

Sergiusz Maćkowiak, Wojciech Mueller, Jerzy Weres, Piotr Boniecki
Instytut Inżynierii Rolniczej
Akademia Rolnicza w Poznaniu

XML JAKO INTERFEJS ZEWNĘTRZNEGO KOMUNIKOWANIA SIĘ SYSTEMU INFORMATYCZNEGO SYMULUJĄCEGO DZIAŁANIE KAMIENNEGO REGENERATORA

Streszczenie

Efektywność wykorzystania systemów informatycznych tworzonych i stosowanych w sektorze rolniczym jest nie tylko konsekwencją oferowanych przez nie funkcjonalności, ale możliwością wkomponowania ich w pewną spójną całość. Jednym z prostszych sposobów pozwalających na kompleksowe wykorzystanie aplikacji, co niejednokrotnie stwarza nowe możliwości jest ujednoczenie zapisu danych wejściowych i wyjściowych, w takim formacie, który nie generuje problemów związanych z ich przesyłaniem w Internecie. Efektywną techniką informatyczną pozwalającą na realizację tego celu jest technologia XML, która swą nazwę zawdzięcza rozszerzalnemu językowi znakowania. Bazując na tej technologii autorzy dokonali stosowanych zmian w systemie informatycznym Akterm 2003 obrazującym fazę ładowania kamiennego regeneratora ciepła. Dopuszenie aplikacji w moduł pozwalający na pobieranie prognoz pogodowych w postaci dokumentów XML udostępnianych przez specjalistyczne serwisy, daje możliwość szacowania z wybranym wyprzedzeniem czasowym ilość energii skumulowanej w złożu kamiennym. Natomiast udostępnienie zapisu uzyskanych wyników symulacji komputerowych na bazie nowej aplikacji Akterm2005_NET w wspomnianym standardzie XML stwarza możliwości ich dalszego zindywidualizowanego przetwarzania przez kolejne systemy informatyczne. Postawione zadania rozbudowy aplikacji wraz z dążeniem do ograniczenia pracochłonności przesądziły o wyborze środowiska programistycznego. Implementację zrealizowano z wykorzystaniem platformy Visual Studio .NET 2003 firmy Microsoft.

Słowa kluczowe: regenerator kamienny, akumulator energii cieplnej, technologia XML

Wprowadzenie

Rozszerzalny język znakowania XML jest już powszechnie uznawanym standardem wymiany danych i publikowania informacji w bazach danych i systemach informatycznych. Usługi XML sieci Web wykorzystują z powodzeniem ten otwarty standard internetowy w celu umożliwienia komunikacji między systemami i źródłami danych lub między systemami napisanymi w różnych językach na różnych platformach. Aktualnie XML to znacznie więcej niż tylko prosty interfejs wymiany danych pomiędzy systemami i użytkownikami. Generalnie jest to pojęcie wieloznaczne [Ray 2001], ale najogólniej mówiąc jest to taki sposób patrzenia na informacje, który prowadzi do ich prezentacji w najczystszej i najlepiej ustrukturalizowanej postaci. Gwarantuje to zwiększenie użyteczności informacji dostępnych w tej formie i ułatwia proces wielokierunkowego ich przetwarzania. Trudno dziś znaleźć internetowe systemy informatyczne, które nie wykorzystują, choć w pewnej części tej technologii.

Istniejące i powstające aplikacje wspomagające funkcjonowanie sektora rolniczego mają z reguły charakter wycinkowy, co jest wynikiem złożoności produkcji rolniczej, a także ograniczeniami ekonomicznymi, mającymi swoje źródło w niezbyt licznej grupie potencjalnych użytkowników tych wysoce specjalistycznych aplikacji. Spoiwem dla tych różnorodnych systemów informatycznych, powstałych i powstających w oparciu o różne technologie informatyczne oraz posiadających różnicowany charakter, jest technologia XML. Należy nadmienić, iż występująca różnorodność technologiczna jest i będzie immanentną cechą informatyki stosownej.

