

dr inż. Jerzy Zając  
mgr inż. Grzegorz Chwajot  
mgr inż. Łukasz Ślusarczyk  
Politechnika Krakowska

## WYBRANE PROBLEMY SAMOKREOWANIA SIĘ ROZPROSZONYCH SYSTEMÓW STEROWANIA WYTWARZANIEM

*W pracy omówiono etapy procesu samokreowania się rozproszonego systemu sterowania wytwarzaniem. System ten zbudowany jest z jednolitych, konfigurowalnych, autonomicznych i kooperatywnych agentów wytwórczych. Proces samokreowania się systemu realizowany jest poprzez wymianę komunikatów XML-owych.*

### SELECTED PROBLEMS OF SELF-CREATION OF DISTRIBUTED MANUFACTURING CONTROL SYSTEMS

*The paper describes steps of self-creation of distributed manufacturing control system which is based on the integration of unified, easy to re-configure, autonomous and cooperative manufacturing agents. The process of self-creation is performed through exchange of XML messages.*

## 1. WPROWADZENIE

Jedną z podstawowych cech współczesnego środowiska wytwarzania jest jego rozproszenie. Zjawisko to dotyczy zarówno skali makro, np. wirtualnych przedsiębiorstw zintegrowanych informacyjnie poprzez zastosowanie technologii internetowych, jak i skali mikro, w której klasyczne, scentralizowane systemy sterowania zautomatyzowanym wytwarzaniem zastępowane są przez systemy zdecentralizowane, zbudowane z autonomicznych modułów charakteryzujących się zdolnością do współdziałania oraz możliwością samodzielnego podejmowania decyzji. Zjawisko rozproszoneści w systemach wytwarzania jest obecnie traktowane jako jeden z najistotniejszych czynników umożliwiających nadanie nowego impulsu rozwojowego tym systemom.

Problematyka poszukiwania nowych koncepcji systemów wytwarzania wykorzystujących zdecentralizowane systemy sterowania stała się przedmiotem intensywnego zainteresowania już w latach dziewięćdziesiątych minionego stulecia. Przejawiało się to między innymi w przekazywaniu znacznych środków finansowych na programy badawcze o zasięgu światowym. Najbardziej znanym przykładem jest tu dziesięcioletni międzynarodowy program dotyczący opracowania i wdrożenia koncepcji Inteligentnych Systemów Wytwórczych IMS (ang. *Intelligent Manufacturing Systems*) [9]. Jest to najprawdopodobniej największy i najbardziej kosztowny (budżet 1 mld USD) program w historii dotyczący systemów wytwarzania. Przejście od koncepcji do implementacji wymagało bowiem wielu lat badań i dopiero obecnie pojawiają się w środowisku przemysłowym pierwsze pilotowe implementacje opracowanych koncepcji [12][13].

