

- Ressourceneinsparung: Der gesamte Produktlebenszyklus wird überwacht. In der Planungsphase wird festgelegt, welche Teile davon für die sekundäre Nutzung verwendet werden [2].

Die Förderung von «Industrie 4.0» wird in den nächsten zehn bis fünfzehn Jahren radikale Veränderungen für die Unternehmen und die Wirtschaftsbranche mit sich bringen. Kurzfristig werden einige Personen von den Entlassungsmaßnahmen betroffen sein. Langfristig könnte sich jedoch die Arbeitsweise verschiedener Unternehmen im Hinblick auf die Ergebnisse ihrer Endprodukte deutlich verbessern.

Höhere Qualitätsstandards werden folglich zu einem Wandel in der Industriepraxis und zu einer höheren Verbrauchernachfrage führen. Das ist eine großartige Geschäftsstrategie für Hersteller nicht nur in Deutschland, sondern auch in anderen Ländern, um sich von überholten Praktiken zu befreien und in eine neue Epoche der Innovationen einzutreten.

1. . . . -  
« 4.0» // . - 2016. - 5 - . 11-

12.

2. Industrie 4.0 – Volkswirtschaftliches Potenzial für Deutschland. Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e. V., Das Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO. – Berlin, Stuttgart, 2014. – 46 S.

3. Industrie 4.0: Produktionsprozesse der Zukunft [Elektronische Ressource]. – Das Regime des Zugriffes : <https://up-pro.ru/library/strategi/opinion/industriya-4.0>. – Das Datum des Zugriffes : 15.04.2022.

## GRUNDLAGEN DER PROGRAMMIERSPRACHEN

Bis Mitte der 60er Jahre des 20. Jahrhunderts waren Computer zu teure Maschinen. Die Programmiersprachen dieser Ära wurden, wie auch die Computer, auf denen sie verwendet wurden, für spezifische Aufgaben wie wissenschaftli-

che Berechnungen entwickelt. Da die Maschinen teuer und langsam waren, war auch die Maschinenzeit teuer – daher stand die Geschwindigkeit der Programmausführung an erster Stelle.

Zu Beginn der Computerisierung war die maschinelle Sprache die einzige Sprache, die der Mensch bis dahin nicht erfunden hatte. Um Programmierer vor der harten Maschinenprogrammiersprache zu retten, wurden hochrangige Sprachen (d.h. Nicht-Maschinensprachen) erstellt. Neue Sprachen wurden zu einer Art Verbindungsbrücke zwischen Mensch und Computersprache.

Der Fortschritt der Computertechnologie hat den Prozess der Entstehung neuer verschiedener ikonischer Systeme für die Aufzeichnung von Algorithmen und die Entwicklung des Programmierprozesses als Ganzes bestimmt.

Der Begriff Programmieren bedeutet Prozess und die Kunst, Computerprogramme mit speziellen Programmiersprachen zu erstellen.

Im allgemeinen Sinne des Wortes ist die Programmierung eine Formalisierung eines vordefinierten Zustands, basierend auf einer Reaktion auf ein Ereignis, das durch die Mittel der Mathematik oder der Naturwissenschaften realisiert wird.

Im engeren Sinne wird die Programmierung als Codierung von Algorithmen in einer bestimmten Programmiersprache angesehen. Im weiteren Sinne ist Programmierung der Prozess der Erstellung von Programmen, d.h. die Entwicklung von Software.

Programmierung beinhaltet: Analyse, Design, Codieren und Kompilieren, Testen, Test- und Übergabeprogramme, Begleitung.

Verschiedene Programmiersprachen unterstützen verschiedene Programmierstile (sogenannte Programmierparadigmen). Zum Teil besteht die Kunst der Programmierung darin, eine der Sprachen auszuwählen, die am besten geeignet ist, um das vorhandene Problem zu lösen.

Die einzige Sprache, die direkt vom Prozessor ausgeführt wird, ist die Maschinensprache (auch als «Maschinencode» bezeichnet). Programmierer schreiben den Quellcode und der Computer (mit einem Compiler, Interpreter oder Assembler) überträgt ihn in einer oder mehreren Schritten, indem er alle Details verfeinert, in Maschinencode, der bereit ist, auf dem Zielprozessor ausgeführt zu werden. In einigen Sprachen wird jedoch ein interpretierbarer Binärcode der «virtuellen Maschine», auch als Byte-Code bezeichnet, anstelle von Maschinencode generiert. Dieser Ansatz wird in Forth, Lisp, Java verwendet.

Der materielle Teil des Programmerstellungsprozesses ist die Gesamtheit der elektrischen, elektronischen und mechanischen Komponenten automatisierter Systeme. Sie bildet technische Unterstützung (im Gegensatz zu den Softwaretools).

Die Grundlage einer bestimmten Programmiersprache ist eine leitende Idee, die einen wesentlichen Einfluss auf den Stil der jeweiligen Programme hat. Je

nach Zweck und/oder Art und Weise, wie Programme geschrieben werden, werden Programmierparadigmen(auch bekannt als Ansätze oder Technologien) unterschieden:

Die strukturelle Programmierung ist eine Programmiermethodik, die auf dem Systemansatz für die Analyse, das Design und die Implementierung von Software basiert.

