht benötigt – immer vorausgesetzt, die eingesetzte Lösung unterstützt Low-Code.

### Vom Digital Native zum Citizen Developer

Insbesondere in Zeiten, in denen quasi jeder ein Smartphone besitzt und dieses intensiv benutzt, ist der Umgang mit Software etwas ganz Normales geworden. Die sogenannten Digital Natives kennen es gar nicht mehr anders – ein Leben ohne Software ist für sie kaum denkbar. Durch Low-Code und No-Code werden ganz normalen Menschen zu sogenannten Citizen Developer – also zu Laien-Entwicklern, die eine nötige Änderung an der Software ganz einfach und schnell selbst erledigen können. Selbst wenn eine Funktion fehlt, wird diese kurzerhand mit wenigen Klicks zusammengestellt und genutzt. Diese Darstellung ist überspitzt – in der Theorie aber durchaus denkbar.

### Abgrenzung zur klassischen Programmierung

Oftmals liest man auch, dass No-Code die Steigerung von Low-Code sei – also noch weniger bis gar kein Code mehr benötigt wird. Einerseits stimmt das, denn bei No-Code wird tatsächlich kein Code mehr benötigt, allerdings sind viele Plattformen und Anbieter dann so restriktiv, dass auch keine Programmierung mehr möglich ist. Das begrenzt die Flexibilität, was je nach Anwendungsfall hinderlich sein kann.

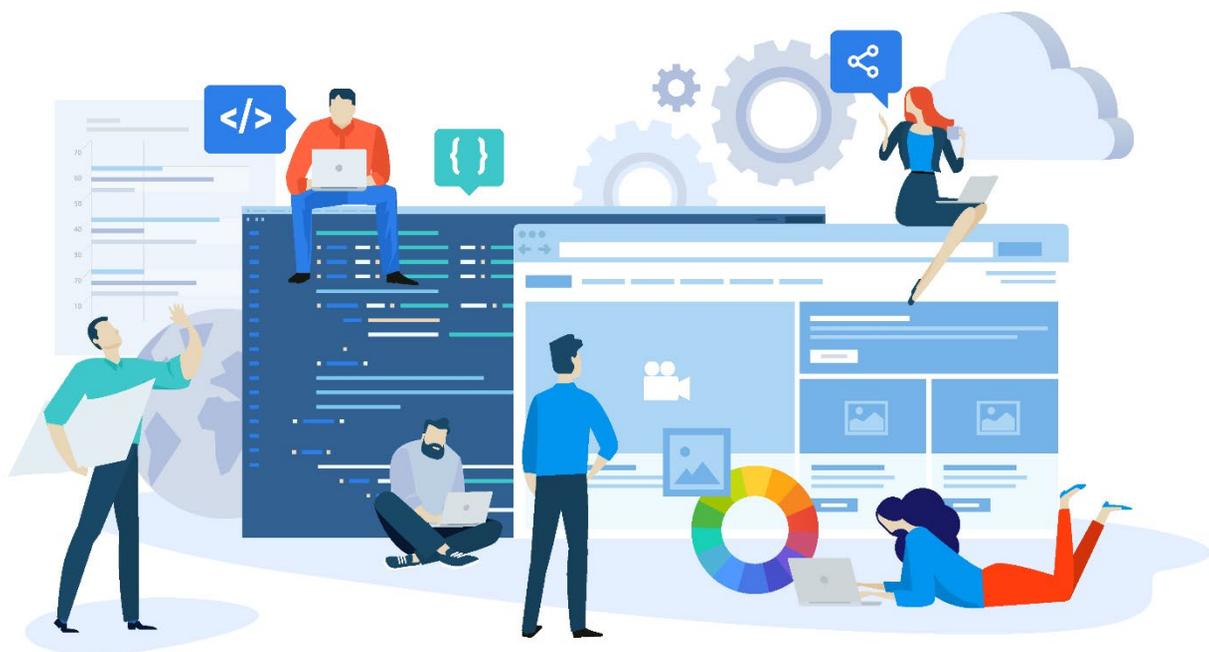
MPDV bietet neben der klassischen Programmierung beides an und definiert die Begriffe Low-Code und No-Code wie folgt:

**No-Code:** Das Anpassen und Erstellen von Anwendungen erfolgt mittels Assistenten, grafischer Modellierung sowie Templates und komplett ohne Programmierung. Diese Methode eignet sich besonders für einfache Anwendungsfälle wie das Zusammenstellen eines Dashboards.

