

MPDV-Lösungen einfach individualisieren

Smart Factory Development Suite



Vorwort

Immer komplexere und individuellere Anforderungen an die Fertigungs-IT sorgen dafür, dass Standardanwendungen angepasst werden müssen. Im Idealfall erfolgen solche Anpassungen durch Modellierung und nicht durch Veränderungen am Quellcode (Programmierung). Man spricht beim Modellieren auch von Low-Code oder No-Code, da man eben ohne Code auskommt oder zumindest nur wenig einfachen Code in Form einer Beschreibungssprache erzeugen beziehungsweise verändern muss. Was genau hinter Low-Code und No-Code steckt, erfahren Sie in unserem **Whitepaper Modellieren statt Programmieren**, welches Sie ganz einfach unter whitepaper.mpdv.com anfordern können.

Damit Individualisierungen noch einfacher werden, hat MPDV die Smart Factory Development Suite entwickelt. Diese Suite fasst innerhalb des MPDV Produktportfolios alle Werkzeuge zusammen, die man zum Individualisieren bestehender und zum Entwickeln neuer Anwendungen benötigt. Die Development Suite selbst besteht aus vier Komponenten, die in diesem Whitepaper erläutert werden:

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

Was man mit diesen vier Komponenten machen kann und wie sie funktionieren, erfahren Sie auf den nächsten Seiten.

Viel Spaß beim Lesen.

Thorsten Strebel
Geschäftsführer Products & Services bei MPDV



UI Development Suite

Bedienoberflächen von mApps auf der Manufacturing Integration Platform (MIP) können 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.

Während der UI-Designer, mit dem man bestehende **Widgets** anordnet (No-Code), zum Lieferumfang gehört, bietet die UI Development Suite die Möglichkeit, neue Widgets zu erstellen oder bestehende Widgets anzupassen. Dies erfolgt über eine einfache Beschreibungssprache und ist somit der Methode Low-Code zuzuordnen. Widgets wiederum bestehen aus einzelnen Komponenten (sogenannten **Bricks**), zum Beispiel Eingabefelder, Tabellen, Diagramme oder Grafiken. Das Erscheinungsbild wie auch die Anordnung der Bricks innerhalb des Widgets wird mittels Low-Code modelliert. Dafür kommt eine einfache Beschreibungssprache zum Einsatz.

The screenshot displays the UI Development Suite interface. On the left, there is a filter sidebar for 'OEE-Report' with options for 'Zeitraum' (30.1.2021, 06:00 to 10.6.2021, 00:00) and 'Arbeitsplatz' (Arbeitsplatz, Gruppe, Auswertungsgruppe: MECHAN_PRODUCTION). The main area shows a table of machine data with columns for 'Stammobjekt', 'Bezeichnung', 'Kurzbezeichnung', 'OEE', and 'Verfügbarkeit'. A table with 6 rows is visible, with the first two rows highlighted in blue. Below the table, there are two charts: 'Kennzahlen Maschinenbezogen' and 'Maschinenübergreifende gewichtete Kennzahlen'. A 'Widget' configuration dialog is open, showing a list of 'Bricks' (OEE-Report, Line-Selbstliste, Filter, Hauptgewicht, OEE-Report, Kennzahlen-Maschinenbezogen, Maschinenübergreifende gewichtete Kennzahlen) and a 'Data' section. A 'Widget' configuration panel is also visible on the right, showing a list of 'Bricks' (OEE-Report, Line-Selbstliste, Filter, Hauptgewicht, OEE-Report, Kennzahlen-Maschinenbezogen, Maschinenübergreifende gewichtete Kennzahlen) and a 'Data' section. Red arrows point from the 'Widget' configuration panel to the table and charts, and yellow arrows point from the 'Brick' configuration panel to the table and charts.

UI Development Suite im Einsatz: Widgets (rot) bestehen aus konfigurierbaren Bricks (gelb). (Quelle: MPDV)

Damit mehrere Widgets miteinander interagieren können, wird eine gemeinsame Datenbindung modelliert, die für diese Anwendung selektierte Daten bereitstellt. So können zum Beispiel in einem Widget Maschinen ausgewählt werden und in einem anderen Widget der gleichen Anwendung werden Kennzahlen dieser Maschinen visualisiert. Die Widgets untereinander haben keine Verbindung – die Verknüpfung erfolgt über die darunterliegende Datenquelle. Diese Zweistufigkeit erleichtert die Modellierung und reduziert unerwünschte Nebeneffekte.

