

Henryk Juszka, Marcin Tomasik, Stanisław Lis  
Katedra Energetyki Rolniczej  
Akademia Rolnicza w Krakowie

## WIZUALIZACJA KOMPUTEROWA NARZĘDZIEM WSPOMAGAJĄCYM MODELOWANIE PROCESÓW ROLNICZYCH

### Streszczenie

Przedstawiono możliwość zastosowania wizualizacji komputerowej do przedstawiania wyników modelowania matematycznego. Wizualizacja polega na zobrazowaniu wyników symulacji komputerowej modelu matematycznego w postaci animowanych obiektów tekstowych i graficznych, których spójną całość komponuje się za pośrednictwem akcji sterujących ich wyświetlaniem i usuwaniem. Mechanizm jej tworzenia przedstawiono dla komputerowej symulacji realizowanej w programie Matlab®. Komunikacja pomiędzy programami: realizującym symulację oraz odpowiedzialnym za wizualizację, odbywa się za pośrednictwem protokołu DDE.

**Słowa kluczowe:** wizualizacja komputerowa, DDE, skrypty, trendy

### Wstęp

Wizualizacja komputerowa stanu procesu jest zobrazowaniem w postaci animowanych obiektów tekstowych i graficznych zmian w nim zachodzących, których spójną całość komponuje się za pośrednictwem akcji sterujących ich wyświetlaniem, usuwaniem itp. Coraz częściej istnieje potrzeba obserwacji procesu, a także kontaktu z aparaturą kontrolno-pomiarową. Temu właśnie służy wizualizacja komputerowa. Oprogramowanie umożliwiające przeprowadzenie wizualizacji oraz kontroli procesu technologicznego nazywane jest systemem SCADA (Supervisory Control and Data Acquisition) [Wonderware Corporation 2005]. System ten służy do zbierania danych ze sterowanego procesu i przesyłania ich do centralnego komputera, gdzie są wykorzystywane do zarządzania i sterowania w czasie rzeczywistym.

Zastosowanie wizualizacji komputerowej do modelowania matematycznego, pozwala na czytelne przedstawienie wyników modelowania oraz wygodną kontrolę symulacji modelu.

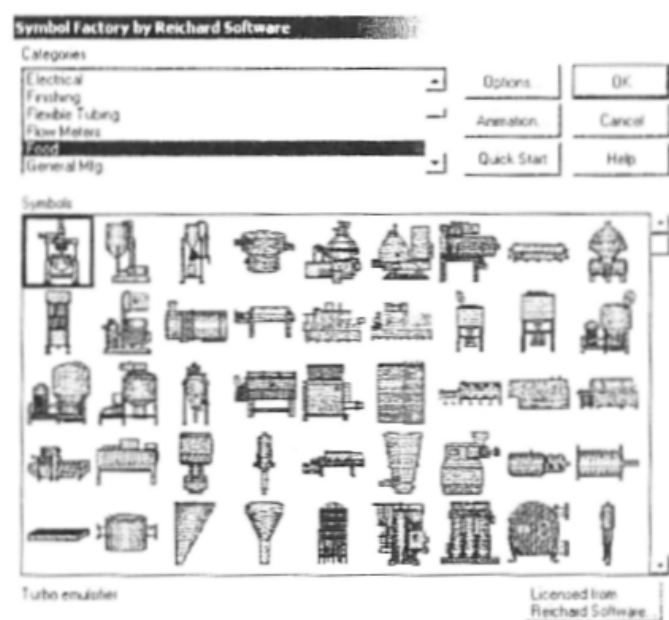
Celem pracy jest przedstawienie możliwości graficznego przedstawiania wyników symulacji komputerowej modelu na przykładzie procesu przepływowej pasteryzacji mleka.

Jako narzędzia do opracowania wizualizacji w/w procesie użyto programu InTouch firmy Wonderware® [Wonderware Corporation 2005]. Model procesu pasteryzacji przepływowej został przygotowany w programie Matlab®. W programie tym realizowana jest również symulacja komputerowa.

## Wizualizacja pracy pasteryzatora przepływowego

Aplikacja InTouch przedstawia proces przepływowej pasteryzacji mleka. W przypadku przekroczenia wartości krytycznych, uruchamia alarm i odpowiednie procedury uniemożliwiające wyprodukowanie wadliwego produktu. Przejmuje archiwizację danych procesowych i ich analizę.