Rozpatrując zagadnienie automatyzacji dyskretnych procesów przemysłowych należy zwrócić uwagę nie tylko na kwestię rozwoju zautomatyzowanych maszyn wytwórczych [8], ale również na kwestię wymiany informacji pomiędzy elementami tworzącymi system zautomatyzowany [15]. Wymiana informacji we współczesnych systemach zautomatyzowanych napotyka bowiem na barierę wynikającą z różnorodności stosowanych rozwiązań sprzętowych i programowych. Z brakiem jednolitego standardu mamy do czynienia w zakresie technologii obiektów rozproszonych [3] stosowanych szeroko w zdecentralizowanych systemach sterowania. Na rynku rywalizują bowiem technologie CORBA [11], DCOM [5] i Enterprise Java Beans [6]. Podobna sytuacja występuje w zakresie komunikacji pomiędzy urządzeniami automatyki przemysłowej. Choć istnieje kilka powszechnie akceptowanych standardów [10], takich jak: MAP (ang. *Manufacturing Automation Protocol*) czy MMS (ang. *Manufacturing Message Specification*), które wspierane są przez liczące się standardy sieci miejscowych tzn. Profibus, InterBus-S czy LONWorks, to jednak brak jest jednolitego, akceptowanego przez wszystkich standardu *de facto*, jakim w sieciach komputerowych jest obecnie TCP/IP. Poszukiwanie szeroko akceptowanego standardu dotyczy również samych układów sterowania urządzeń wytwórczych. Chodzi tu o opracowanie niezależnych sprzętowo, otwartych układów sterowania numerycznego urządzeń wytwórczych [14] obejmujących funkcjonalnością układy NC (ang. *Numerical Control*), RC (ang. *Robot Control*), PLC (ang. *Programmable Logic Control*) i CC (ang. *Cell Control*). Prace nad standaryzacją tych układów realizowane były między innymi w ramach inicjatyw: OSAKA w Europie, JOB w Japonii czy też OMAC w USA. Jednym z efektów prac nad standaryzacją układów sterowania urządzeń wytwórczych jest powstanie układów sterowania wykorzystujących komputery przemysłowe. W tym przypadku sterowanie realizowane jest albo wyłącznie w obszarze oprogramowania, tak jak dzieje się to w przypadku oprogramowania typu Soft PLC lub też stanowi połączenie sprzętu (np. karta typu PCI) i oprogramowania. Nowoczesne układy sterowania wykorzystujące komputery przemysłowe typu PC nie wymagają już nawet systemów operacyjnych czasu rzeczywistego i mogą działać pod nadzorem standardowych systemów operacyjnych (np. Windows NT)[1]. Oznacza to ich większą otwartość w stosunku do klasycznych układów sprzętowych oraz łatwiejszą integrację z pozostałymi elementami tworzącymi system sterowania dzięki możliwości wykorzystania szeroko akceptowanych standardów znanych ze środowiska komputerów PC.

Proces sterowania wytwarzaniem realizowany jest poprzez przetwarzanie w czasie rzeczywistym jakiejś formy modelu systemu wytwarzania zapisanej w kodzie komputerowym. W przypadku sterowania rozproszonego należy przyjąć, że każdy z elementów składowych systemu wyposażony jest w prosty algorytm działania (współdziałania), a budowa modelu całego systemu realizowana jest poprzez samokreowanie. Opis działań niezbędnych do zbudowania rozproszonego systemu sterowania wytwarzaniem przedstawiono w pracy [4]. W kolejnych dwóch rozdziałach niniejszej pracy zostaną omówione zagadnienia, które zdaniem autorów mogą stanowić istotne uzupełnienie przedstawionej tam problematyki. Pierwsze z tych zagadnień dotyczy etapów procesu samokreowania się systemu sterowania wytwarzaniem. Zwrócono uwagę na ważne dla tego procesu kwestie występowania sytuacji nadzwyczajnych, takich jak np. awarie wynikające z niedoskonałości wykorzystywanych rozwiązań informatycznych. Drugie zagadnienie dotyczy natomiast możliwości wykorzystania języka XML jako środka przekazywania informacji w

systemie rozproszonego sterowania zautomatyzowanym wytwarzaniem. Ma to szczególnie istotne znaczenie w charakterystycznym dla współczesnego wytwarzania heterogenicznym środowisku informatycznym.

## 2. ETAPY SAMOKREOWANIA SIĘ ROZPROSZONEGO SYSTEMU STEROWANIA WYTWARZANIEM

Rozpatrywany system sterowania wytwarzaniem zbudowany jest z dwóch typów agentów: *agentów wytwórczych* oraz *agentów dostosowujących*. Zasadniczą rolę w proponowanej koncepcji systemu sterowania wytwarzaniem odgrywają uniwersalne, konfigurowalne agenty wytwórcze. Ich inteligencja związana jest bezpośrednio ze zbiorem wykonywanych w systemie czynności elementarnych oraz zdolnością do realizacji wspólnych celów przy udziale innych agentów. W celu zapewnienia współpracy pomiędzy agentem wytwórczym a sterownikiem urządzenia wytwórczego niezbędne jest wprowadzenie do systemu sterowania wytwarzaniem modułu programowego zwanego *agentem dostosowującym*. Agent taki musi być przygotowany indywidualnie dla każdego urządzenia wytwórczego tworzącego system wytwarzania.