Die Bricks werden von MPDV zur Verfügung gestellt, können mittels Programmierung aber auch selbst entwickelt werden – zum Beispiel um eine neue Art von Diagramm in das Portfolio aufzunehmen.

Bei der Gestaltung von Bedienoberflächen zeigt sich, dass unterschiedliche Entwicklungsmethoden von Vorteil sind:

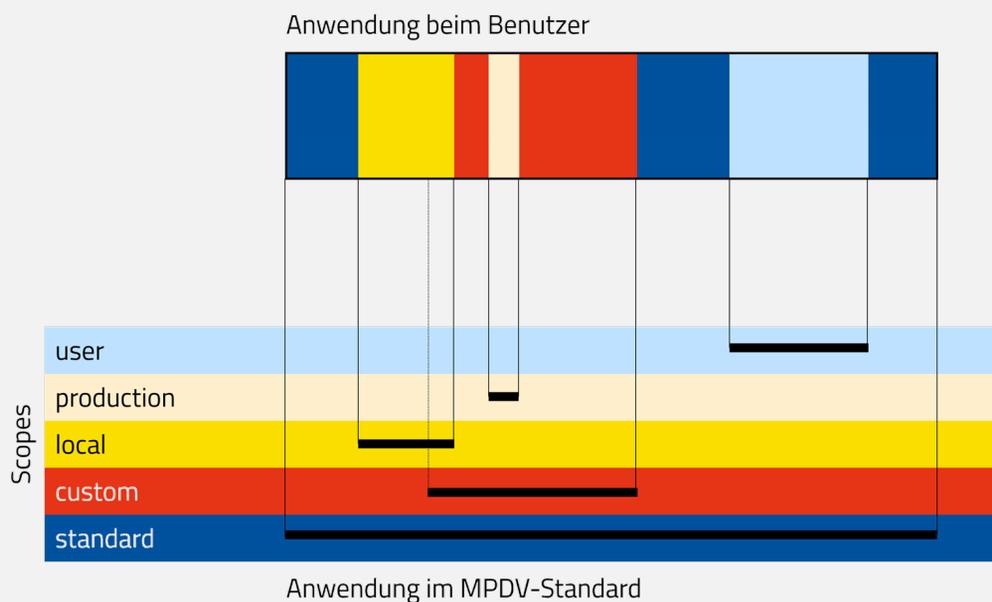
- No-Code zur **Anordnung** von Widgets innerhalb einer Anwendung
- Low-Code zur Gestaltung und Konfiguration der einzelnen **Widgets** auf Basis von Bricks
- Programmierung zur Erstellung und Anpassung einzelner **Bricks**

Hintergrundinformation: Scopes

Damit nicht jede Anpassung gleich den vom Hersteller ausgelieferten Standard verändert, hat MPDV das Scopes-Prinzip entwickelt und bei allen Produkten implementiert. Dieses Prinzip stellt sicher, dass Anpassungen feingranular durchgeführt werden können und damit den Standard nicht komplett überschreiben, sondern nur an der gewünschten Stelle übersteuern. Die sogenannten Patches enthalten die gewünschten Änderungen und werden auf die Standardmodellierung angewendet. Zunächst einmal gibt es vier Scopes, die aber nach Bedarf um zusätzliche Scopes erweitert werden können:

- **standard**: Hier befinden sich alle Anwendungen und Vorgaben seitens MPDV, die für alle gelten, die dieses Produkt verwenden.
- **custom**: Wenn kundenspezifische Anpassungen erfolgen, so werden diese hier abgelegt und gelten nur für diesen Kunden.
- **local**: In diesem Scope kann der Kunde eigene Anpassungen ablegen, an denen MPDV nicht beteiligt ist.
- **user**: Insbesondere bei Bedienoberflächen hat jeder Benutzer die Möglichkeit, eigene Anpassungen zu speichern, die nur für ihn selbst gelten.

Innerhalb der Scopes werden nur die Unterschiede zum Standard gespeichert. Diese überschreiben dann den jeweils darunter liegenden Scope.