**Low-Code:** Durch eine einfache Beschreibungssprache können bestehende Anwendungen verändert und neue Anwendungen erstellt werden – auch ohne explizite Kenntnis einer klassischen Programmiersprache und der dahinter liegenden Technik (z. B. HTML5). Diese Methode bietet deutlich mehr Flexibilität, erfordert jedoch grundlegende technische Kompetenzen.

Das **klassische Programmieren** (häufig auch Pro-Code oder Full-Code genannt) mit etablierten Sprachen wie .net, Java oder C++ ermöglicht zwar volle Flexibilität, braucht aber leider auch deutlich mehr IT-Know-how. Mittels User Exits und Scripting kann nahezu jedes Vorhaben umgesetzt werden.

Dabei entsteht ein Spannungsdreieck aus Flexibilität, Komplexität und dem Umgang mit Seiteneffekten. Je mehr Flexibilität benötigt wird, desto mehr Komplexität muss der Bediener beim Individualisieren akzeptieren. Auch auf Seiteneffekte muss er enorm achten. Wird eine hohe Flexibilität gefordert, führt das oftmals zur klassischen Programmierung. Wenn der Anwender mit weniger Individualisierungsmöglichkeiten auskommt und möglichst keine Seiteneffekte auslösen möchte, so klingt das eher nach No-Code. Damit lassen sich aber nur sehr einfache Änderungen durchführen, was die Komplexität in Grenzen hält. Daher sollte in jedem Anwendungsfall entschieden werden, welche Methodik dafür am besten geeignet ist.



### Vorteile der Modellierung

Low-Code und No-Code machen es also möglich, dass Fachexperten unabhängiger von IT-Experten werden. Implementierte Regelwerke stellen sicher, dass keine unerlaubten Veränderungen vorgenommen werden und dass das Gesamtkonstrukt der Software lauffähig bleibt. Zudem helfen vordefinierte Templates dabei, dass sich die Ergebnisse auch optisch harmonisch in das Gesamtbild einfügen. Somit können Anforderungen aus dem Anwendungsumfeld schneller, einfacher und konsequenter umgesetzt werden. In Zeiten eines globalen und stetig zunehmenden Wettbewerbs sind Low-Code und No-Code also hervorragende und heutzutage notwendige Werkzeuge – auch in der Smart Factory.

## Low-Code und No-Code in der Fertigungs-IT

---

Als führender Anbieter von Produkten, Services und Lösungen für die Smart Factory fokussiert sich MPDV auf Fertigungs-IT. Auch diese Softwarekategorie bietet ein breites Feld an Möglichkeiten, Low-Code und No-Code gewinnbringend einzusetzen.

Je härter der Wettbewerb wird, desto wichtiger ist es, dass sich Unternehmen von der Masse abheben. Dies geschieht meist durch das gezielte Optimieren von Prozessen. Möchte man diese Prozesse dann digital abbilden, so stößt man oftmals auf die Herausforderung, dass Sonderlösungen und Anpassungen nötig sind. Viele Unternehmen greifen dann zu individuell für sie programmierten Softwarelösungen und verzichten auf die Vorteile von Standardsoftware. Viele davon bereuen diese Entscheidung irgendwann.

