

Edytor stron WWW

Jakub Duda,
Hydro-Eco-Invest, Gliwice

Dokumentacja użytkownika

1.	Wprowadzenie	3
2.	Zastosowania i możliwości programu	3
3.	Praca z programem	3
3.1	Sposób instalacji i uruchomienia	3
3.2	Tworzenie nowego projektu	4
3.3	Rozmieszczenie pomiarów	5
3.4	Edycja parametrów	6
3.5	Zapis projektu	8
3.6	Klient FTP	8
3.7	Projekty zdalne	9
3.8	Konfiguracja	9
3.9	Podgląd apletu.	9
4.	Formaty plików	13
4.1	Plik „application.xml”	14
4.2	Plik „controls.xml”	15
4.3	Wzorce html	18
4.4	Plik projektu w formacie xml	19
4.5	Plik html inicjujący aplet	20
Dodatek A – Instalacja i konfiguracja środowiska Javy w przeglądarce Internet Explorer		22
A.1	Poziom bezpieczeństwa	22
A.2	Instalacja maszyny wirtualnej firmy Sun.	22
Dodatek B – przygotowania środowiska testowego systemu e-FlowNet		22
Dodatek C – Testowy projekt apletu		23

1. Wprowadzenie

Jednym z głównych zadań systemu e-FlowNet jest wizualizacja danych – informacji pochodzących z obiektów przemysłowych przezeń obsługiwanych. Istotna jest nie tylko prezentacja danych użytkownikowi, ale też przedstawienie graficznej reprezentacji obiektu wraz z umiejscowieniem na nim pomiarów. Ta metoda upraszcza odbiór informacji i daje wyobrażenie o fizycznym pochodzeniu danych.

Jako, że w systemie pracować może szeroka gama urządzeń, prezentowane dane mogą mieć bardzo różny charakter – może to być np. odczyt z pracującego gdzieś w sieci kanalizacyjnej przepływomierza czy też stan zaworu lub pompy. Innym ciekawym przykładem jest zintegrowany system monitorowania liczników energii elektrycznej znajdujących się na terenie zakładu przemysłowego – pozwalający na błyskawiczne wykrycie strat energii powodowanych np.: przez nielegalne instalacje.

Zadaniem będącego częścią oprogramowania systemu e-FlowNet *Edytora stron WWW* jest wspomaganie procesu przygotowania apletów wizualizacji. Do pewnego stopnia są one alternatywą wobec *Oprogramowania wizualizacji e-FlowNet* i serwerów OPC. Umożliwiają prezentację danych pomiarowych (tablic synoptycznych) przy użyciu przeglądarki internetowej, bez konieczności instalacji specjalnego oprogramowania na komputerze klienta. *Edytor stron WWW* całościowo wspomaga proces tworzenia apletów wizualizacji, począwszy od pobrania informacji o dostępnych urządzeniach i pomiarach, poprzez zaprojektowanie wyglądu i określenie parametrów pracy, aż do wysłania gotowych plików HTML inicjalizujących aplet do serwera www. Dzięki niemu nową tablicę synoptyczną można przygotować bez konieczności pisania programów a nawet znajomości html.

2. Zastosowania i możliwości programu

Aplety pomiarowe znajdują zastosowanie tam gdzie z różnych przyczyn nie można zastosować tradycyjnego oprogramowania wizualizacji. Co prawda takie rozwiązanie nie oferuje szerokiej możliwości przetwarzania danych charakterystycznych dla rozbudowanych programów klienckich, jednak nie jest wymagana instalacja wyspecjalizowanego oprogramowania (a nawet systemu Windows) – wystarczy komputer wyposażony w przeglądarkę internetową posiadającą wirtualną maszynę Javy i dostęp do sieci (lokalnej lub internetowej). Ważna jest też możliwość udostępniania danych urządzeniom mobilnym (np. palmtopom) wykorzystującym standard Java Micro Edition.

Jeden aplet przedstawia jedną tablicę synoptyczną, nic jednak nie stoi na przeszkodzie, by z wykorzystaniem łańcuchów hipertekstowych stworzyć złożony z wielu tablic duży system wizualizacji.