Durch das Scopes-Prinzip werden Anpassungen quasi Release-fähig. (Quelle: MPDV)

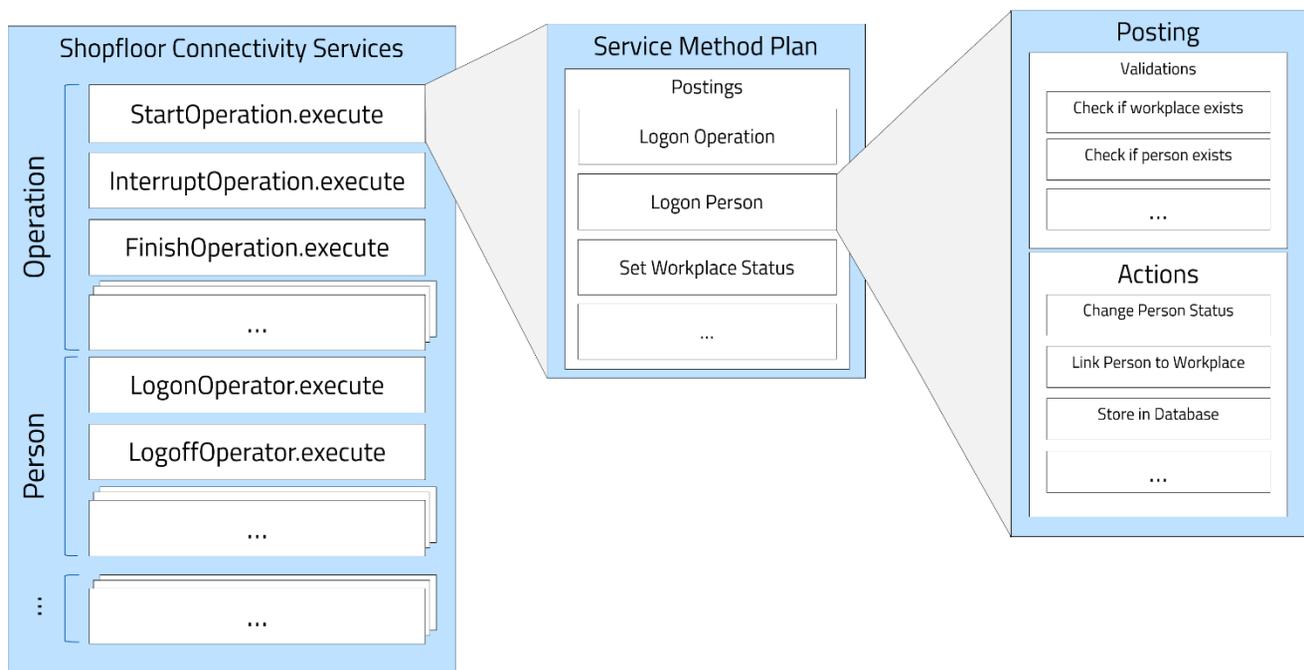
Ein Beispiel: MPDV liefert eine Anwendung aus, die neben einer Tabelle auch zwei Diagramme anzeigt (standard). Speziell für einen Kunden wurden die Farben im Diagramm so angepasst, dass nur noch blau und gelb darin vorkommen (custom). Der Kunde selbst hat festgelegt, dass in der Tabelle bestimmte Spalten nicht angezeigt werden (local). Und einer der Anwender hat die Position der beiden Diagramme vertauscht (user). Ändert MPDV nun die Anwendung und ergänzt ein drittes Diagramm, so wird diese auch an den zuvor genannten Kunden ausgeliefert (standard). Die Änderung der Farben bleibt davon unberührt und ist folglich immer noch aktiv (custom). Auch die angepassten Spalten bleiben erhalten (local). Die vertauschten Diagramme werden zwar immer noch in der vom Benutzer festgelegten Art und Weise angezeigt, allerdings könnte durch das neue, dritte Diagramm eine Repositionierung nötig sein (user). Wenn der Kunde nun für einen bestimmten Bereich innerhalb der Fertigung dieses neue Diagramm nicht haben möchte, so kann er das über einen eigenen Scope (z. B. production) einstellen und an die betroffenen Mitarbeitenden verteilen. In der Regel positioniert sich dieser Scope oberhalb von "local" und überschreibt somit alles, was von MPDV kommt bzw. selbst angepasst wurde. Eigene Einstellungen der Benutzer (user) werden nicht überschrieben.

In Summe stellt das Scopes-Prinzip sicher, dass man durch eigene Anpassungen nicht von der Weiterentwicklung des MPDV Standards abgeschnitten wird.

Business Logic Development Suite

In der Business Logic Development Suite geht es darum, Abläufe zum Erfassen und Verarbeiten von Daten zu entwickeln oder zu individualisieren. Auch hier kommt eine Beschreibungssprache zum Einsatz (Low-Code). Einerseits verfügen die Verarbeitungsabläufe in MPDV-Lösungen über diverse Konfigurationsmöglichkeiten und „Schalter“, die bestimmte Schritte aktivieren oder deaktivieren. Andererseits können Anwender eigene Schritte einfügen. Solche Schritte können zum Beispiel eine Plausibilisierung oder eine Umrechnung sein. Auch das Aggregieren von Daten ist möglich. Zur Entwicklung eigener Verarbeitungsschritte wird ebenfalls Low-Code genutzt. Komplexere Vorgänge können aber auch mittels Scripting programmiert werden.