Dodatkowo wzrastające ryzyko działalności gospodarczej w tym również rolniczej wraz z dostępem do coraz większej ilości informacji, zgromadzonych w specjalistycznych serwisach, mających wpływ na produkcję rolniczą stawia coraz większe wymagania system informatycznym w obszarze prognozowania. Również w tej dziedzinie rola technologii XML jest i będzie nieoceniona.

Biorąc to pod uwagę oraz zważając na fakt przewidywanego zwiększonego wykorzystania w rolnictwie systemów niekonwencjonalnych źródeł energii, których elementami składowymi są akumulatory ciepła, rozbudowano system informatyczny symulujący działanie kamiennego regeneratora. Wyposażając tą aplikację w moduł do pobierania danych z serwisów pogodowych, zawierających prognozy, w postaci dokumentów XML uzyskano możliwość szacowania ilości energii zgromadzonej w złożu w trakcie przewidywanych okresów ładowania. Tego typu

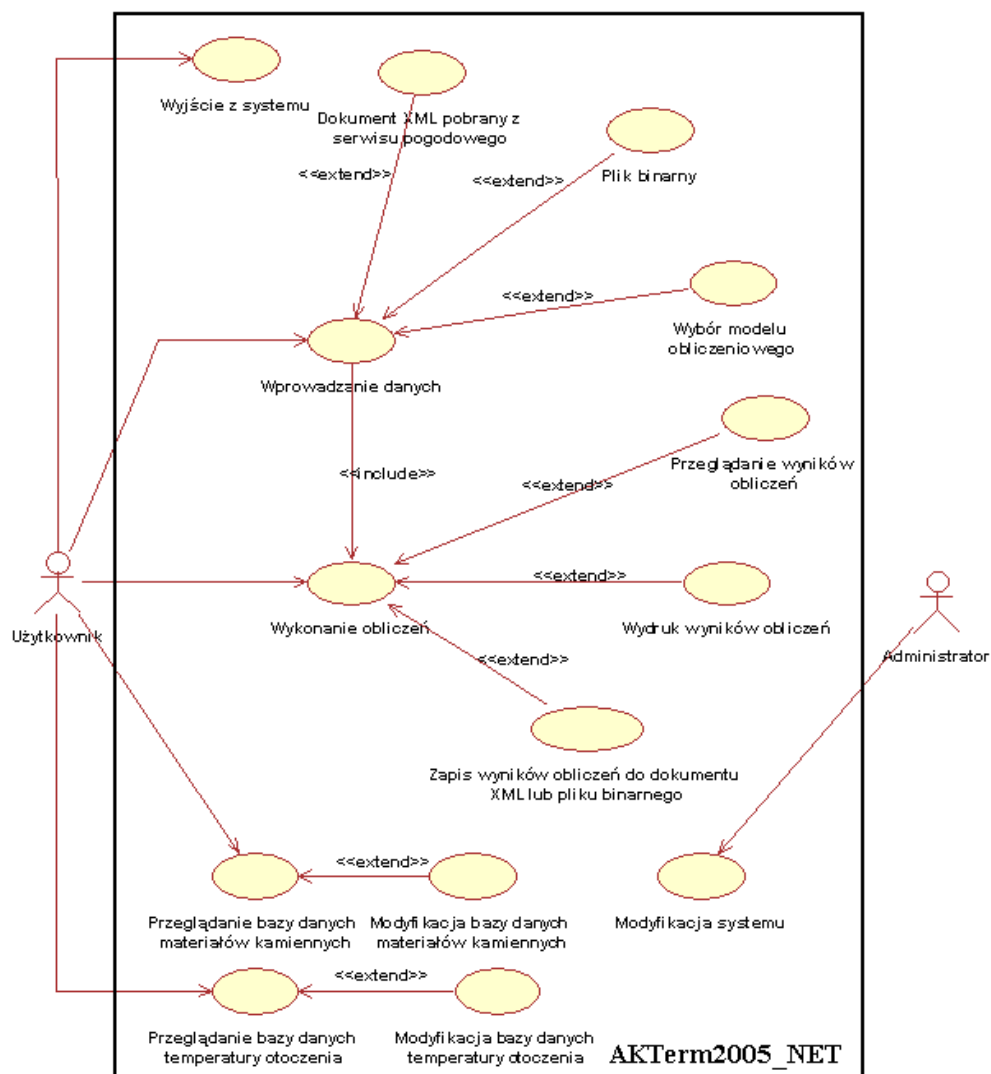
prognozy są szczególnie przydatne w prostych systemach niekonwencjonalnych źródeł energii, gdzie dysponujemy tylko jednym źródłem. Przewidując powstawanie kolejnych systemów informatycznych, dla których dostarczycielem danych będzie aplikacja Akterm2005_NET, a także celem ułatwienia analiz danych pod kątem badawczym, doposażono ją w moduł pozwalający na zapis informacji w sygnalizowanym standardzie.