Die objektorientierte Programmierung, deren Idee darin besteht, die Daten mit den Prozeduren zu verknüpfen, die diese Daten verarbeiten, zu einem Objekt. Die objektorientierte Programmierung basiert auf drei entscheidenden Prinzipien: Kapselung, Vererbung und Polymorphismus.

Die Anwendungsprogrammierung – Entwicklung und Debugging von Programmen für Endbenutzer, z. B. Buchhaltung, Textverarbeitung usw.

Die Systemprogrammierung – Entwicklung von allgemeinen Software-Tools, einschließlich Betriebssysteme, Hilfsprogramme, Systempakete, z. B. automatisierte Managementsysteme, Datenbankmanagementsysteme usw.

Die deklarative (logische, produktive) Programmierung ist eine Programmiermethode, die zur Lösung von Problemen künstlicher Intelligenz entwickelt wurde. Im angegebenen Kontext beschreibt das Programm die logische Struktur der Lösung des Problems und gibt hauptsächlich an, was getan werden muss, ohne ins Detail zu gehen, wie es gemacht wird. Es werden solche Programmiersprachen wie Prolog verwendet.

Die parallele Programmierung – Entwicklung von Programmen, die die gleichzeitige (parallele) Ausführung von Datenverarbeitungsvorgängen ermöglichen.

Die prozedurale (prozedural-orientierte) Programmierung ist eine Programmiermethode, nach der Programme als Listen von nacheinander ausgeführten Befehlen geschrieben werden.

Die funktionale Programmierung ist eine Programmiermethode, die auf der Aufteilung des Algorithmus zur Problemlösung in einzelne Funktionsmodule sowie zur Beschreibung ihrer Beziehungen und der Art der Interaktion basiert. Für die funktionale Programmierung werden die Sprachen NORE und ML am häufigsten verwendet. Elemente der funktionalen Programmierung werden auch von anderen Sprachen implementiert, z. B. C.

Zurzeit entwickelt sich die Richtung, die mit der Technologie der Bildung der Softwareprodukte verbunden ist, rasant. Dies ist auf den Übergang zur industriellen Produktionstechnologie von Programmen zurückzuführen, auf den Wunsch, Fristen, Arbeits- und Materialkosten für die Produktion und den Betrieb von Programmen zu reduzieren und ein garantiertes Qualitätsniveau zu gewährleisten.

Im Rahmen dieser Bereiche wurden folgende Softwaregruppen gebildet:

1. tools zum Erstellen von Anwendungen;

2. CASE-Technologie (Computer Aided System Engineering), die Methoden der Analyse, Planung und Erstellung von Software-Systemen entwickelt und für die Automatisierung der Prozesse der Entwicklung und Realisierung von Informationssystemen dient.

Der Hauptvorteil der CASE-Technologie ist die Unterstützung der Teamarbeit am Projekt durch die Möglichkeit, in einem lokalen Netzwerk von Entwicklern zu arbeiten, Teile des Projekts zu exportieren/ zu importieren, das Projekt zu organisieren.

Im Rahmen der CASE-Technologien wird das Projekt vollständig begleitet, nicht nur durch seine Programmcodes. Die in der CASE-Technologie vorbereiteten Projektmaterialien dienen den Programmierern als Aufgabe, und die Programmierung selbst wird eher auf die Codierung reduziert – die Übersetzung in eine bestimmte Sprache von Datenstrukturen und Verarbeitungsmethoden, wenn keine automatische Codierung vorgesehen ist.

Die meisten CASE-Technologien verwenden auch die „Prototyping“ - Methode, um Programme in den frühen Entwicklungsstadien schnell zu erstellen. Die Programmcodegenerierung erfolgt automatisch – bis zu 90% der Objektcodes und Texte in hochrangigen Sprachen, und die am häufigsten verwendeten Sprachen sind Ada, C, Cobol.

Heute gilt das weltweit führende CASE-System als Rational Rose der Rational Software Corporation. Das Rational Rose-System zielt darauf ab, Module mit der Unified Modeling Language zu erstellen. Die neueste Version des CASE-Systems dieses Unternehmens wird bereits für die Erstellung kommerzieller Software verwendet und unterstützt die gängigen Programmiersprachen Java, C ++, Teer, Ada, Visual Basic und Forte.

Mit diesen Technologien kann man interaktiv die Architektur der zu erstellenden Anwendung entwickeln, deren Quellcode generieren und parallel an der Dokumentation des zu erstellenden Systems arbeiten.

Die Erfindung einer Programmiersprache auf höchster Ebene ermöglichte, mit der Maschine zu kommunizieren. Wenn man wirklich auf das Wachstum und die Entwicklung der neuesten Technologien im Bereich der Programmierung achtet, kann man davon ausgehen, dass in naher Zukunft menschliche Erkenntnisse in diesem Bereich dazu beitragen werden, Sprachen zu erzeugen, die Informationen in Form von Gedanken, Worten, Ton oder Gesten empfangen, verarbeiten und übertragen können. Solche Sprachen können zu Recht den Namen der «Höchsten» tragen.

1. Eine kleine Geschichte der Programmiersprachen [Elektronische Ressource]. – Das Regime des Zugriffes : <https://www.computerwoche.de/a/eine-kleine>