Edytor stron WWW jest modulem programu *Konfigurator e-FlowNet*. Pracuje w systemie Windows (w wersji nowszej niż Windows 95, tzn 98, ME, NT 4 sp5, 2000 lub XP). Do wygodniejszej pracy potrzebny jest procesor klasy Pentium 200 Mhz i 32 MB pamięci RAM.

3. Praca z programem

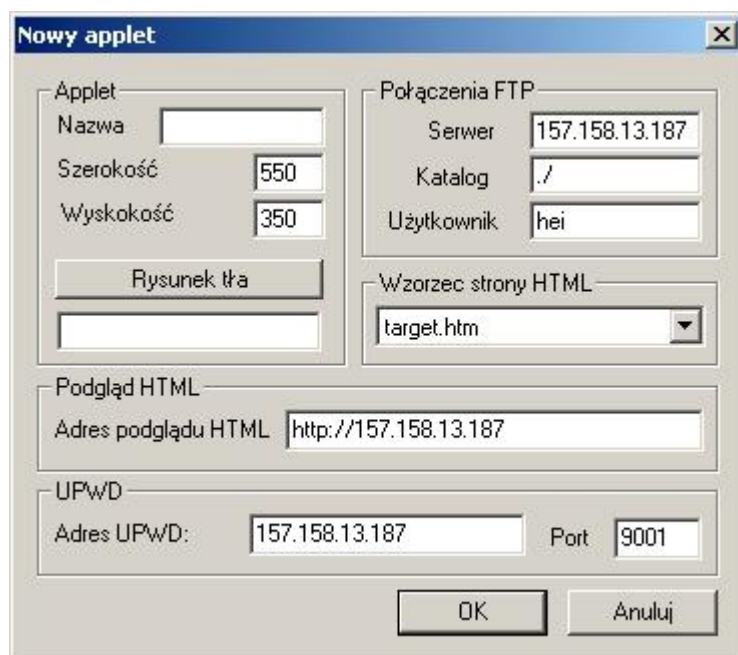
3.1 Sposób instalacji i uruchomienia

Program *Edytor stron WWW* wykorzystuje standardowy program instalacyjny InstallShield. Aby rozpocząć instalację należy wybrać opcję „Konfigurator „e-FlowNet” w menu płyty lub ręcznie uruchomić (poprzez dwukrotne kliknięcie) znajdujący się w katalogu „Konfigurator e-FlowNet” program setup.exe. Program domyślnie instalowany jest na dysku systemowym w katalogu `Program Files\Konfigurator e-FlowNet`, na przykład: `C:\Program Files\Konfigurator e-FlowNet`. Po zainstalowaniu program można uruchomić poprzez wybranie „Konfigurator e-FlowNet” w katalogu Programy menu startowego.

3.2 Tworzenie nowego projektu

Aby utworzyć nowy projekt należy wybrać pozycję „Nowy” z menu „Plik” i wybrać Aplet jako typ dokumentu.

Wówczas pojawi się okno dialogowe pozwalające na podanie początkowych parametrów projektu.



Rys. 3.1. Okno dialogowe z parametrami nowego apletu.

Widoczne na rysunku 3.1 parametry przykładowego apletu zostały zainicjowane wartościami domyślnymi (są zdefiniowane w pliku application.xml opisanym w rozdziale czwartym). Wartości powiązane znaczeniowo zostały zebrane w grupy (ramki). W pierwszej z nich użytkownik może określić nazwę, początkową wielkość a także wybrać bitmapę tła dla apletu. Nazwa projektu określa nazwę pliku html i katalogu w którym zostanie on zapisany, dlatego istotne jest by podane nazwy były poprawne w rozumieniu systemu plików.

Z kolei ramka „Połączenia FTP” pozwala na skonfigurowanie adresu i użytkownika serwera FTP na który zostanie wysłany projekt apletu (tj. pliki html, xml oraz bitmapy).

Adres podglądu HTML to adres wykorzystywany przy podglądzie pracujących apletów w przeglądarce (temat ten zostanie dokładniej wyjaśniony w dalszej części dokumentacji).