Bogate biblioteki klas przeznaczonych do budowy i przetwarzania dokumentów XML, będące na wyposażeniu środowiska Microsoft Visual Studio .NET 2003, zdecydowały o przyjęciu tego narzędzia programistycznego do wytworzenia niezbędnych interfejsów pobierania i udostępniania danych systemu informatycznego Akterm2005_NET.

Model obiektowy systemu informatycznego Akterm2005_NET

Dotychczasowy system informatyczny Akterm 2003, pozwalający na przeprowadzenie szerokiej gamy symulacji procesu ładowania kamiennego regeneratora z uwzględnieniem zarówno losowego jak i deterministycznego charakteru wybranych wielkości fizycznych istotnych z punktu widzenia procesu przepływu ciepła, będący samodzielnym bytem informatycznym posiadał ograniczone możliwości współpracy z innymi aplikacjami. Ograniczenia te nie pozwalały na automatyzację procesu pobierania danych oraz udostępniania ich innym systemom w postaci niegenerującej różnorodnych problemów związanych z ich dalszym przetwarzaniem.

Rozbudowę systemu informatycznego Akterm2005_NET, zmierzającą do eliminacji tych barier, poprzedziło zgodnie z regułami inżynierii oprogramowania modelowanie obiektowe zrealizowane w notacji graficznej UML. Efektem podjętych działań było wytworzenie diagramów przypadków użycia, których jeden z poziomów przedstawiono na rys. 1. Ta graficzna forma prezentacji w sposób przejrzystszy i z różnych perspektyw szczegółowości pozwala spojrzeć na dynamikę projektowanego systemu, mającego swe źródło w zidentyfikowanych wymaganiach przyszłych użytkowników. Uaktualniony model przewiduje zarówno pobieranie danych w standardzie XML, dostępnych w specjalistycznych serwisach pogodowych zlokalizowanych na zdalnych serwerach WWW, jak również na zapis przetworzonych informacji w tym formacie. Statyczna, zidentyfikowana struktura systemu na etapie modelowania została odwzorowana w diagramach klas.



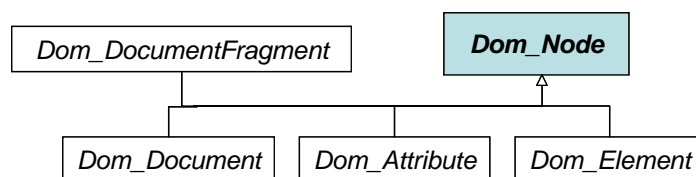
Rys. 1. Diagram UML przypadków użycia
 Fig. 1. Diagram UML of use cases

XML w środowisku programistycznym Microsoft Visual Studio .NET

Przetwarzanie dokumentów XML w ramach języków programowania odbywa się z wykorzystaniem interfejsów API, które z uwagi na obszar działania dzielimy na trzy główne poziomy [Arciniegas 2002]:

- interfejsy API ogólnego przeznaczenia,
- interfejsy API implementujące niebazowe specyfikacje XML,
- interfejsy API opracowane dla określonych terminologii.