Budowę aplikacji zrealizowano w programie WindowMaker. Polega ona na umieszczeniu wybranych obiektów w zadeklarowanym oknie ekranu i określeniu parametrów, które decydują dynamicznie o wyglądzie obiektów. Znajdują się w nim narzędzia do tworzenia obiektów (proste elementy graficzne, bitmapy itp.). Dodatkowo w bibliotece SymbolFactory – zawarto tysiące zaawansowanych, prekonfigurowanych obiektów graficznych powszechnie używanych w procesach technologicznych (rys. 1).



Rys. 1. Zestaw SymbolFactory  
Fig. 1. Symbol Factory Kit

Kolejnym krokiem po przygotowaniu strony graficznej jest określenie zmiennych i związanych z nimi animacji.

Program InTouch umożliwia wykorzystanie dwóch podstawowych typów zmiennych:

- własnych, inaczej pamięciowych (w oryginale nazywanych Memory tags),
- globalnych, wymiennych z innymi aplikacjami Windows (I/O tags).

Jeśli uruchamiamy prostą aplikację demonstracyjną bez połączenia z rzeczywistym programem komunikacyjnym (I/O Server) lub programem symulującym komunikację ze sterownikiem PLC, zmienne własne dyskretne będą wystarczające do tego celu. Zmienne własne są również używane do uaktywniania i wyłączania połączeń animacyjnych.

Drugim podstawowym typem zmiennych są zmienne globalne. Program InTouch umożliwia zdefiniowanie zmiennych, których wartości mogą być na bieżąco wykorzystywane przez inne aplikacje Windows® za pośrednictwem programów komunikacyjnych korzystających z dynamicznej wymiany danych (DDE). Wartości zmiennych tego typu wykorzystywanych przez inne aplikacje, są uaktualniane automatycznie bezpośrednio po zmianie wartości zmiennej u źródła. W celu umożliwienia wymiany danych z programem komunikacyjnym zmienna musi być typu globalnego (I/O). Zastosowane zmienne w aplikacji przedstawiającej pasteryzację mleka wraz z opisem zamieszczono na rys. 2.

Z punktu widzenia użyteczności programu wizualizacyjnego do przedstawiania wyników symulacji komputerowej realizowanej na modelu, najistotniejszy jest wspomniany protokół DDE. Jest on odpowiedzialny za wymianę danych pomiędzy jednym programem a drugim. Dodatkowo zachodzi ona w czasie rzeczywistym. Dynamic Data Exchange (DDE) jest protokołem opracowanym przez Microsoft® w celu umożliwienia przesyłania danych i instrukcji pomiędzy aplikacjami pracującymi w środowisku Windows. Protokół ten implementuje architekturę klient- serwer pomiędzy dwoma, współbieżnie pracującymi aplikacjami. Aplikacja serwera udostępnia dane i przyjmuje żądania przesłania danych wysyłane przez inne aplikacje. Aplikacje wysyłające żądania nazywane są klientami. Niektóre aplikacje, jak na przykład InTouch czy Microsoft Excel® mogą być jednocześnie zarówno klientami jak i serwerami.

Dane w programie komunikacyjnym identyfikowane są poprzez podanie nazwy, składającej się z trzech składników: nazwy aplikacji, nazwy tematu, oraz nazwy elementu. W celu uzyskania danych z innej aplikacji, program klienta (InTouch) otwiera kanał do programu serwera poprzez podanie tych trzech składników.

Nazwa zmiennej	Typ zmiennej	Nazwa dostępu	Grupa alarmów	Komentarz
\$Time	System Integer			Time
\$TimeString	System Message			TimeString
\$VerifiedUserName	System Message			VerifiedUserName
\$Year	System Integer			Year
a	Memory Discrete		\$System	
agregat	Memory Discrete		\$System	
Analog_Tag	Memory Integer		\$System	
AppNam	I/O Discrete	PLC1	\$System	
AppName	Memory Integer		\$System	a
b	Memory Integer		\$System	
c	Memory Discrete		\$System	
f	Memory Discrete		\$System	
Format-Text	Memory Integer		\$System	
Format_Text	I/O Discrete	PLC1	\$System	
HTrend	Hist Trend		\$System	
MIARKA	Memory Integer		\$System	
MIARKA1	I/O Discrete	PLC1	\$System	
MIARKA1IF	Memory Integer		\$System	
miarkaIF	Memory Integer		\$System	
p2	Memory Integer		\$System	
para	Memory Integer		\$System	
para1	Memory Real		\$System	
pomp3	Memory Discrete		\$System	
pompa2	Memory Discrete		\$System	
poziom_mleka	Memory Integer		\$System	
r	Memory Discrete		\$System	
s	Memory Discrete		\$System	
t	Memory Real		\$System	
temp	Memory Integer		\$System	
temperatura	Memory Integer		\$System	
temperatura1	Memory Discrete		\$System	
Text1	Memory Integer		\$System	
y	Memory Real		\$System	
z	Indirect Analog		\$System	
z1	Memory Integer		\$System	
zawór	Memory Discrete		\$System	