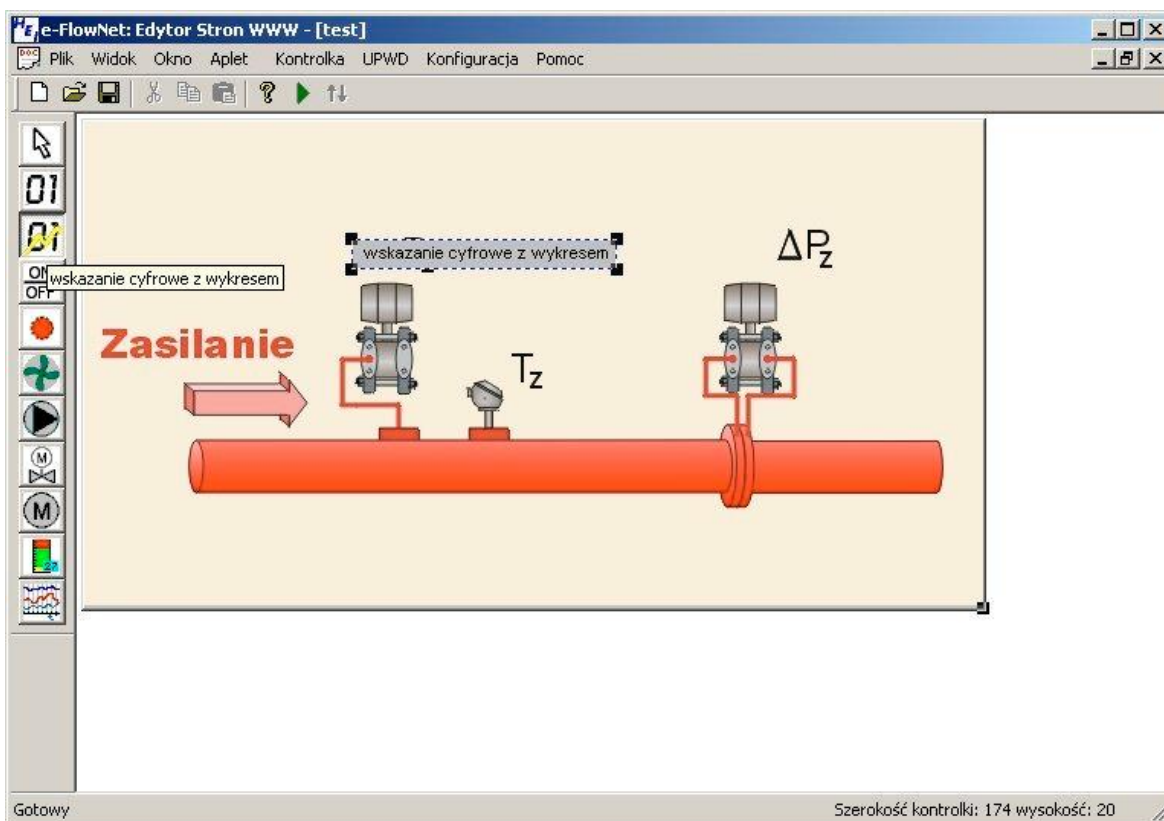
Adres i port serwera UPWD określają koncentrator danych lub serwer e-FlowNet z którego przy użyciu protokołu UPWD pobrane zostaną informacje o udostępnianych pomiarach.

Ponieważ często adresy serwerów FTP, UPWD i http pokrywają się, wpisany przez użytkownika adres serwera FTP jest automatycznie kopiowany do odpowiednich pól w ramkach „Podgląd HTML” i „Adres UPWD”.

3.3 Rozmieszczenie pomiarów

Po podaniu opisanych powyżej parametrów, program tworzy nowy projekt apletu. Kolejnym krokiem jest rozmieszczenie na nim pomiarów (kontrolki).

Przykładowy aplet przedstawia rysunek 3.2. Widoczna jest bitmapa tła i umieszczony na aplecie pomiar „wskazanie cyfrowe”.



Rys. 3.2. Projekt apletu.