Ein Blick auf die Shopfloor Connectivity Services der Manufacturing Integration Platform (MIP) soll das verdeutlichen. Beim Anmelden eines Arbeitsgangs werden im Meldedialog weitere Daten abgefragt, die zu weiteren Aktionen führen können: zum Beispiel „Person anmelden“, „Status setzen“ oder „Werkzeug anmelden“. Dabei sollte man prüfen, ob die anzumeldende Person überhaupt anwesend und dazu berechtigt ist. Eine solche Plausibilisierung kann aber auch bewusst deaktiviert werden, wenn keine personenbezogene Erfassung erfolgt. All das kann mittels Low-Code in einer anschaulichen Beschreibungssprache realisiert werden.



Geschachtelter Aufbau von Shopfloor Connectivity Services: Beim Anmelden eines Arbeitsgangs wird auch eine Person angemeldet und der Arbeitsplatz-Status gesetzt. (Quelle: MPDV)

Hintergrundinformation: Namensräume

Ein weiteres Konzept, um Konflikte zu verhindern, wenn mehrere Parteien an einer Lösung arbeiten, ist die Definition von Namensräumen. Das Prinzip funktioniert relativ einfach: Sämtliche Objekte und Services bekommen eine Art Vorsilbe (Präfix). Zum Beispiel bekommt der Kunde ABC den Namensraum "abc_". Somit beginnen die Namen aller Objekte, die Kunde ABC erstellt mit "abc_" – also "abc_pallet", wenn er ein Objekt erstellen möchte, das einer Palette entspricht. Dieses Vorgehen stellt sicher, dass kein Konflikt entsteht, wenn MPDV ebenfalls ein Objekt Namens Palette erstellt. Sowohl Partner als auch Kunden können von MPDV einen eigenen Namensraum bekommen.

Eigene Services erstellen

Da die MIP als semantische Basis für alle MPDV-Lösungen und die mApps vieler MIP-Partner eine modellierte Service-Schnittstelle hat, kann auch diese ganz einfach erweitert und angepasst werden. Wenn es zum Beispiel darum geht, ein neues Objekt in das Informationsmodell der MIP einzufügen, dann kann dies mit einer Kombination aus No-Code und Low-Code erfolgen. Hierzu bietet MPDV im Rahmen der Business Logic Development Suite den Repository Client an.

Im Repository Client lassen sich Objekte mittels No-Code modellieren. Die dazu gehörenden Zugriffsservices können ebenfalls angelegt werden. In Fachkreisen spricht man dabei von den CRUD-Services. Das steht für create (erstellen), read (auslesen), update (ändern) und delete (löschen). Die Parameter der Zugriffsservices korrelieren in der Regel mit den Datenbankfeldern einer relationalen Datenbank, die man in der klassischen Programmierung nutzen würde.

The screenshot displays the MPDV Repository Client interface. The main window is divided into several panes. On the left, there is a 'Domains' pane showing a tree view of domains like 'person', 'PersonnelShiftData', 'BOPerson', etc. The central pane shows a list of 'Services' with columns for Domain Set, Domain, Name, Function, Service Type, DLG, List Mode, System Call, Parent, Modified, Is Invalid, Validation Result, Version, and Category. Below this, there are panes for 'ServiceParameters' and 'DomainAttributes'. The interface includes various toolbars and filters for navigating through the data.

Der MPDV Repository Client listet alle verfügbaren Objekte und Services. Hier können auch eigene Objekte erstellt werden.
(Quelle: MPDV)

Neben den Zugriffsservices können im Repository Client auch Auswerteservices modelliert werden – natürlich auch mit Low-Code und Beschreibungssprache. Diese Beschreibungssprache wird dann zur Laufzeit interpretiert, was das Testen und Optimieren von Services besonders einfach macht. Die Anzeige der ausgewerteten Daten erfolgt dann in der grafischen Bedienoberfläche, die mit der UI Development Suite modelliert werden kann.