Pole: <zaden>      OK

Filtr: <zaden>      Anuluj

69 elem.      \$AccessLevel

Rys. 2. Zmienne zastosowane w aplikacji

Fig. 2. Variables used in application

Przykładowo dla arkusza kalkulacyjnego Excel, nazwą aplikacji jest „Excel”, nazwą tematu jest nazwa konkretnego arkusza kalkulacyjnego zawierającego dane, natomiast nazwą elementu jest identyfikator komórki arkusza, która będzie wykorzystywana w trakcie operacji czytania lub zapisywania danych.

=EXCEL|ARKUSZ1!'w1k1'

W celu skorzystania z danych programu InTouch z poziomu Excela w komórce do której ma być zapisana odczytywana wartość, należy wpisać następujące wyrażenie:

```
=VIEW|TAGNAME!'NazwaZmiennej'
```

Jednym z najbardziej istotnych elementów funkcjonalności aplikacji InTouch jest możliwość korzystania ze skryptów. Skrypty dają możliwość wykonywania poleceń oraz operacji logicznych w zależności od spełnienia określonych kryteriów. Przykładowo, wciśnięcie klawisza może powodować otwarcie okna, zmianę wartości zmiennej, itd. Funkcje skryptowe wykorzystano do sterowania pracą systemu symulującego proces pasteryzacji.

Funkcje skryptowe QuickScript są skryptami, które można wywoływać z poziomu innych skryptów i połączeń animacyjnych. Kod źródłowy takiej funkcji pamiętany jest w jednym miejscu. Tak więc jej edycja umożliwia wprowadzenie zmian we wszystkich aplikacjach, w których funkcja taka jest wywoływana.

Skrypty umożliwiają utworzenie szeregu zautomatyzowanych funkcji systemowych dostosowanych do potrzeb użytkownika. Wszystkie skrypty są sterowane zdarzeniami. Zdarzeniem może być zmiana wartości, warunek, kliknięcie myszą, timer, itp. Kolejność przetwarzania zależna jest od aplikacji. Chociaż może się wydawać, że istnieje pewna kolejność wykonywania skryptów zainicjowanych przez to samo zdarzenie, nie ma żadnej gwarancji co do tej kolejności. Z tego powodu nie należy budować algorytmów w oparciu o kolejność przetwarzania. Edytor skryptów QuickScript w zasadzie nie ulega zmianie dla różnych typów skryptów (rys. 3).

Animację przedstawiającą pracę linii pasteryzacji wywołuje się w programie WindowViewer. Podczas pracy program rejestruje wszystkie parametry procesu pasteryzacji i przechowuje w bazie danych. Wybranie w menu programu okna trendy umożliwia odczytanie tych danych z dowolnego przedziału okresu czasu. System informuje również, za pomocą ostrzeżeń i alarmów (przypisanych do określonych parametrów m.in. temperatura pasteryzacji lub schładzania), o wszelkiego rodzaju nieprawidłowościach występujących w czasie tego procesu. Na rys. 4 zamieszczono ekran wizualizacji przy uruchomionym alarmie. Włączenie sygnalizacji alarmowej wywołała za niska temperatura pasteryzacji. Mleko które przepływało przez pasteryzator i uzyskiwało temperaturę niższą od wymaganej powtórnie zostało skierowane do pasteryzacji. Taka sytuacja została spowodowana wyłączeniem kotła, który jest odpowiedzialny za podgrzewanie czynnika pośredniczącego przy ogrzewaniu mleka. Nie działał również agregat, który z racji braku przepływu strumienia masowego mleka po pasteryzacji wyłączył się.