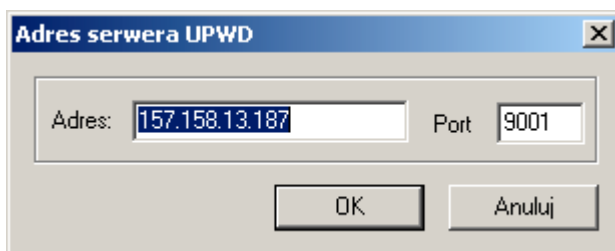
Wszystkie dostępne pomiary wyświetlane są na pasku narzędzi po lewej stronie ekranu. Jest on dynamicznie tworzony przy starcie programu na podstawie pliku konfiguracyjnego controls.xml (opisanego w rozdziale czwartym). Gdy kursor znajdzie się nad ikoną, wyświetlana jest informacja o typie pomiaru.

Aby umieścić kontrolkę na aplecie należy zaznaczyć symbolizującą go ikonę a następnie kliknąć na powierzchni apletu (na rysunku 3.2 całą powierzchnię zajmuje bitmapa tła) – na przykładowym aplecie umieszczono kontrolkę „wskazanie cyfrowe z wykresem”.

Wybranie kursora pozwala na edycję (m.in. pozycjonowanie i zmianę rozmiaru) istniejących kontrolki i samego apletu. Informacja o rozmiarze zaznaczonej kontrolki jest wyświetlana w pasku statusu.

3.4 Edycja parametrów

Po rozmieszczeniu kontrolki na powierzchni apletu należy podać ich pozostałe, niezwiązane z wyglądem parametry. Wcześniej jednak wskazane jest pobranie z sieci drzewa pomiarów i rejestrów zapisywalnych. Wykorzystanie w programie opracowanego przez firmę HEI protokołu UPWD (Uniwersalny Protokół Wymiany Danych) pozwala na pobranie informacji o udostępnianych pomiarach i rejestrach bezpośrednio z serwera komunikacyjnego lub serwera e-FlowNet. Aby to zrobić należy wybrać funkcję „Pobierz pomiary i rejestry” z menu UPWD. Wtedy pojawi się okno dialogowe pozwalające określić adres i port serwera udostępniającego pomiary.

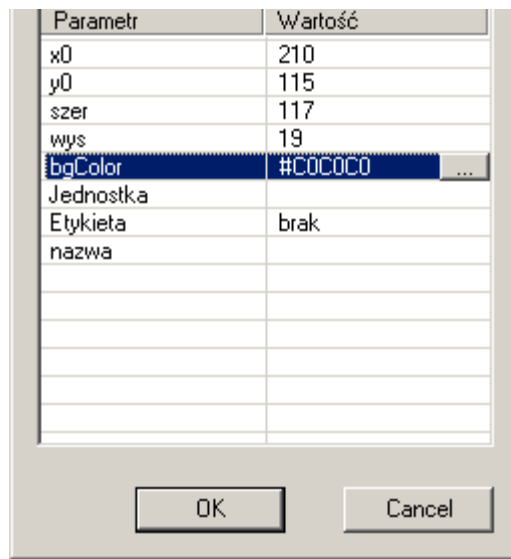


Rys. 3.3. Konfiguracja połączenia UPWD

Jak widać na rysunku 3.3 parametry połączenia zostały zainicjowane wartościami domyślnymi podanymi przy tworzeniu nowego projektu.

Po naciśnięciu Ok. program spróbuje pobrać drzewo pomiarów z serwera. Wtedy można przystąpić do edycji parametrów pomiarów i apletu.

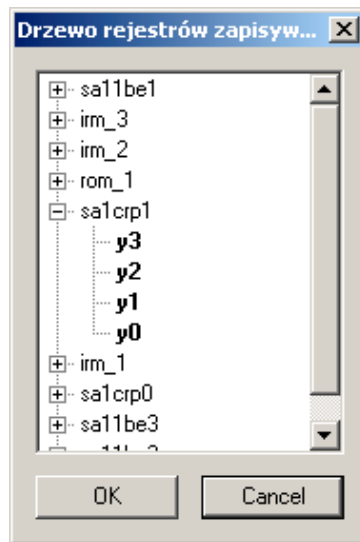
Dwukrotne kliknięcie na kontrolce lub aplecie powoduje wyświetlenie listy właściwości tego obiektu (można ją uzyskać również wybierając pozycję „Parametry” w menu „Aplet” lub „Kontrolka”).