Działanie systemu sterowania wytwarzaniem opiera się na założeniu, że każdy agent wytwórczy jest aktywny i dąży do wykonania przyporządkowanych mu czynności elementarnych. Rozpoczęcie realizacji większości z czynności wymaga zazwyczaj współpracy kilku agentów wytwórczych. Tak więc, każdy agent - analizując możliwość realizacji swojej czynności elementarnej - uzyskać musi akceptację współdziałania przy jej realizacji od wszystkich agentów wytwórczych niezbędnych do jej wykonania.

Jedną z ważnych kwestii dotyczących rozproszonego systemu sterowania wytwarzaniem jest sposób uaktywniania agentów wytwórczych. Chodzi w tym przypadku o zapewnienie podejmującym decyzje agentom ograniczonej czasowo wyłączności na współdziałanie z niezbędnymi dla nich agentami wytwórczymi, nie realizującymi aktualnie żadnych czynności. W systemie sterowania nie może dojść do sytuacji, w której procesy decyzyjne „podbierać” będą sobie wzajemnie agenty niezbędne dla realizacji ich czynności elementarnych. Prowadziłoby to bowiem do zablokowania działania systemu. Jednym ze sposobów zapewnienia wzajemnego wyłączenia w rozproszonym systemie sterowania wytwarzaniem jest takie przekazywanie agentom wytwórczym uprawnień decyzyjnych, aby w danej chwili tylko jeden z nich dysponował tymi uprawnieniami. Zastosować można w tym celu algorytm pierścienia z żetonem. Jego działanie polega na zbudowaniu logicznego pierścienia z agentów wytwórczych oraz wprowadzeniu do tego pierścienia żetonu (znacznika), który w nim krąży - przemieszczając się kolejno pomiędzy agentami wytwórczymi. Decyzje podejmować może wyłącznie ten agent wytwórczy, który posiada znacznik. Po podjęciu decyzji znacznik przekazywany jest do następnego agenta. Jeżeli agent, który posiada znacznik, nie może w danym momencie podejmować decyzji, bo bierze udział w realizacji czynności, to natychmiast przekazuje go do kolejnego agenta wytwórczego.

Aby umożliwić zastosowanie algorytmu pierścienia z żetonem, niezbędne jest wprowadzenie do systemu sterowania elementu odpowiedzialnego za stworzenie logicznego pierścienia złożonego z agentów wytwórczych, wprowadzenie do niego znacznika oraz zapewnienie poprawności działania takiego układu. Elementem odpowiedzialnym za realizację tego działania jest w rozpatrywanym systemie sterowania wybrany agent wytwórczy. Przyjęto bowiem, że każdy agent wytwórczy

pełnić może w systemie sterowania również rolę koordynacyjną i kontrolną występując jako tzw. agent koordynujący (*broker*). Aby jednak zapewnić poprawność działania takiego systemu, występować w nim może wyłącznie jeden broker.

Pierwszym zasadniczym etapem w procesie samokreowania się rozproszonego systemu sterowania wytwarzaniem jest wybór agenta wytwórczego, który pełnił będzie rolę brokera. Na tym etapie uruchamiania systemu zaproponować można różne sposoby wyboru brokera. Bez względu na to, jaki sposób przyjmiemy, ważne jest aby agent wytwórczy „podejmujący się tej roli” był pierwszym agentem wytwórczym uruchomionym w systemie. Po uruchomieniu przechodzi on bowiem w stan nasłuchiwania i analizowania komunikatów przekazywanych przez uruchamiane następne agenty wytwórcze. Pierwszym zadaniem agenta wytwórczego, który stał się brokerem, jest jednak uruchomienie w systemie operacyjnym swojej kopii i przekazanie jej wszystkich danych pierwotnej konfiguracji agenta wytwórczego. Tę kwestię można również interpretować jako uruchomienie przez pierwszego w systemie agenta wytwórczego swojej kopii i nadanie mu funkcjonalności brokera. Oznacza to, że po uruchomieniu pierwszego agenta wytwórczego pojawi się w systemie sterowania dodatkowy agent realizujący cele koordynacyjne czyli broker. Aby umożliwić pozostałym agentom jednoznaczne zidentyfikowanie brokera zastosować można np. tablicę ogłoszeń (ang. *blackboard*).