### Individuelllösung vs. Standardsoftware

Wie der Name schon sagt, ist eine Individuelllösung speziell für diesen Anwender entwickelt und bildet dessen Anforderungen exakt ab. Auf der anderen Seite steht die Standardsoftware, die zwar nicht jeden Sonderfall abdeckt, dafür aber von vielen Anwendern gleichzeitig genutzt wird und daher meist stabiler läuft, ständig weiterentwickelt wird, besser gewartet werden kann und so in Summe günstiger im Betrieb ist. Es scheint, als müsse man sich zwischen einer teuren Ideal-Lösung und einer günstigen „Notlösung“ entscheiden. Glücklicherweise ist das heute nicht mehr so: Viele Standardsoftwarelösungen bieten Möglichkeiten, die Anwendungen in bestimmtem Umfang zu individualisieren. Zum Beispiel können Bedienoberflächen selbst gestaltet oder Programmabläufe angepasst werden. Genau an dieser Stelle entscheidet der Softwareanbieter, mit welchen Werkzeugen und Methoden eine Individualisierung möglich ist. Neben der klassischen Programmierung bieten immer mehr Softwarelösungen die Möglichkeit, sowohl Bedienoberflächen als auch Programmabläufe mittels Low-Code oder No-Code zu modellieren.



*Standardsoftware bietet viele Vorteile, doch manchmal braucht es individuelle Lösungen. (Quelle: MPDV)*

## Use Cases der Smart Factory

---

Die folgenden Use Cases sollen exemplarisch zeigen, welche Einsatzmöglichkeiten es für Low-Code und No-Code in der Fertigungs-IT gibt und welche Herausforderungen damit gemeistert werden können. Im Anschluss daran werden spezifische Lösungen von MPDV vorgestellt.

### Individuelles Dashboard

Einer der wichtigsten Erfolgsfaktoren der Smart Factory ist die Transparenz. Dashboards eignen sich hervorragend, um eine solche Transparenz herzustellen. Sie zeigen Kennzahlen, Auswertungen, Listen und Grafiken aller Art. Je besser die gezeigten Informationen auf die Bedürfnisse der Mitarbeitenden abgestimmt sind, desto nützlicher ist das Dashboard. Mittels Modellierung, also mit Hilfe von Templates und modularen Bausteinen kann jeder Mitarbeiter sein individuelles Dashboard zusammenstellen.

### Manuelle Eingaben im Shopfloor

Auch das manuelle Erfassen von Daten im Shopfloor sollte möglichst gut an die Abläufe in der Fertigung angepasst werden. In manchen Unternehmen wird zum Beispiel bereits beim Anmelden eines Auftrags geprüft, ob Paletten für die fertigen Produkte zur Verfügung stehen. Oder es sollen bestimmte Informationen zu genutzten Werkzeugen eingegeben werden. Wenn die Standardfunktionen der Fertigungs-IT diese Eingaben nicht vorsehen, können sie mittels Low-Code nachgerüstet werden. Auch die Reihenfolge, in der ein Werker Daten eingibt, ist in vielen Unternehmen unterschiedlich. Mit Low-Code oder No-Code ist eine solche Anpassung ganz einfach möglich.

### Verbuchen und verarbeiten

Sind Daten erst einmal im System, so gilt es, diese gewinnbringend zu nutzen. Aber auch hier gibt es in der Smart Factory teils komplexe Anforderungen. Wenn zum Beispiel beim Herstellen von Metallteilen nur die Anzahl der einzelnen Teile erfasst wird, im ERP-System aber die Zahl der Kartons oder Kisten erwartet wird, so kann eine einfache Umrechnung hier die Lösung sein. Wenn die Kisten oder Kartons unterschiedlich groß sind, dann macht gegebenenfalls auch eine weitere Eingabemaske Sinn. Beides lässt sich mittels Modellierung genauso implementieren wie das automatische Benachrichtigen von Experten, wenn die zu verbuchenden Daten einmal nicht in das erwartete Raster passen. Im Idealfall kann an jeden Verarbeitungsschritt eine individuelle Aktion angehängt werden.