Najczęściej w procesie programowego manipulowania dokumentami XML wykorzystujemy interfejsy ogólnego przeznaczenia, które tworzą zarówno interfejsy zdarzeniowe, jak i zorientowane obiektowo. Do grupy interfejsów zorientowanych obiektowo zaliczamy ten sposób modelowania dokumentu, w którym jest on odwzorowywany w zbiorze obiektów. Stanowią one następnie przedmiot działania aplikacji, która poprzez odpowiednie metody obiektów pobiera i modyfikuje zawarte w nich dane. Najważniejszym i jednocześnie najpopularniejszym zestawem interfejsów tego rodzaju jest opracowany przez konsorcjum W3C obiektowy model dokumentów DOM (Document Object Model). Podstawowe interfejsy służące do reprezentacji zasadniczych obiektów abstrakcyjnych tworzących dokument jak i samego dokumentu reprezentuje rysunek 2 [Arciniegas 2002]. Umieszczony na nim diagram wskazuje, że wszystkie interfejsy są podrzędne w stosunku do interfejsu węzła.



Rys. 2. Podstawowe komponenty modelu dokumentu DOM

Fig. 2. The basic components of model of document DOM

Na poziomie środowiska programistycznego zarówno Visual Studio .NET 2003 i 2005 mamy zaimplementowane zarówno interfejsy zdarzeniowe i obiektowe [Mackenzie, Sharkey 2002]. Stosowanie interfejsów obiektowych może przebiegać w dwojaki sposób dzięki wykorzystaniu:

- DOM (*Document Object Model*) – obiektowego modelu opisu dokumentu,
- zestawu klas służących do odczytu i zapisu informacji w formacie XML.

Porównanie implementacji interfejsów obiektowych dostępnych w środowisku Visual Studio .NET ułatwiające dokonanie wyboru przedstawiono w tabeli 1. Z punktu widzenia celu podjętej rozbudowy aplikacji wygodniejszy do wykorzystania okazał się zestaw klas pozwalający na odczyt i zapis danych w standardzie XML.

Tabela 1. Wady i zalety mechanizmów dostępu do dokumentów XML
 Table 1. Disadvantages and advantages of mechanisms of access to documents XML

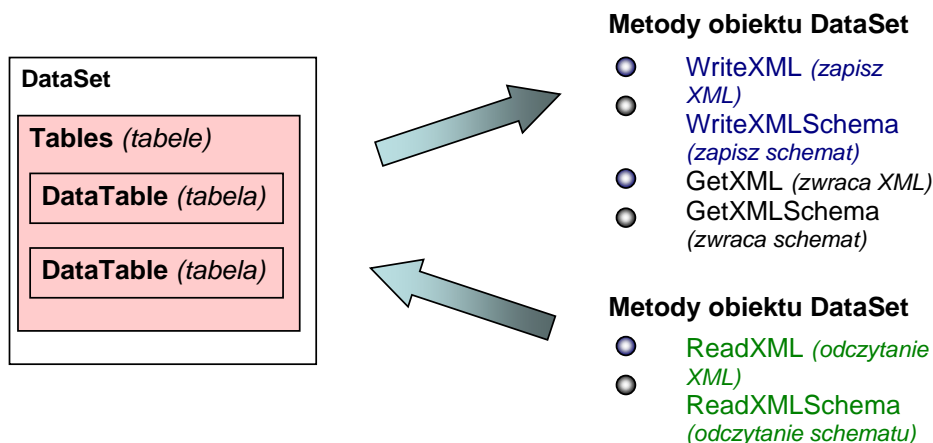
Mechanizm dostępu	Zalety	Wady
DOM	Pozwala na pełne jednoczesne odwzorowanie całego dokumentu.	Wymaga wczytania całego dokumentu do pamięci komputera, co powoduje przy dużych rozmiarach pliku XML znaczne obciążenie zasobów jednostki obliczeniowej.
Klasy XmlTextReader, XmlTextWriter	Odczyt i zapis odbywa się szybciej, ponieważ nie trzeba ładować całej często dosyć rozległej struktury dokumentu.	Odczytują tylko zadane fragmenty dokumentu.