Drugim zasadniczym etapem w procesie samokreowania się rozproszonego systemu sterowania wytwarzaniem jest osiągnięcie przez system sterowania „masy krytycznej”, tzn. gotowości do realizacji założonych procesów wytwórczych. Za realizację tej kwestii odpowiedzialny jest broker, który w momencie osiągnięcia niezbędnej gotowości systemu informuje o tym fakcie pozostałe agenty wytwórcze.

W kolejnym etapie agenty tworzące rozproszony system sterowania są szeregowane przez brokera, co umożliwi zbudowanie logicznego pierścienia ustalającego kolejność przekazywania znacznika. W systemie występują dwa znaczniki: kontrolny oraz systemowy. Znacznik kontrolny wykorzystywany jest przez brokera do sprawdzania dostępności agentów wytwórczych oraz ich gotowości do realizacji założonych zadań. Znacznik systemowy przekazywany kolejno pomiędzy agentami tworzącymi logiczny pierścień pozwala natomiast na realizację wzajemnego wyłączenia agentów wytwórczych w procesie decyzyjnym.

Ostatnim etapem samokreowania się rozproszonego systemu sterowania wytwarzaniem jest etap pracy, w którym agenty wytwórcze realizują założone zadania.

Omówione powyżej etapy samokreowania się rozproszonego systemu sterowania wytwarzaniem nie uwzględniają występujących w realnych warunkach awarii związanych z niedoskonałością połączeń sieciowych, niesprawnością sprzętu komputerowego czy też błędami w oprogramowaniu. Projektując taki system należy więc przewidzieć typowe sytuacje awaryjne, aby system mógł sam sobie z nimi poradzić, o ile tylko jest to możliwe. Rozpatrując stan pracy systemu, rozważyć można następujące typowe sytuacje nadzwyczajne:

- awarię agenta wytwórczego,
- awarię brokera,
- utratę znacznika.

Awaria agenta wytwórczego wymaga usunięcia uszkodzonego agenta z logicznego pierścienia oraz „przenumerowania” agentów, aby odtworzyć logiczny pierścień. Elementem odpowiedzialnym za zrealizowanie tego zadania jest broker.

Awaria brokera uniemożliwia normalną pracę systemu sterowania. Niezbędne jest więc dokonanie elekcji innego agenta wytwórczego na funkcję brokera oraz ponowne przeprowadzenie całej procedury samokreowania systemu, przy czym agenty wytwórcze muszą jako swój stan początkowy przyjąć stan sprzed awarii.

Za zdiagnozowanie i usunięcie awarii polegającej na utracie znacznika odpowiedzialny jest broker. Jego zadaniem jest również wprowadzenie nowego znacznika do systemu. Proces ten musi być jednak realizowany w taki sposób, aby nie było możliwe pojawienie się dwóch znaczników tego samego typu w logicznym pierścieniu.

Jako sytuację nadzwyczajną w procesie samokreowania się systemu sterowania wytwarzaniem uznać należy także kwestię wprowadzania do pracującego systemu nowych agentów wytwórczych. Mogą to być również „naprawione” po usuniętej awarii agenty. Za realizację tego zadania odpowiedzialny jest broker, który w czasie pracy systemu cały czas znajduje się w trybie nasłuchiwania zgłoszeń pochodzących od nowych agentów wytwórczych.