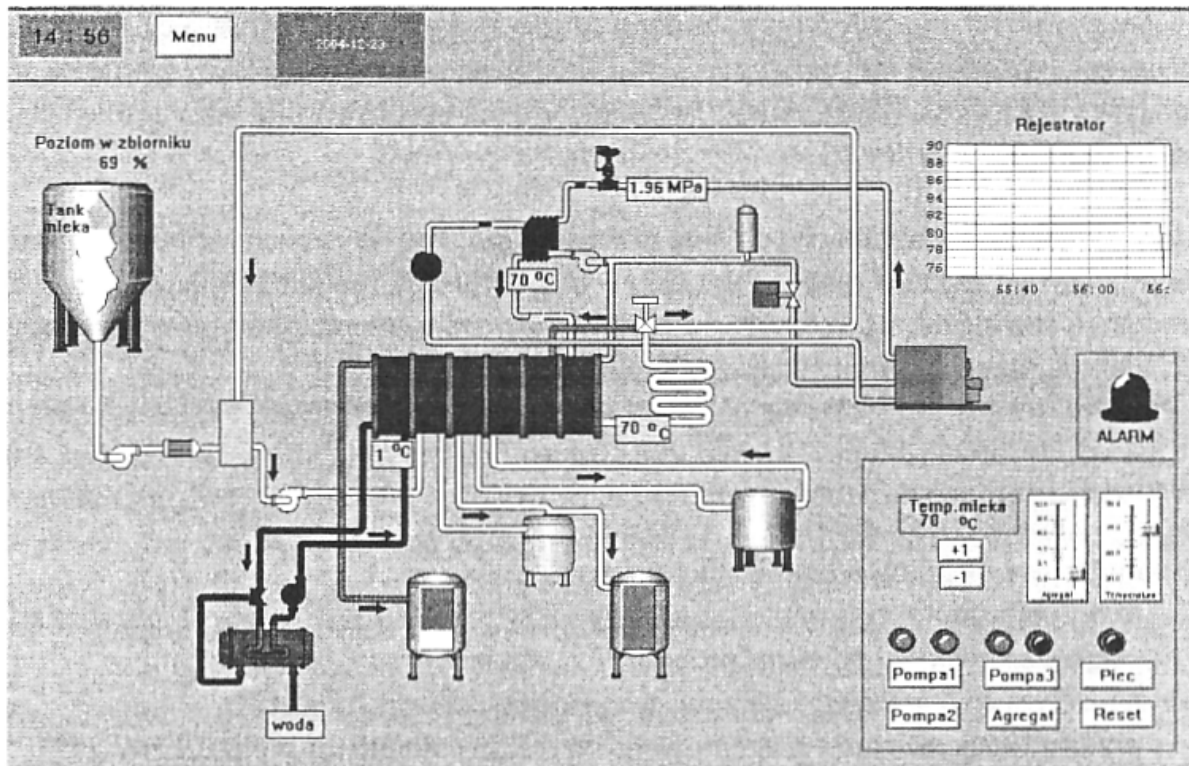


Rys. 3. Edytor skryptów w programie InTouch

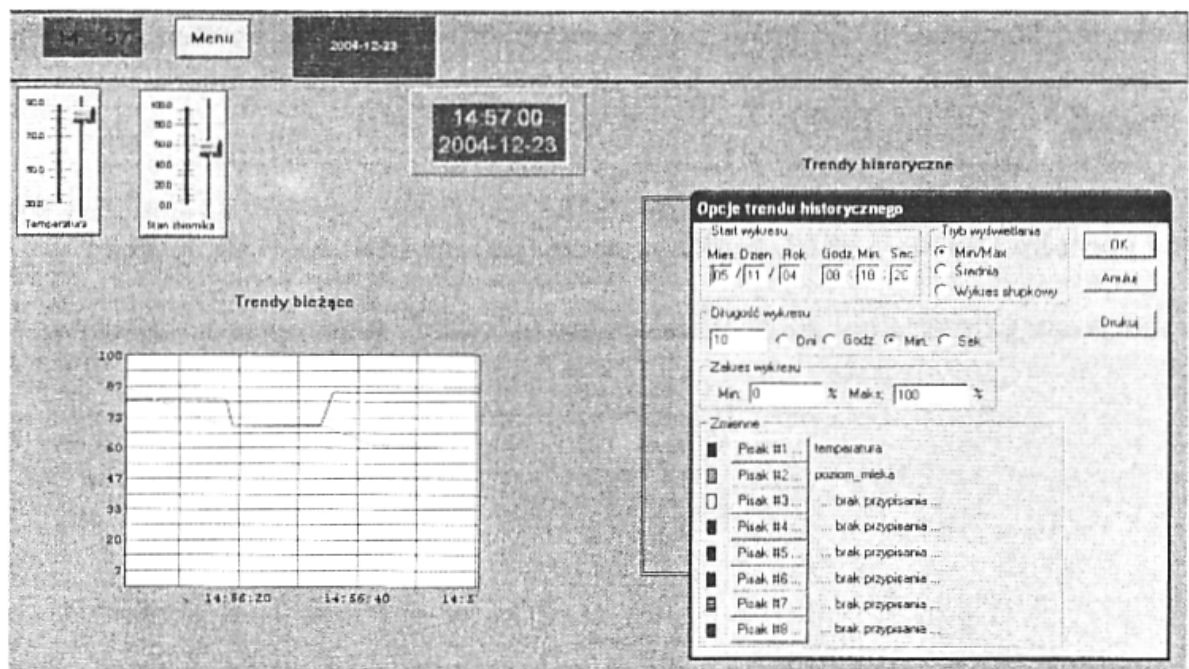
Fig. 3. The script editor in InTouch

Odczytu danych archiwalnych (rys. 5), (m.in. zarejestrowanej temperatury i poziomu w zbiorniku) można dokonać z poziomu okna „Trendy” po wejściu w zakładkę „Menu”.

Umieszczono na nim dwa wykresy. Jeden przedstawia charakterystykę poziomu w zbiorniku i temperaturę mleka w bieżącym przedziale czasowym. Drugi pozwala wywołać po podaniu daty początkowej i przedziału czasowego archiwalne dane zarejestrowanych wartości temperatury i poziomu mleka w zbiorniku.



Rys. 4. Tryb pracy awaryjnej  
 Fig. 4. Stand-by working mode



Rys. 5. Odczyt danych bieżących i wywoływanie danych archiwalnych  
 Fig. 5. The reading current data and the generating archival data

Zastosowanie wizualizacji komputerowej podnosi jakość modeli. Cytując Edwards'a Deminga uważanego za „guru” jakości: „Nie można poprawić tego, czego nie da się zobaczyć” [Rewilak 1997]. Zatem łatwo jest uzasadnić stosowanie takich rozwiązań jak prezentowane powyżej.

Rozwój technik komputerowych oraz przedstawione możliwości systemów wizualizacji ułatwiają modelowanie i interpretowanie wyników modelowania. Zastosowanie takich rozwiązań na szczeblu przedsiębiorstwa wspomaga podejmowanie decyzji, ułatwia przewidywanie stanów alarmowych i innych.

## **Wnioski**

Komputerowe systemy wizualizacyjne umożliwiają:

- tworzenie interaktywnych ekranów synoptycznych pozwalających na wizualizację sterowanego obiektu (procesu) lub modelu,
