



Stubs programmieren

Um nun noch die vertikale Kommunikation in einem realen, modernen Industriebetrieb abzubilden, brauchen wir in der ERP-Ebene drei Komponenten :

- Eine database, in der alle für die Fertigung nötigen Informationen abgelegt sind. Aufträge, Stücklisten, Lagerplätze usw. Das haben wir fertig. (Der weitaus größere Teil des ERP-Systems und der Datenbank sind natürlich die betriebswirtschaftlich geprägten Daten. Uns interessiert hier aber nur die Technik).
- Einen Zugriffsmechanismus, der über einen vernünftigen Protokollstandard (heute Webservice (REST), früher Corbra oder EDifact usw.) für die MES- oder Prozessebene in „Echtzeit“ (also schritthaltend mit der laufenden Fertigung) Auftragsdaten zugreifbar macht. Das bauen wir jetzt.
- Ein Datenformat, das, einem anerkannten Protokollstandard entsprechend, einfachen Zugang zu diesen Daten ermöglicht : JSON oder XML, im nächsten Paket.

Der Mechanismus soll bei uns immer als Testablauf durchgeführt werden. Das heißt, wie simulieren keine „endlos“ laufende Fertigung, sondern lassen mit jedem Lauf die Perlenkette immer wieder mit dem ersten Auftrag in der Auftragstable beginnen, und zählen die Auftrags-ID (primary key der auftrag-table) einfach hoch.

Wenn sie den Fertigungsauftrag für eine Modultätigkeit der Fabrik brauchen, müssen sie also aus MES der ERP-Ebene einen http-Request schicken (GET), der die Auftrags-ID beinhaltet.

Im PHP-Stub wird dann auf die Datenbank zugegriffen, und das zugehörige Produkt (in table **Auftrag**) gesucht.

Damit greift ihr PHP dann in die Stücklistentabelle (**Produkt**) und holt die drei benötigten Bauteile für das gefundene Produkt.

Damit müssen sie zum Schluß in die **Lager**-table, um auf diese Art zu ermitteln, welche Orders den zuständigen SPS gegeben werden müssen. Es kommen also zunächst 3 Zahlen zurück (später XML). Daraus decodieren sie in ihrem Python-MES-Programm die Order, die sie für den aktuellen Schritt brauchen, und beauftragen damit die SPS.