### **3. XML JAKO NARZĘDZIE WYMIANY INFORMACJI W ROZPROSZONYM SYSTEMIE STEROWANIA WYTWARZANIEM**

Tekst jest uniwersalnym środkiem przekazywania informacji, niezależnym od platformy systemowej czy sprzętowej. Podstawową cechą języka XML [7] jest właśnie wykorzystanie tekstu jako nośnika informacji. XML jest metajęzykiem opartym o znaczniki. Reguły XML-a nie określają z góry ani nazw znaczników, ani też ich znaczenia, a jedynie formułują zestaw zasad dotyczących sposobu formatowania danych (znaczniki otwierające i zamykające, zagnieżdżanie elementów itp.).

Stosowane nazewnictwo jest zazwyczaj związane z charakterem rozpatrywanego zagadnienia i zależy od przyjętych standardów. Dane przekazywane są w postaci atrybutów określonego znacznika oraz pomiędzy znacznikami otwierającym i zamykającym. Zagnieżdżanie elementów wewnątrz pakietu XML-owego pozwala na odwzorowanie strukturalnej budowy przesyłanej informacji.

Wykorzystanie technologii XML w systemie rozproszonego sterowania wytwarzaniem realizowane jest w ramach dwóch zasadniczych etapów. W pierwszym etapie przyjmuje się określony zestaw nazw (pojęć) definiujących dane konieczne do realizacji procesu rozproszonego sterowania wytwarzaniem oraz tworzy się strukturę tych danych w formie wymaganej przez XML. W drugim etapie realizowana jest natomiast implementacja przyjętego rozwiązania na wymaganej platformie systemowej przy użyciu określonych narzędzi programistycznych. XML pozwala projektantowi w tym względzie na dużą elastyczność postępowania.

W procesie samokreowania się wieloagentowego systemu sterowania wytwarzaniem niezbędne jest wykonanie określonych działań, które opisane zostały szerzej w rozdziale drugim. Wykonanie każdego z tych działań wiąże się z koniecznością wymiany komunikatów pomiędzy agentami tworzącymi rozproszony system sterowania. Poniżej przedstawiono przykład pakietu XML wysyłanego przez nowo uruchomionego (wprowadzonego do systemu) agenta wytwórczego do brokera w celu

zarejestrowania się w systemie. Przykład ten pokazuje kolejną istotną cechę języka, jaką jest samoopisowość. Krótka analiza treści pozwala nawet bez szczegółowej znajomości przyjętego protokołu wymiany informacji zorientować się, iż nadawcą jest agent o nazwie *Tokarka\_TKX*, adresatem *broker*, a celem przesyłanej informacji jest rejestracja agenta.

```
<root>
  <header>
    <source srcID="Tokarka_TKX"/>
    <destination destID="broker"/>
  </header>
  <body>
    <message type="rejestracja"/>
  </body>
</root>
```

W odpowiedzi na powyższy komunikat broker automatycznie wysyła agentowi zestaw parametrów umożliwiających konfigurację agenta w systemie.

Przystępując do implementacji XML-a w procesie rozproszonego sterowania opracować należy najpierw wspólny dla wszystkich elementów systemu protokół czyli zestaw pakietów XML-owych wykorzystywanych do przesyłania informacji w systemie. Pakiety te muszą być interpretowane jednoznacznie przez współdziałające w systemie agenty. Dotyczy to zarówno komunikatów wykorzystywanych wyłącznie do *przekazywania danych* jak i komunikatów reprezentujących *zapytania* oraz *odpowiedzi* na te pytania. Jako przykład *zapytania* w systemie rozproszonego sterowania wytwarzaniem można przedstawić np. żądanie podania informacji na temat stanu agenta, jego dostępności oraz oferowanej przez niego funkcjonalności. Uzyskanie odpowiedzi na takie pytanie umożliwia pytającemu agentowi zorientowanie się w strukturze systemu oraz określenie możliwych do wykonania akcji w powiązaniu z dostępnymi i potrzebnymi do jej wykonania elementami systemu. Istnieje także możliwość wykorzystania XML-a do zapisu danych o przebiegu pracy poszczególnych elementów systemu oraz zautomatyzowane przedstawianie tychże danych w Intranecie/Internecie w celu monitoringu działania systemu, analizy jego pracy a także przeprowadzania niezbędnych statystyk. Te ostatnie zastosowania są pomocnicze i drugorzędne w omawianym zagadnieniu, ukazują jednak rozległość i uniwersalność technologii XML. Innym sposobem wykorzystania XML-owej formy komunikacji są zdalne wywołania procedur, które można zastosować w procesie konfiguracji rozproszonego systemu sterowania wytwarzaniem. Przykładami istniejących i dynamicznie rozwijanych systemów zdalnego wywoływania procedur przy użyciu XML-a są projekty XML-RPC czy działający w oparciu o podobne reguły a oferujący większą funkcjonalność SOAP, którego przykład wykorzystania w procesie sterowania wytwarzaniem znaleźć można w pracy [2].