- nadzorowanie w czasie rzeczywistym (trendy, alarmy, raporty),
- przejrzyste (graficzne) prezentowanie wyników symulacji komputerowych,
- współpracę z różnymi systemami i aplikacjami, łączność z bazami danych, programami w środowisku Windows® korzystającymi z DDE.

## **Bibliografia**

Juszka H., Tomasik M., Jagła W. 2003. Komputerowa symulacja systemu automatycznego sterowania procesem przepływowej pasteryzacji wina. *Problemy Inżynierii Rolniczej*. nr 3 (41), s. 89-96.

Rewilak J. 1997. Komputerowe wspomaganie systemów zapewnienia jakości według norm ISO 9000 - część I. *Biuletyn automatyki*, nr 14/4, ASTOR, Kraków.

Wonderware Corporation. 2005. *Wonderware InTouch®*. Astor sp. z o.o. Kraków.



## THE COMPUTER VISUALIZATION – THE INSTRUMENT WHICH AIDS MODELLING AGRICULTURAL PROCESS

### Summary

It was shown the possibility of using the computer visualization in showing results of the mathematical modelling. The visualization consist in illustrating results of the mathematical model in the computer simulation. It was illustrated by using animated text and graphic objects. The connected total is making by the agency of control actions of displaying and deleting them. The way of making it was shown for the computer simulation which was carried out in Matlab®. The communication between programmes: the first which carries out the simulation and the second which is responsible for visualization, is making by the agency of DDE protocol.

**Keywords:** computer visualization, DDE, scripts, trends

Praca naukowa finansowana ze środków KBN w latach 2002-2005 jako projekt badawczy