Klasy tworzące ten zestaw *XmlTextReader* oraz *XmlTextWriter*, jak i pozostałe klasy będące implementacją interfejsów związane z przetwarzaniem danych w formacie XML dostępne są w przestrzeni nazw *System.XML* środowiska wizualnego NET [Mackenzie D., Sharkey K. 2002]. Pierwszy z sygnalizowanych obiektów wraz z metodą *Read* pozwala na sekwencyjne wczytywanie węzłów dokumentu XML, czemu towarzyszy zmiana właściwości *Name* i *NodeType* tego obiektu. Wspomniane cechy z reguły stanowią kryterium dalszego przetwarzania pozyskanych informacji. Efekty operacji wykonanych na danych mogą być udostępniane w postaci dokumentu XML, poprzez wykorzystanie obiektu *XmlTextWriter*. Wyposażony jest on w szereg metod typu *Write* takich jak:

- *WriteElementString* tworzenie nowego elementu,
- *WriteStartElement* tworzenie znacznika początkowego nowego elementu,
- *WriteEndElement* tworzenie znacznika zamykającego.

Metody te wykorzystujemy na poziomie publikowania danych w standardzie XML. Tworzenie bardziej złożonych dokumentów opartych na definicji typu dokumentu DTD lub schemacie przy użyciu tych narzędzi jest utrudnione.

Alternatywą do przedstawionych powyżej obiektów przeznaczonych do przetwarzania dokumentów XML zwłaszcza w sytuacji, gdy miejscem docelowym danych ma być system zarządzania bazami danych jest użycie komponentu *DataSet* [MacManus, Goldstein 2005; <http://msdn.microsoft.com/library/>]. Ów obiekt przynależny do przestrzeni nazw *System.Data* wyposażony jest w bogaty zestaw metod pozwalających na dwukierunkowy przepływ informacji w formacie XML – rys. 3. Uniwersalny charakter tego „zasobnika” danych i jego niezależność od źródeł informacji pozwala na ich wygodne przetwarzanie włączając w to mechanizm łączenia danych pochodzących od różnych dostawców.