Podstawowym atutem zastosowania omawianej technologii w zagadnieniach rozproszonego sterowania wytwarzaniem jest jednak możliwość wykorzystania jednego wspólnego „tekstowego” protokołu wymiany informacji między elementami systemu sterowania obejmującego wiele platform sprzętowych i systemowych. W zasadzie w każdym środowisku możliwa jest implementacja mniej lub bardziej skomplikowanych narzędzi do analizy i przetwarzania informacji w postaci tekstowej. Możliwe jest zatem

przetwarzanie informacji przekazywanych w formacie XML. Oczywiście najczęściej nie będzie potrzeby tworzenia „od zera” niezbędnych narzędzi, bowiem w chwili obecnej na rynku istnieje wiele dedykowanych aplikacji i bibliotek. Są to rozwiązania zarówno komercyjne jak i dystrybuowane na zasadach *public domain* lub podobnych. Mowa tu o popularnie zwanych parserach XML. Ich zadaniem jest, najogólniej rzecz ujmując, pobranie pakietu XML i zwrócenie użytkownikowi konkretnych, zawartych w tym pakiecie danych. Parsery takie dostępne są na platformy Windows, Unix/Linux, Mac w formie bibliotek w takich językach jak np. C/C++, Java, czy też w językach skryptowych: Perl, Python i innych. Przykładowe parsery to: Expat XML Parser (C), Xerces (Java, C++).

Istotną cechą omawianej technologii XML jest fakt, iż jest to otwarty standard nadzorowany przez niekomercyjną organizację (W3C), wspierającą rozwój technologii internetowych. Nie ma więc zasadniczego niebezpieczeństwa braku wsparcia rozwoju kolejnych wersji specyfikacji, czy też np. wprowadzenia odpłatności za jej wykorzystanie.

Tekstowy format przesyłania danych niesie za sobą obok bezsprzecznych zalet także pewne niedogodności. Jedną z nich jest zwiększona objętość przesyłanych informacji, która w porównaniu do komunikacji opartej o tryb binarny może być nawet kilkakrotnie większa. Druga kwestia dotyczy narzutu informacji dodatkowych (nazwy znaczników, atrybutów). Obie te niedogodności mogą mieć jednak większe znaczenie w przypadku przekazywania bardzo dużych zbiorów danych zapisanych w formacie XML-a, w których zwiększenie objętości o kilkadziesiąt procent w porównaniu do tekstu pozbawionego XML-owego narzutu czy też nawet kilkakrotne w odniesieniu do formy binarnej może być czynnikiem dyskwalifikującym. Pakiety informacji przekazywanych w rozproszonych systemach sterowania wytwarzaniem są zazwyczaj niewielkie, co przy mocy obliczeniowej współczesnych komputerów oraz przepustowości stosowanych aktualnie łącz sieciowych pozwala skupiać się przede wszystkim na zaletach wykorzystania technologii XML.