Services testen

Soll die Funktion eines Services getestet werden, so bietet der Service Tester als Teil der Business Logic Development Suite eine einfache Möglichkeit dazu. Dafür braucht der Service Tester eine aktive Verbindung zur MIP. Zunächst fragt die Anwendung alle verfügbaren Services der MIP ab und prüft, welche davon mit den aktuellen Berechtigungen ausgeführt werden können. Im nächsten Schritt wählt der Anwender einen Service aus und gibt die gewünschten Parameter an. Nach dem Ausführen des Services wird das Ergebnis angezeigt. Außerdem besteht die Möglichkeit, die Syntax des Serviceaufrufs zu kopieren, um sie in eine selbstentwickelte mApp zu übernehmen. Somit ist auch der Service Tester ein nützliches Tool zur Entwicklung eigener Anwendungen für die MIP und andere MPDV-Lösungen.

Ein Package Creator zur Verteilung von Software rundet den Umfang der Business Logic Development Suite ab. Damit können mehrere Änderungen zu einem Paket zusammengefasst werden, um sie mit wenig Aufwand auf mehrere Systeme zu verteilen.

Enterprise Connectivity Development Suite

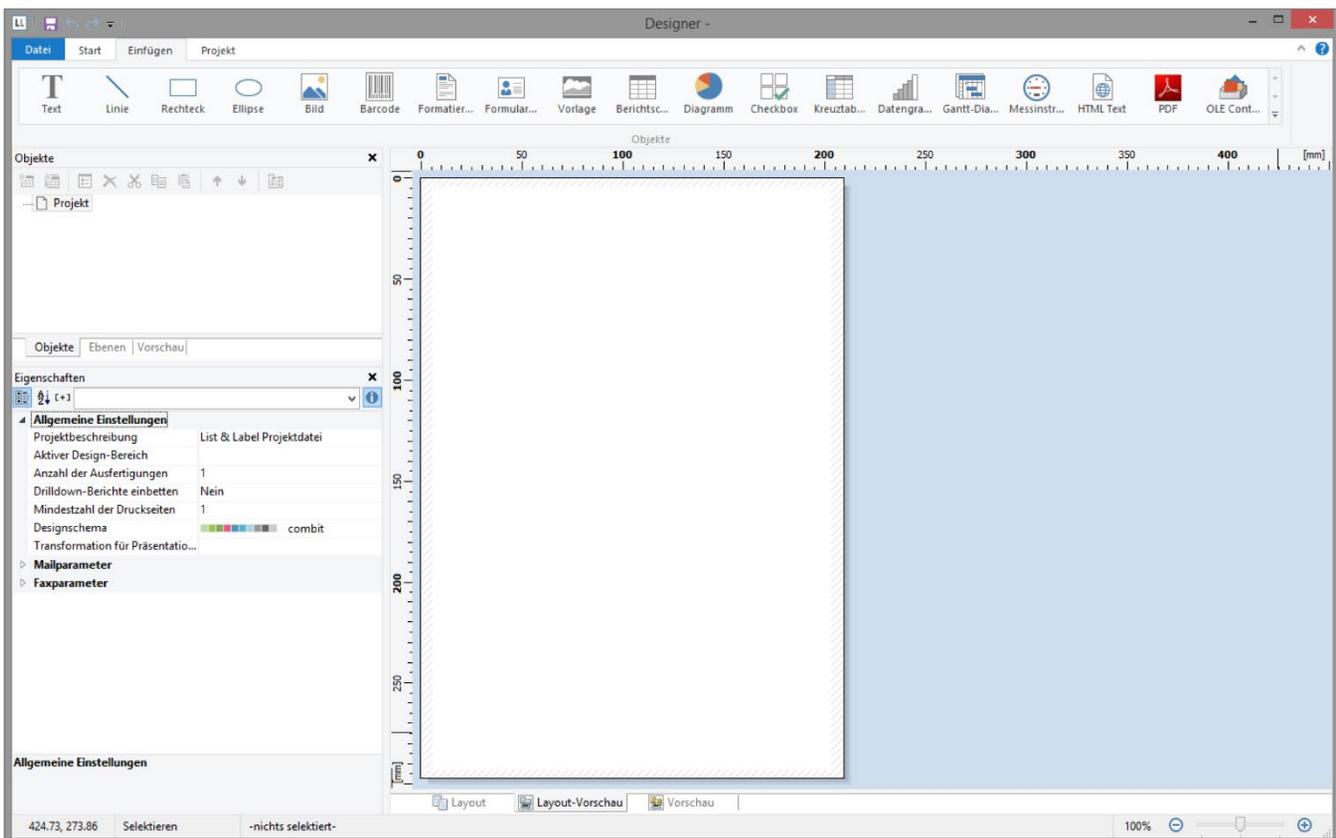
Immer dann, wenn ein ERP-System oder eine andere Softwarelösung angebunden werden soll, aus der Daten übernommen werden, braucht es eine verlässliche Schnittstelle. Viele ERP-Systeme lassen sich mittels Standard-Schnittstelle an die Lösungen von MPDV anbinden – aber leider nicht alle. Die Enterprise Connectivity Development Suite bietet für alle Anbindungen, die nicht im Standard realisiert werden können, einfache Werkzeuge für die Zuordnung von Daten. Das nennt man auch Daten-Mapping und kann mittels No-Code erfolgen. Man weist ganz einfach die Datenfelder des Fremdsystems den Datenfeldern der MPDV-Lösung zu. Sollte eine einfache Zuordnung nicht ausreichen oder beim Übernehmen der Daten eine Umrechnung nötig sein, so besteht die Möglichkeit, mittels Beschreibungssprache (Low-Code) die Schnittstelle zu modellieren. Zum Beispiel sind Umrechnungen oder logische Verknüpfungen mehrerer Datenfelder möglich.