### Komplexe Montagevorgänge

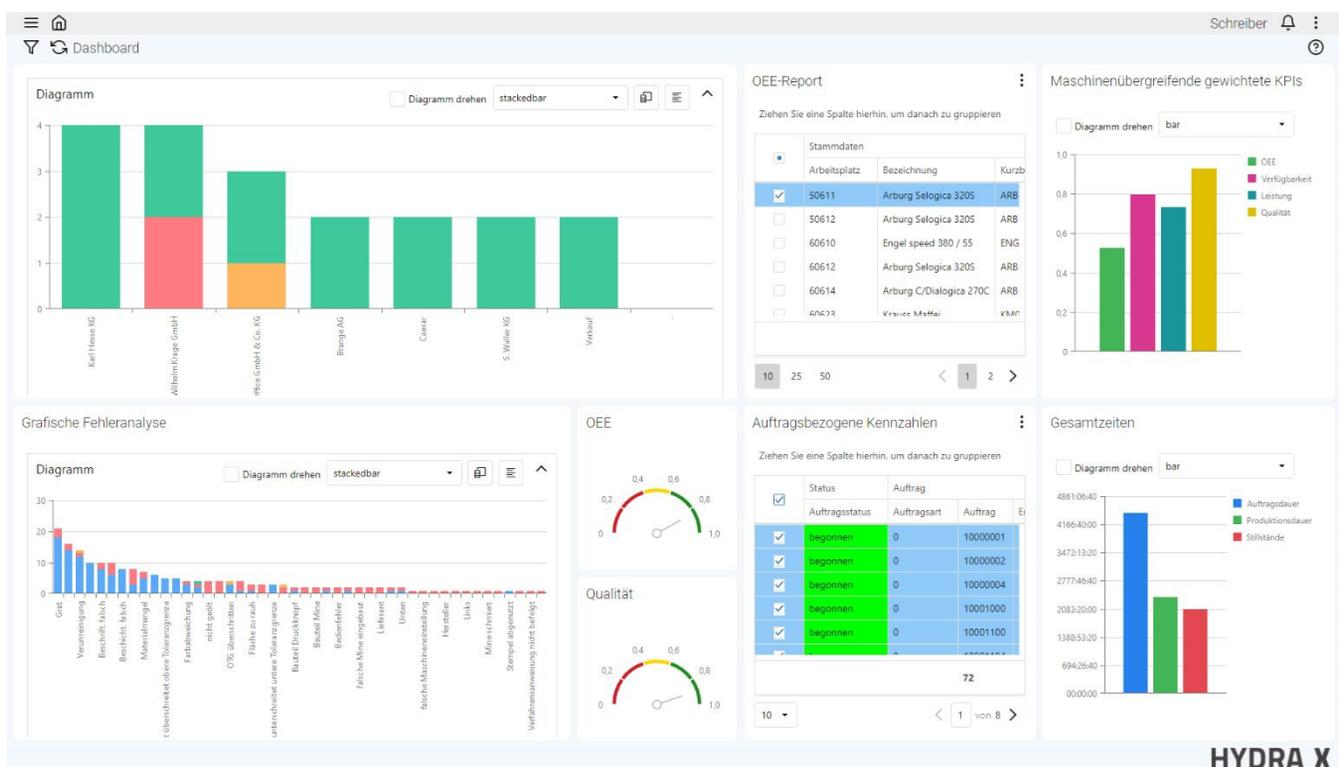
In der variantenreichen Serienfertigung sind komplexe Montageprozesse an der Tagesordnung. Das übliche Abbilden von Arbeitsplätzen und Arbeitsgängen reicht hier nicht mehr aus – es braucht eine weitere Detaillierung. Arbeitsstationen, Arbeitsschritte und Workpieces sind Objekte, die eine detailliertere Modellierung von Montageprozessen erlauben. Zudem gilt es, angeschlossene Peripherie wie intelligente Schrauber, Barcode-Leser oder Pick-by-Light Systeme anzubinden. Letztendlich müssen sowohl schrittweise Anweisungen hinterlegt als auch deren Ausführung überwacht werden. Bislang werden insbesondere Montagelinien durch eine sogenannte Kopfsteuerung auf SPS-Basis überwacht und gesteuert. Dies erforderte jedoch, dass jeder Schritt in einer meist komplexen Programmiersprache kodiert werden muss. Durch das Modellieren mittels Low-Code und No-Code wird die Montage deutlich flexibler.