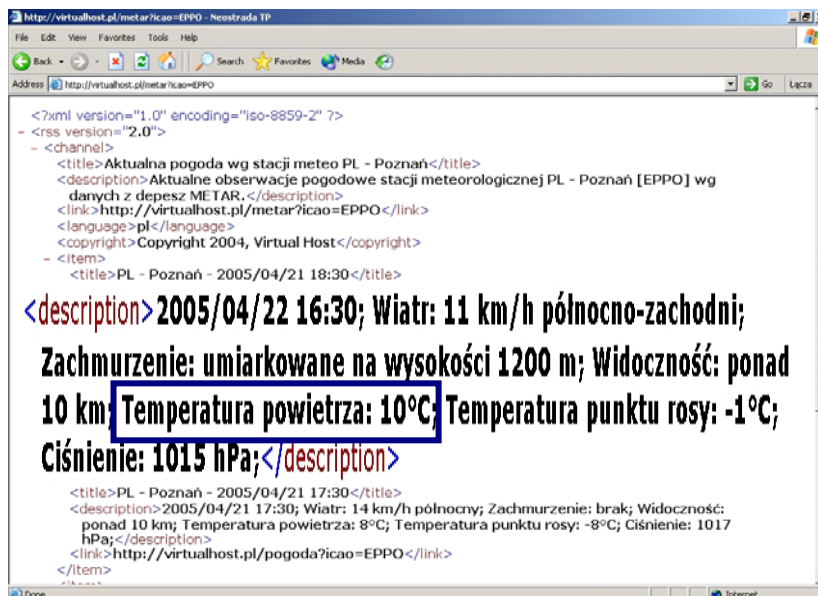


Rys. 3. Zestaw metod obiektu DataSet ukierunkowanych na dane XML

Fig. 3. The set of methods of object DataSet oriented on data XML

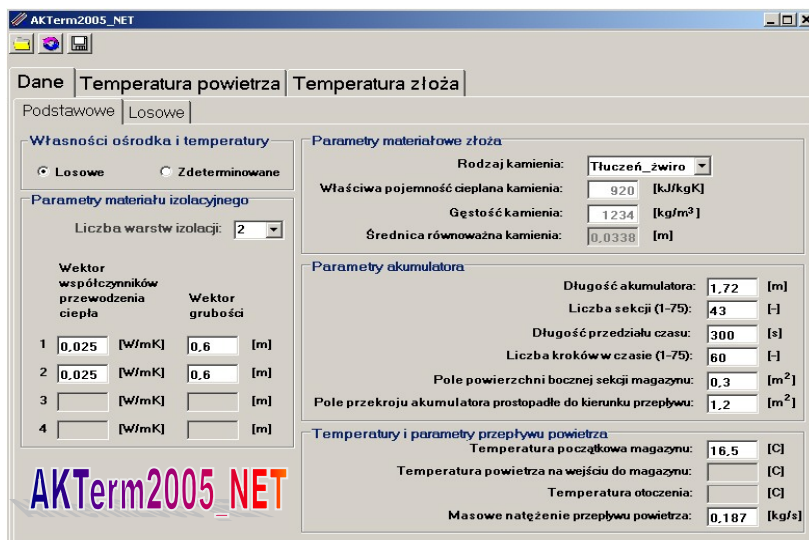
Interfejsy XML w systemie AKTerm2005_NET

Na bazie efektów modelowania obiektowego dostępnych w postaci diagramów klas oraz przypadków użycia, wykorzystując możliwości jakie oferuje wizualne środowisko programistyczne Microsoft Visual Studio .NET 2003 przystąpiono do implementacji systemu. Wytworzony system informatyczny Aktem2005_NET, będący następcą poprzedniej wersji o roboczej nazwie Akrem 2003, został wzbogacony o możliwość pobierania danych z serwisu pogodowego. Informacje o aktualnym stanie pogody jak ich prognozy pobierane są ze strony: <http://virtualhost.pl/pogoda> (rysunek 4). Owe dane dostępne są w formacie XML, a ich automatyczna aktualizacja odbywa się, co 30 minut. Wykorzystując mechanizm dostępu do danych przy pomocy klasy XMLTextReader, informacje o prognozach pogody przekazane zostają do formularza aplikacji (rysunek 5). Na ich podstawie wykonywana jest symulacja procesu ładowania kamiennego regeneratora. Wyniki uzyskanych parametrów termodynamicznych obrazujących opisany proces, jak i ilość energii cieplnej zgromadzonej w złożu mogą zostać zapisane w pliku binarnym lub dzięki wykorzystaniu XML-owego interfejsu komunikacji mogą zostać opublikowane w tym standardzie, co ilustruje rys. 6. Stwarza to możliwości dalszego ich przetwarzania przez kolejne systemy informatyczne.



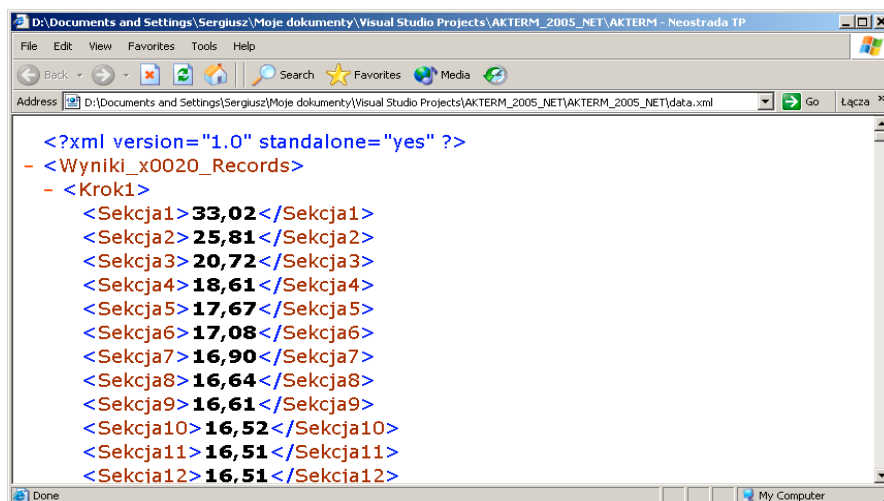
Rys. 4. Informacje pogodowe zapisane w formacie XML