Label & Report Designer

An vielen Stellen in der Smart Factory braucht es Reports und Etiketten – Ersteres zum Beispiel als Auswertung im definierten Layout und Letzteres zum Beispiel, um Material zu kennzeichnen. Beide Arten von Dokumenten benötigen ein Layout und Vorgaben, welche Informationen an welcher Stelle angezeigt werden sollen. Sollte eine der Vorlagen, die mit den MPDV-Lösungen ausgeliefert wird, nicht passen, so kann diese mit dem Label & Report Designer entweder angepasst oder komplett neu erstellt werden. Auch hier muss keineswegs programmiert werden. Der Designer ähnelt in seinem Aussehen und seinem Funktionsumfang einer Office-Applikation und ist daher intuitiv zu bedienen. An den Stellen, an denen Daten aus dem System erscheinen sollen, werden Platzhalter eingesetzt. Da es sich um einen grafischen Editor handelt und keinerlei Programmierung nötig ist, fällt dieses Gestaltungswerkzeug in die Kategorie No-Code.

Diese Vorlagen können genutzt werden, um automatische Reports zu generieren und zu verteilen. Wenn nötig, kann mittels Low-Code in den Erstellungsprozess eingegriffen werden.



Im Label & Report Designer können Daten und sonstige Elemente beliebig angeordnet werden. (Quelle: combit GmbH)

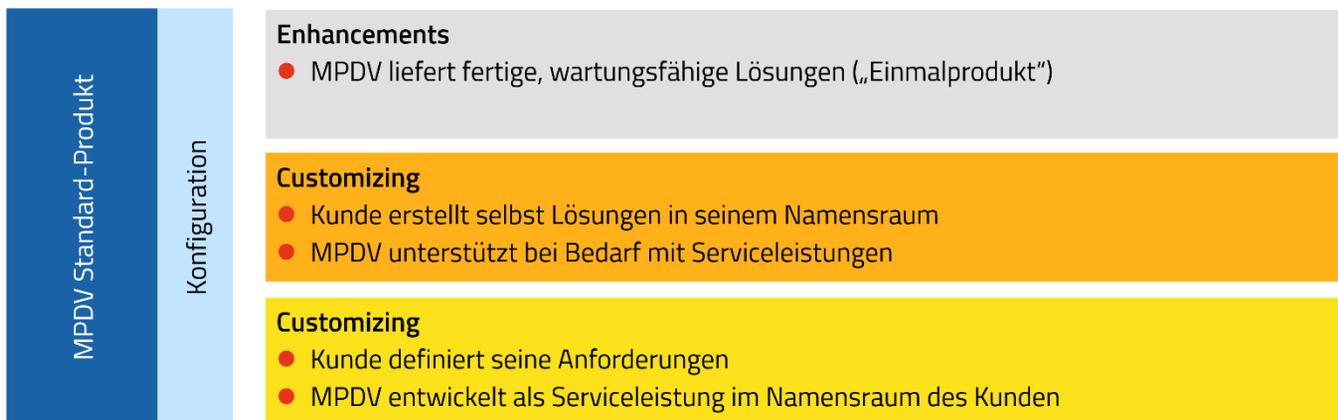
Alternative Lösungsansätze

Nicht immer haben Fertigungsunternehmen eigene Kapazitäten, um die verwendeten Applikationen anzupassen oder neue Anwendungen zu entwickeln. Insbesondere kleinere und mittelständische Betriebe konzentrieren sich lieber auf deren Kernkompetenz Produktion und vergeben jegliche IT-Aufgaben an externe Dienstleister. Daher bietet MPDV unterschiedliche Möglichkeiten an, solche Unternehmen zu unterstützen.