#### 4. PODSUMOWANIE

Intensywne i prowadzone na szeroką skalę prace dotyczące systemów wytwarzania nowej generacji zwiększyły zainteresowanie problematyką rozproszonego sterowania zautomatyzowanym wytwarzaniem. Jednym z istotnych aspektów budowy systemu sterowania wytwarzaniem jest jego samokreowanie. W pracy omówiono etapy procesu samokreowania się rozproszonego, wieloagentowego systemu sterowania wytwarzaniem zbudowanego z jednolitych i konfigurowalnych agentów wytwórczych. W trakcie procesu samokreowania się systemu sterowania wystąpić mogą różne sytuacje nadzwyczajne takie jak awarie elementów tworzących system czy też wprowadzanie nowych elementów do systemu. Te kwestie muszą być więc uwzględnione w trakcie opracowywania samokreującego się systemu sterowania. Proces samokreowania się systemu sterowania realizowany jest poprzez wymianę komunikatów pomiędzy elementami tworzącymi ten system. We współczesnym heterogenicznym środowisku informatycznym szczególnie ważne jest wykorzystanie akceptowalnych przez wszystkich standardów jakimi są TCP/IP i XML. Zdaniem autorów przyjęcie tych standardów otwiera nowe możliwości wprowadzania standaryzacji w systemach zarządzania i sterowania wytwarzaniem.

## LITERATURA

- [1] *ACRAMATIC 2100 Web Site*. <http://www.sea.siemens.com/machine/product/mcA2100.html>, 2002.
- [2] Baumann C., Kittl B.: *Cell Control Software and Web Services*. Proc. of 13<sup>th</sup> International DAAAM Symposium: "Intelligent Manufacturing & Automation: Learning from Nature", Vienna 2002, s. 027-028.
- [3] *Component Focus Web Site*. <http://www.middleware.net/components/articles/index.html>, 2002.
- [4] Cyklis J., Zając J., Czmich W., Słota A. i Zych J.: *Jak zbudować rozproszony system sterowania wytwarzaniem - od koncepcji do implementacji*. Materiały z konferencji AUTOMATION'2000. PIAP, Warszawa 2000, s.79-84.
- [5] *DCOM Technical Overview. White Paper*. Microsoft Corporation, <http://www.microsoft.com/cominfo>, 1996.
- [6] *Enterprise Java Beans Specification Web Site*. <http://java.sun.com/products/docs.html>, 2002.
- [7] *Extensible Markup Language (XML)*. <http://www.w3c.org/XML>, 2002.
- [8] Honczarenko J.: *Elastyczna automatyzacja wytwarzania*. WNT 2000.
- [9] *IMS, Intelligent Manufacturing Systems Web Site*, <http://www.ims.org>, 1994.
- [10] Koch J., Kosmulska-Bochenek E., Kowalczyk G., Mrzygłód M., Nowak M., Reiner J., Sacha K., Skura K., Smalec Z., Sochan A., Trzciniński P.: *Systemy komunikacyjne w automatyzacji procesów wytwarzania*. Prace Naukowe Instytutu Technologii Maszyn i Automatyzacji Politechniki Wrocławskiej 67, Seria: Konferencje 29, Wrocław 1997, s. 385-416.
- [11] Pancerella C.M., Whiteside R.A.: *Using CORBA to Integrate Manufacturing Cells to a Virtual Enterprise*. Proc. of Integrated Manufacturing - Plug and Play Software for Agile Manufacturing, Boston 1996.
- [12] Parunak H. V. D.: *A Practitioners' Review of Industrial Agent Applications*. Autonomous Agents and Multi-Agent Systems 3:4, 2000, s. 389-407.
- [13] Schaeffer C., Sieverding J.: *Holonic Production and Material Flow in Industry*. HMS Symposium Kitakyushu Japan, 2000, [http://www.yelab.com/hms16open/Executive\\_Summary\\_C\\_Schaeffer.pdf](http://www.yelab.com/hms16open/Executive_Summary_C_Schaeffer.pdf).
- [14] Szafarczyk M.: *Open Architecture Controllers and Automatic Supervision in Manufacturing*. Manufacturing Systems, Vol.25, No.1, 1996, s. 37-41.
- [15] Zając J.: *Wybrane problemy komunikacji i wizualizacji w sterowaniu rozproszonymi systemami wytwarzania*. Pomiar Automatyka Robotyka 6/99. PIAP, Warszawa 1999, s.46-50.