Rys. 3.4. Okno dialogowe edycji parametrów apletu lub kontrolki.

Jak widać na rysunku 3.4, lista właściwości umożliwia edycję wszystkich parametrów kontrolki lub apletu. Można ją więc wykorzystać także do precyzyjnego określenia ich rozmiaru i położenia.

W zależności od typu, parametry reprezentowane są w liście właściwości na różne sposoby. Do modyfikacji typów napisowych i liczbowych wykorzystywane jest zwykle pole edycji, natomiast dla typów „wybór” i „boolean” jest to rozwijalne pole wyboru. Z kolei typy „rejestr”, „pomiar”, „kolor” i „plik” reprezentowane są przez przyciski otwierające odpowiednie okna dialogowe: drzewo rejestrów zapisywalnych, drzewo pomiarów, dialog wyboru koloru lub wyboru pliku.

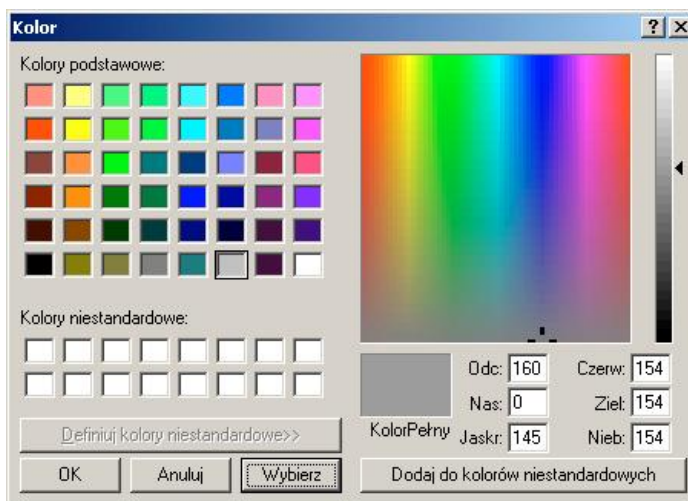


Rys. 3.5. Dialog drzewa rejestrów zapisywalnych.

Drzewo rejestrów zapisywalnych (widoczne na rysunku 3.5) i drzewo pomiarów to niemal identyczne okna dialogowe wyświetlające urządzenia i udostępniane przez nie pomiary(czyli rejestry do odczytu) lub rejestry zapisywalne.

Kliknięcie na nazwę technologiczną urządzenia rozwija listę jego rejestrów. Wytłuszczoną czcionką zaznaczone są pomiary (rejestry), których typ danych odpowiada wymaganemu przez kontrolkę – tylko takie mogą zostać wybrane (powiązane z kontrolką). Do jednej kontrolki może być przypisane wiele pomiarów – w takich przypadkach zazwyczaj efektem pracy kontrolki są przetworzone dane (wyniki operacji matematycznych przeprowadzonych na pomiarach).

Warto zwrócić uwagę, że dialog wyboru koloru (wykorzystywany np. do określania koloru tła tablicy synoptycznej) został wzbogacony o znany z programów graficznych tzw. „color picker”. Umożliwia on próbkowanie koloru z dowolnego elementu apletu, a nawet całej aplikacji.



Rys. 3.6. Dialog wyboru koloru.

3.5 Zapis projektu

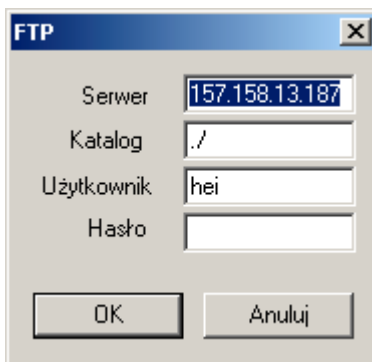
Projekt apletu może być w każdej chwili zapisany (a później otwarty). Zapisanie projektu (funkcja zapisz w menu plik) powoduje utworzenie katalogu o nazwie odpowiadającej nazwie projektu i zapisanie w nim trzech plików:

- nazwa_projektu.html – plik inicjujący aplet,
- nazwa_projektu.xml – parametry apletu w formacie programu,
- application.xml – parametry edytora związane z konkretnym projektem (adresy ftp, http i UPWD).