Serviceangebote von MPDV

MPDV nutzt die angebotenen Werkzeuge der Smart Factory Development Suite, um Kundenwünsche mit möglichst wenig Aufwand zu erfüllen. Außerdem bleibt so die durchgehende Interoperabilität gewährleistet. Entscheidet sich ein Kunde, die Entwicklungsleistungen von MPDV in Anspruch zu nehmen, so kann er dies in unterschiedlicher Form tun und dabei selbst steuern, wie viel Eigenleistung er einbringen möchte. Folgende drei Abstufungen sind möglich:

- MPDV liefert eine wartungsfähige Lösung.
- Der Kunde erstellt die Lösung in seinem eigenen Namensraum und lässt sich mehr oder weniger von MPDV dabei unterstützen und/oder beraten.
- Der Kunde definiert seine Anforderungen und MPDV entwickelt die Lösung im Namensraum des Kunden.

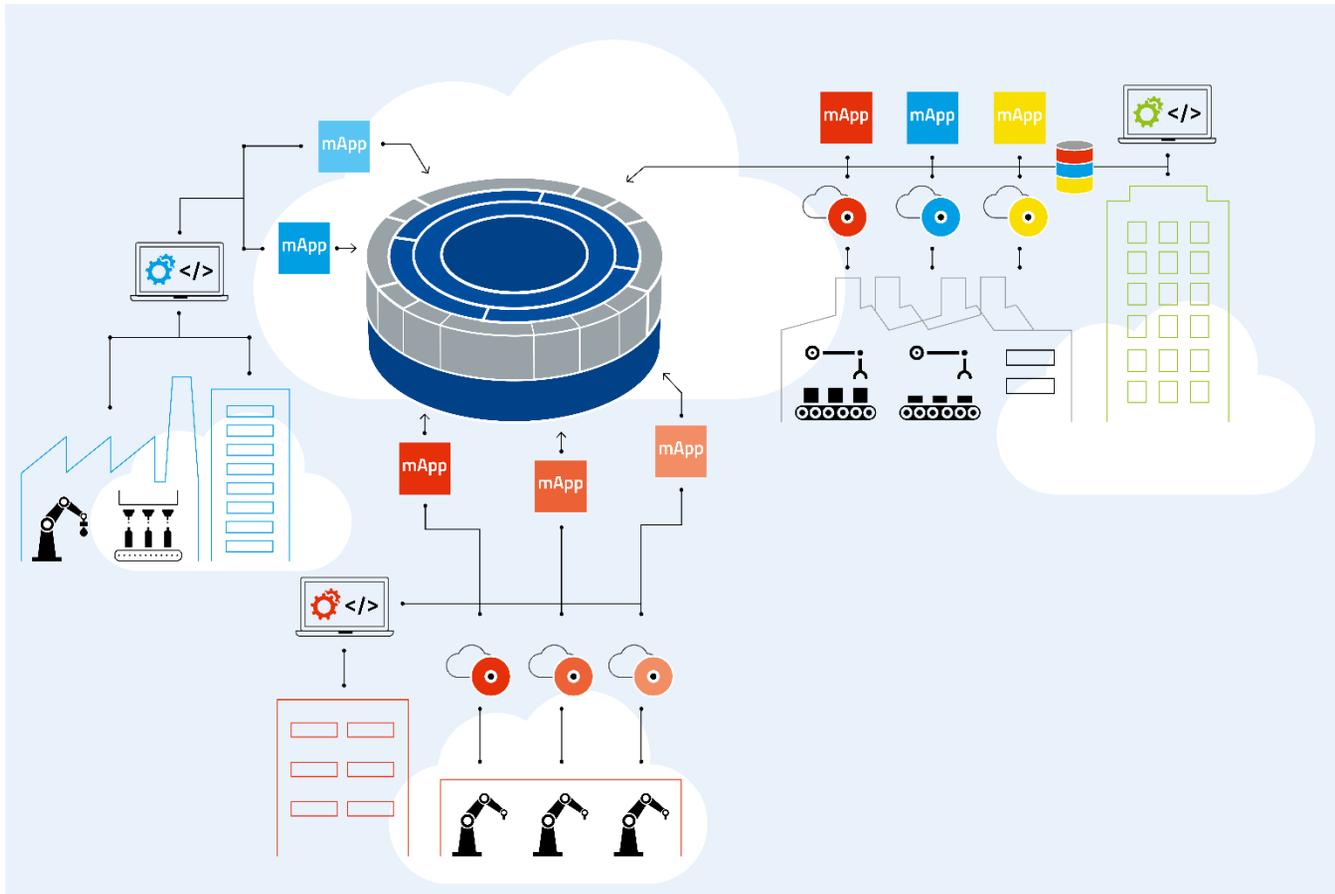


Bei MPDV entscheidet jeder Kunde selbst, wie viel er selbst machen möchte und wie viel MPDV unterstützen soll. (Quelle: MPDV)