## Beispiele in Softwarelösungen von MPDV

Im Folgenden werden die zuvor genannten Use Cases konkretisiert und am Beispiel der Softwarelösungen von MPDV erläutert.

### Dashboards in HYDRA X

Das Manufacturing Execution System (MES) HYDRA X von MPDV besteht aus einer Sammlung sogenannter mApps. Eine davon bietet vorgefertigte Dashboards an. Diese können jedoch bedarfsgerecht an die Anforderungen jedes Unternehmens und jedes Anwenders angepasst werden. Der integrierte UI-Designer bietet dazu viele einfach zu bedienende Möglichkeiten, die auf No-Code Methoden basieren. Zum Beispiel können einzelne Auswertungen, die MPDV in Form sogenannter Widgets zur Verfügung stellt, beliebig angeordnet und in der Größe verändert werden. Die Anpassungen erfolgen intuitiv mittels Klicken und Ziehen. Es können sowohl weitere Widgets platziert als auch nicht benötigte Widgets entfernt werden. Durch Eingeben von einfachen Parametern legt der Benutzer fest, auf welche Objekte innerhalb der Smart Factory sich das jeweilige Widget beziehen soll. Der Kreativität sind hierbei keine Grenzen gesetzt, denn mit dem iFrame-Widget können ganz einfach auch externe Inhalte eingebunden werden – zum Beispiel aus dem Intranet. Einmal gespeichert kann das individuelle Dashboard auch anderen Anwendern zur Verfügung gestellt werden und bleibt auch nach einem Software-Update erhalten.



*Individuell gestaltete Dashboards sind in HYDRA X ganz einfach. (Quelle: MPDV)*

## Meldedialoge individualisieren

Sämtliche Meldedialoge in HYDRA X bestehen aus einem oder mehreren Meldeschritten. Man kann sowohl die Reihenfolge der Schritte ändern als auch Schritte ergänzen oder Schritte hinzufügen. In jedem Meldeschritt können die Bedienoberflächen als Widgets individualisiert und parametrisiert werden. All das erfolgt in einer einfachen Beschreibungssprache, also in Low-Code. Innerhalb der Meldeschritte besteht die Möglichkeit, Daten anzuzeigen und Informationen abzufragen. Jedes Element kann beliebig angeordnet werden. Auch das Platzieren und Konfigurieren von Buttons ist ganz einfach möglich. Am konkreten Beispiel „Arbeitsgang anmelden“ zeigt sich, dass dieser Meldedialog aus mehreren Meldeschritten besteht, von denen zwei aktiv genutzt werden: Arbeitsplatzstatus ändern und Arbeitsgang anmelden. Weitere mögliche Schritte sind Material anmelden oder Werkzeug anmelden. Für jedes Element innerhalb des Meldedialogs gibt es eine Beschreibung, die zur Laufzeit interpretiert wird.

The screenshot displays the 'Expertenmodus - DialogLoggenOperationComplete' interface. On the left, a tree view shows the configuration of the 'Arbeitsgang anmelden' dialog, including elements like 'DialogLoggenOperationComplete', 'DialogLoggenOperation', and various data and widget configurations. A red box highlights the 'Arbeitsgang anmelden' widget, and a blue box highlights the 'Liste aller Meldeschritte' widget. On the right, a preview window shows the 'Arbeitsgang anmelden' dialog form. The form includes a table of steps (e.g., '1. PRODUKTION', '2. BESTELLEN') and a list of materials (e.g., '15. WOLFRAM', '30. NICKEL'). The interface is annotated with red and blue lines connecting the configuration elements to the corresponding UI components.