Fig. 4. Information about weather saved in format XML



Rys. 5. Formularz danych wejściowych do realizacji symulacji

Fig. 5. Form of input data to realization of simulation



Rys. 6. Wyniki zapisane w formacie XML

Fig. 6. Results saved in format XML

Podsumowanie

Podjęty zakres działań, obejmujący wielopłaszczyznowe modelowanie obiektowe oraz przebiegające równolegle rozpoznawanie najnowszych technologii związanych z XML połączone z dokonywaniem wyboru w tym obszarze, a następnie implementacja systemu informatycznego opisującego fazę ładowania kamiennego regeneratora i jego testowanie, pozwoliły na sformułowanie następujących uwag:

1. Zmiana środowiska programistycznego na platformę Microsoft .NET z uwagi na dostępne w nim szerokie bogactwo predefiniowanych klas dedykowanych technologii XML pozwala na pełniejsze wykorzystanie standardu tego języka.
2. Zastosowane klasy XmlTextReader i XmlTextWriter do odczytu i zapisu dokumentów XML o niezbyt złożonej strukturze gwarantuje szybsze pobieranie i udostępnianie danych w tym standardzie w porównaniu z techniką DOM.
3. Wykorzystanie w aplikacji technologii XML w procesie komunikacji i wymiany informacji pozwala na:
 - prognozowanie wielkości energii zgromadzonej w regeneratorze kamiennym na podstawie danych temperaturowych umieszczanych w serwisach pogodowych WWW,
 - udostępnianie danych w powszechnie akceptowanym standardzie innym systemom informatycznym w tym następnym aplikacjom symulującym kolejne fazy działania regeneratora.

Bibliografia

- Arciniegas F. 2002. C++ XML. Wydawnictwo MIKOM Warszawa
<http://msdn.microsoft.com/library/>.
<http://virtualhost.pl/pogoda>.
- Mackenzie D., Sharkey K. 2002. Visual Basic .NET dla każdego. Wydawnictwo Helion.
- MacManus J., P., Goldstein J. 2005. Visual Basic .NET Bazy Danych. Wydawnictwo Helion.
- Ray E., T. 2001. Nauka języka XML. Wydawnictwo RM Warszawa.

XML AS INTERFACE OF OUTSIDE COMMUNICATING OF INFORMATION SYSTEM SIMULATING WORKING OF STONE REGENERATOR

Summary

The efficiency of the use of computer systems created and utilised in the agricultural sector is not only a consequence of the functionalities offered by them but possibility of putting them into a coherent set. One of simpler ways allowing for the complex use of the application, which often generates new possibilities is the unification of input and output data formats which do not create problems with their transfer in the Internet. All these requirements are fulfilled with the XML technology, which takes name after Extensible Markup Language. Basing on the technology the authors made appropriate changes to the computer system "Akterm 2003" demonstrating the loading phase of stone regenerator. The extra module for the download of the weather forecasts in the XML format provided by special services, allows to estimate the amount of the energy accumulated in the stone bed with set time in advance. Moreover, sharing the results of the computer simulation using the new application "Akterm 2005" and in the mentioned XML format gives opportunities for their further individualised utilisation in other computer systems. The set extension task for the application and an intention to reduce the effort shaped the decision as far as the programming environment is concerned. The implementation was realised with the use of Microsoft Visual Studio .NET 2003.

Key words: stone regenerator, thermal energy storage, technology XML