Insbesondere die letzte Variante lässt sich beliebig individualisieren. Zum Beispiel bietet MPDV sowohl Unterstützung bei der eigentlichen Entwicklung – also beim Benutzen der Smart Factory Development Suite – als auch beim Konzipieren der Lösung. Eine langjährige Kompetenz und ein breites Domänenwissen zeichnet die Entwickler und Berater von MPDV aus, da sie sich einerseits schon seit langer Zeit mit solchen Herausforderungen beschäftigen und andererseits im regen Austausch mit den Standard-Entwicklern für die MPDV-Lösungen stehen.

Entwicklungsleistungen aus dem MIP-Ökosystem

Auch MIP-Partner unterstützen beim Entwickeln und Anpassen von mApps für die Smart Factory. Je nach eigener Kompetenz entwickeln einige MIP-Partner auch eigene mApps, die als Standard-Applikationen über den MIP Marketplace erworben werden können. Andere MIP-Partner sehen sich als reine Dienstleister und verweisen dabei auf die Entwicklungserfahrung ihrer Mitarbeitenden. Beide Arten von MIP-Partnern bieten Ihre Leistungen auf dem **MIP Marketplace** an: mip-marketplace.mpdv.com



Im MIP-Ökosystem treffen sich Anbieter, Anwender und Dienstleister. (Quelle: MPDV)

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:

Modellieren statt Programmieren

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

Über uns

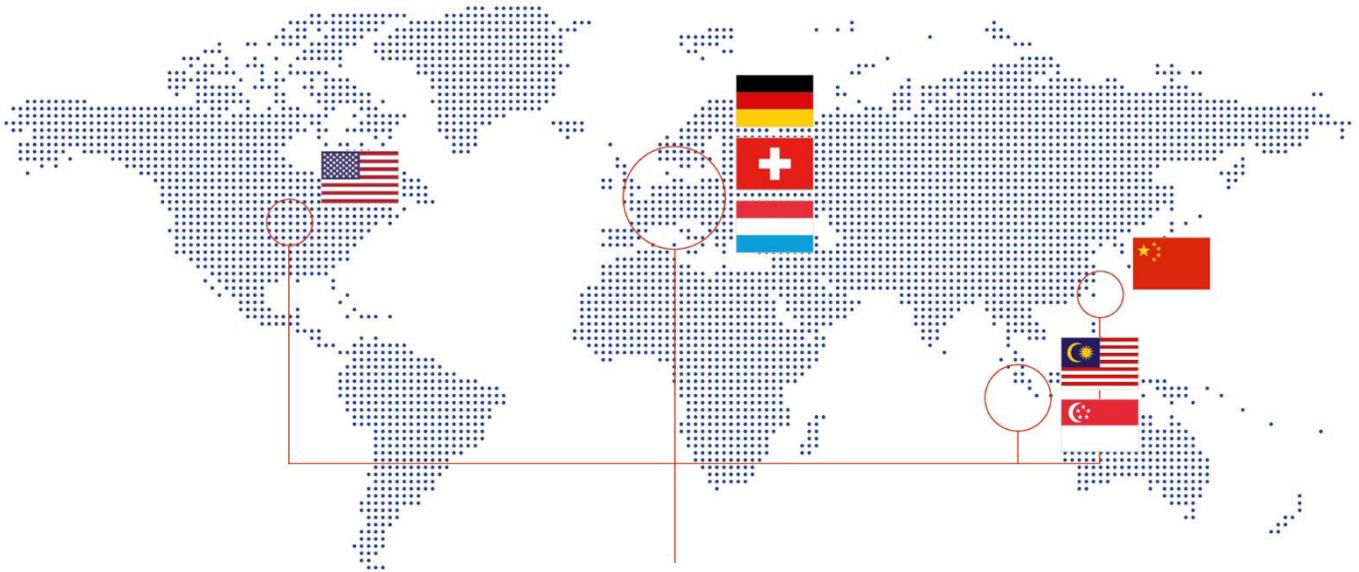


MPDV Mikrolab GmbH

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 · www.mpdv.com