Meldedialoge lassen sich in HYDRA X mittels Low-Code modellieren. (Quelle: MPDV)

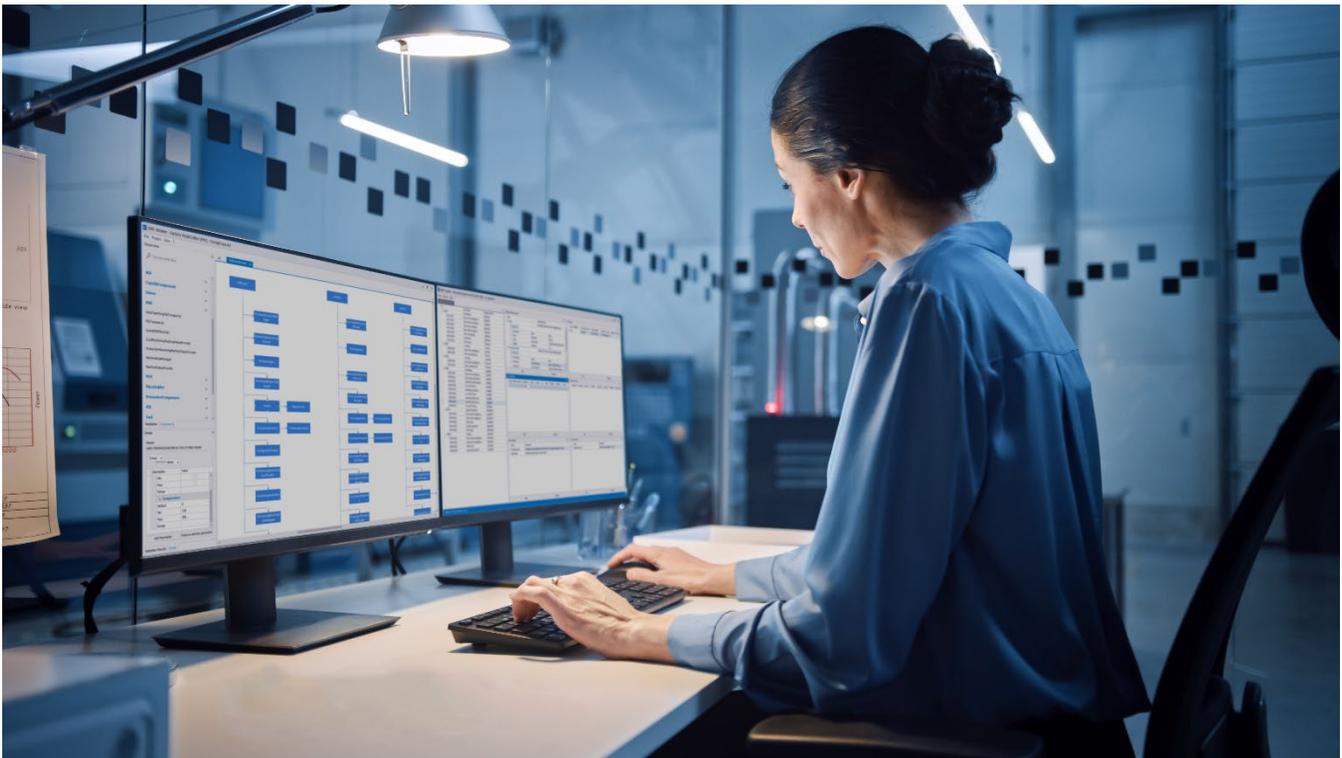
## Datenverarbeitung deskriptiv modellieren

Die aktuellen Versionen von HYDRA und die Integrationsplattform MIP sind so aufgebaut, dass die Erfassung und Verbuchung von Daten nicht mehr klassisch programmiert ist, sondern deskriptiv auf der Meta-Ebene modelliert werden kann. Hierzu bietet MPDV verschiedene Möglichkeiten. Einerseits gibt es vorgedachte Konfigurationsmöglichkeiten. Zum Beispiel wird eine bestimmte Plausibilitätsprüfung aktiviert oder deaktiviert. Auch der Umfang der Prüfung kann eingestellt werden. Andererseits gibt es in den Verarbeitungsprozessen definierte Stellen, an denen eigene Verarbeitungsschritte integriert werden können. Solche Schritte werden mit einer deskriptiven Beschreibungssprache (Low-Code) modelliert. Außerdem besteht an vielen Stellen der Datenerfassung die Möglichkeit, eigene Logiken zu implementieren. Zum Beispiel können erfasste Werte addiert werden. Auch logische Operationen sind möglich: z. B. wenn Signal Y und Signal X gleich sind, dann schreibe eine Eins in das Datenfeld A.