Format i znaczenie tych plików zostały dokładniej omówione w rozdziale czwartym.

3.6 Klient FTP

Wbudowany klient FTP umożliwia wysyłanie konfiguracji apletu bezpośrednio do serwera www lub dowolnego innego serwera FTP. Jest dostępny poprzez funkcję „Wyślij przez FTP” w menu „Aplet”. Po jej wybraniu pojawia się przedstawione na rysunku 3.7 okno dialogowe pozwalające określić parametry połączenia.



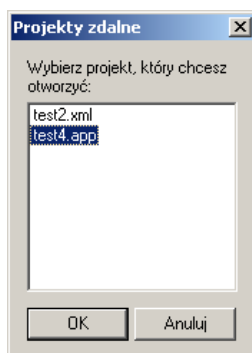
Rys. 3.7. Dialog klienta FTP.

Ono dialogowe domyślnie wykorzystuje wartości zdefiniowane przy tworzeniu nowego projektu.

3.7 Projekty zdalne

Edytor stron WWW umożliwia odczyt projektów z sieci przy użyciu protokołu FTP. Po wybraniu z menu „Plik” opcji „Otwórz zdalny projekt” i podaniu adresu serwera program pobierze z sieci i wyświetli listę dostępnych pod wskazanym adresem projektów (plików app – patrz rozdział czwarty). Wybrany plik zostanie pobrany z sieci i otwarty w aplikacji.

Dzięki temu możliwe jest zmodyfikowanie tablicy synoptycznej bez posiadania lokalnej kopii plików projektu.



Rys. 3.8. Okno wyboru projektów zdalnych.

3.8 Konfiguracja

Użytkownik może zmodyfikować niezwiązane bezpośrednio z drzewem urządzeń korzystając z „Opcji” w menu „Konfiguracja”. Pozwalają one na zmianę nazwy projektu oraz parametrów połączeń FTP i UPWD.

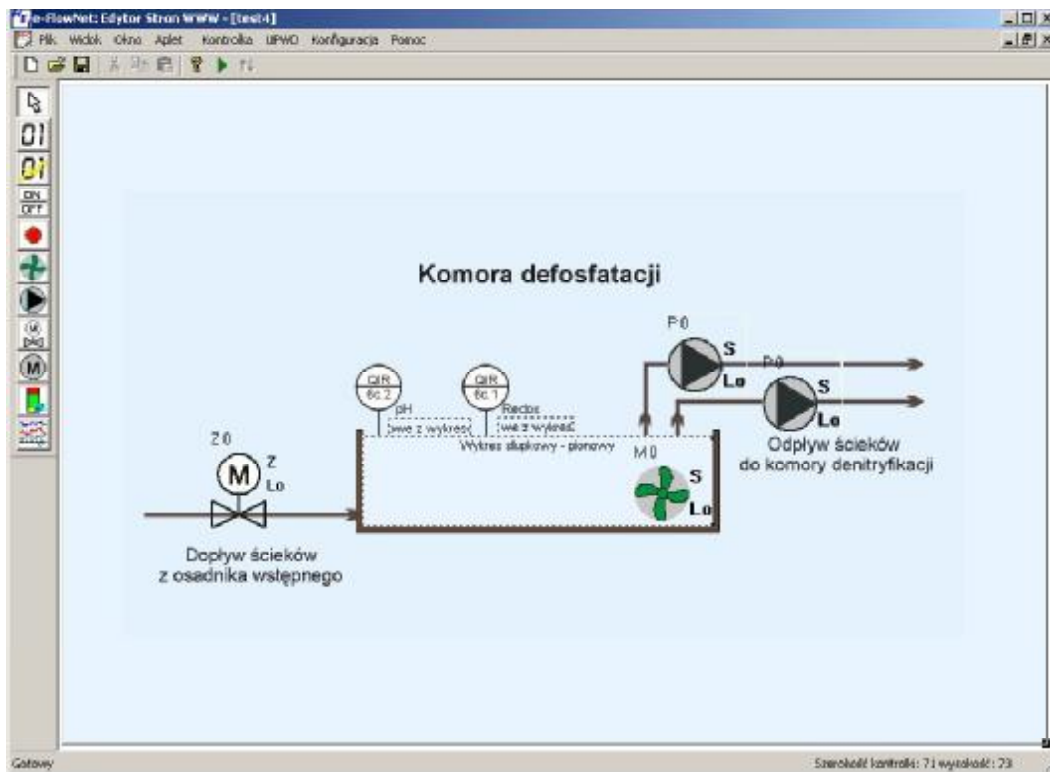
Dodatkowo z menu „Konfiguracja” można przejść do edycji plików konfiguracyjnych aplikacji (opisanych dokładnie w rozdziale czwartym).