Bisher erfolgt das Modellieren von Erfassungs- und Verarbeitungsprozessen in Low-Code. In Zukunft wird MPDV immer mehr grafische Editoren (No-Code) einführen, sodass auch die deskriptive Beschreibungssprache immer seltener benötigt wird.

## Assembly Management

Das Assembly Management ist ein Teil von HYDRA X und bildet komplexe Montageprozesse digital im MES ab. Hierzu bietet MPDV einerseits den grafischen Factory Model Editor (FME), mit dem einzelne Arbeitsstationen mittels No-Code und Low-Code modelliert werden können, und andererseits den Manufacturing Instruction Editor (MIE), der bei der digitalen Abbildung von Arbeitsschritten unterstützt. Der MIE basiert auf dem Low-Code Prinzip. Aus der Kombination aus Factory Model und Manufacturing Instructions generiert HYDRA zur Laufzeit sowohl die kontextbezogene Werkerführung bzw. Werkerassistenz als auch eine Vorlage für die Dokumentation aller vorgesehenen Arbeitsschritte. Das Assembly Management kann auf diese Weise sowohl vollautomatische Montagelinien abbilden als auch lose verkettete, manuelle Montagearbeitsplätze. Sogar das Industrie 4.0 Prinzip „das Material sucht sich eigenständig den Weg durch die Produktion“ lässt sich damit realisieren – alles in Low-Code. Dafür werden keine teuren und ohnehin wenig verfügbaren SPS-Programmierer benötigt.



*Montageprozesse können mit HYDRA X Assembly Management ganz einfach modelliert werden.  
(Quelle: MPDV, Adobe Stock, Gorodenkoff)*

## Werkzeuge für das Individualisieren von MPDV-Lösungen

Innerhalb des MPDV Produktportfolios fasst die Smart Factory Development Suite alle Werkzeuge zusammen, die man zum Individualisieren bestehender und zum Entwickeln neuer Anwendungen braucht. Die Development Suite selbst besteht aus vier Komponenten:

- UI Development Suite
- Business Logic Development Suite
- Enterprise Connectivity Development Suite
- Label & Report Designer

Mehr dazu erfahren Sie in unserem **Whitepaper Smart Factory Development Suite**.

Gleich anfordern unter [whitepaper.mpdv.com](http://whitepaper.mpdv.com)!

## MPDV-Whitepaper

---

### Wissen ist Macht!

Unsere Whitepaper bieten Ihnen Wissenswertes zu Fertigungs-IT und Industrie 4.0 in kompakter Form. Neben interessanten Fachartikeln, Trendberichten und Produktinformationen enthalten die Whitepaper auch spannende Experten-Interviews und nützliche Checklisten für die Praxis.

Hier eine Auswahl unserer Whitepaper:

### Smart Factory Development Suite

---

#### Smart Factory Elements

---

#### Die funktional vernetzte Fabrik

---

#### Die selbstregelnde Fabrik

---

#### Die reaktionsfähige Fabrik

---

#### In vier Stufen zur Smart Factory

---

#### Mit Kennzahlen die Produktion im Griff

---

#### Plattformen & Ökosysteme

---

#### Vom 4-Stufen-Modell zum Regelkreis



**Jetzt weitere Whitepaper anfordern!**  
[whitepaper.mpdv.com](http://whitepaper.mpdv.com)

## Über uns

---

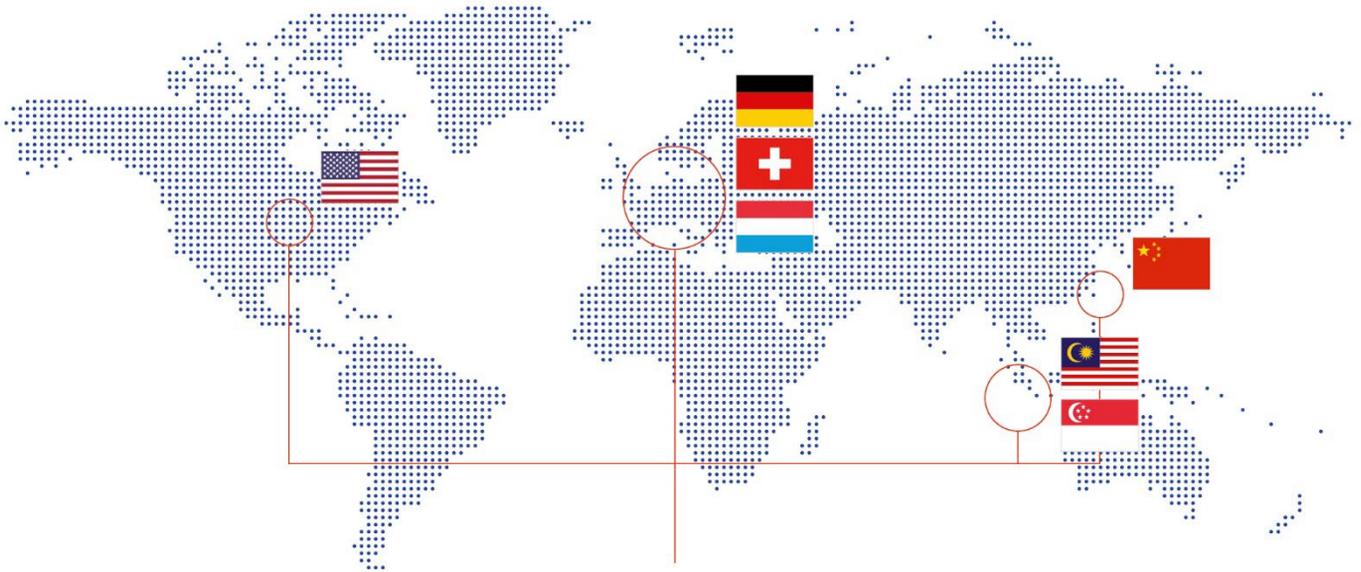


### MPDV Mikrolab GmbH

mit Hauptsitz in Mosbach ist der Marktführer für IT-Lösungen in der Fertigung. Mit mehr als 45 Jahren Projekterfahrung im Produktionsumfeld verfügt MPDV über umfangreiches Fachwissen und unterstützt Unternehmen jeder Größe auf ihrem Weg zur Smart Factory.

Produkte von MPDV wie das Manufacturing Execution System (MES) HYDRA, das Advanced Planning and Scheduling System (APS) FEDRA oder die Integrationsplattform Manufacturing Integration Platform (MIP) ermöglichen es Fertigungsunternehmen, ihre Produktionsprozesse effizienter zu gestalten und dem Wettbewerb so einen Schritt voraus zu sein. In Echtzeit lassen sich mit den Systemen fertigungsnahe Daten entlang der gesamten Wertschöpfungskette erfassen und auswerten. Verzögert sich der Produktionsprozess, erkennen Mitarbeiter das sofort und können gezielt Maßnahmen einleiten.

Täglich nutzen weltweit mehr als 1.100.000 Menschen in über 1.750 Fertigungsunternehmen die innovativen Softwarelösungen von MPDV. Dazu zählen namhafte Unternehmen aller Branchen. Die MPDV Gruppe beschäftigt rund 520 Mitarbeiter an 13 Standorten in Deutschland, China, Luxemburg, Malaysia, der Schweiz, Singapur und den USA.



Chicago · Hamburg · Hamm · Heidelberg · Kuala Lumpur · Luxemburg  
Mosbach · München · Serrig · Shanghai · Singapur · Stuttgart · Winterthur



---

MPDV Mikrolab GmbH · Römerring 1 · 74821 Mosbach  
+49 6261 9209-0 · [info@mpdv.com](mailto:info@mpdv.com) · [www.mpdv.com](http://www.mpdv.com)