3.9 Podgląd apletu.

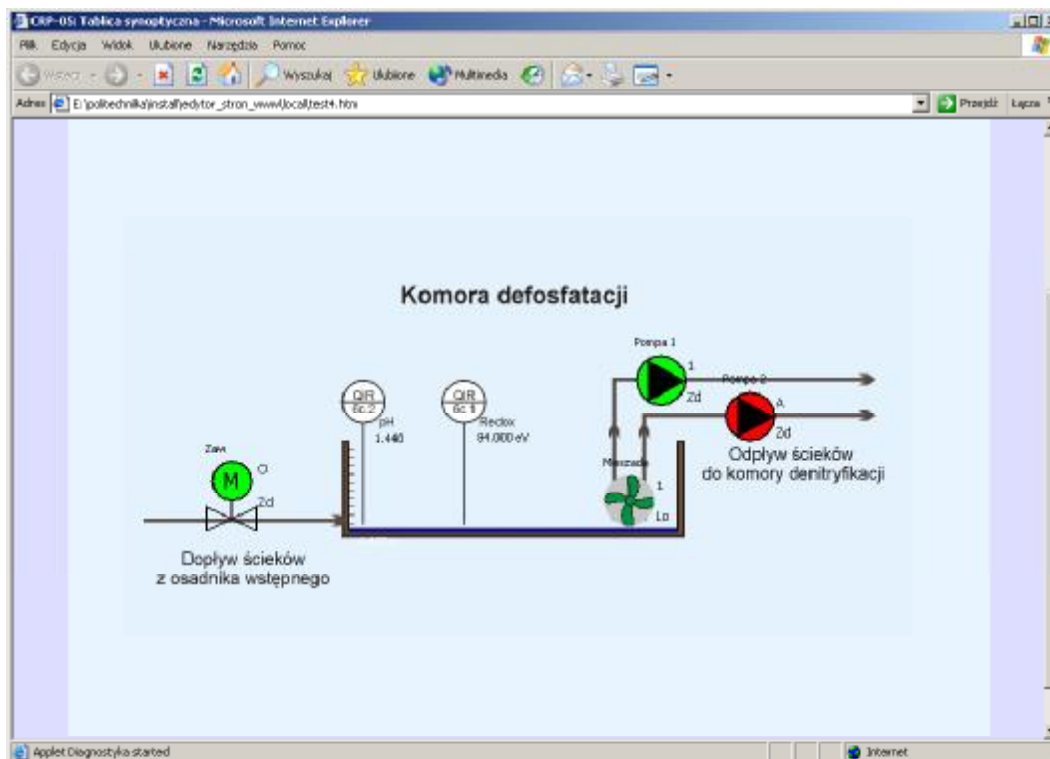
W pewnych przypadkach przydatna jest możliwość sprawdzenia, jak zachowuje się aplet działający w oparciu o przygotowaną konfigurację. Pozwala to na przetestowanie parametrów niezwiązanych bezpośrednio z wyglądem apletu ale istotnych dla jego pracy (np.

powiązania pomiarów z urządzeniami). Jest to możliwe dzięki funkcji podglądu, obecnej w programie w dwóch wersjach.

Podgląd zdalny wysyła stronę na serwer używając klienta FTP a następnie otwiera ją, natomiast podgląd lokalny wykorzystuje pliku obecnego na komputerze. Ten drugi sposób ma duże znaczenie dla testów i prezentacji, bowiem eliminuje konieczność przygotowania serwera FTP. Niestety, do poprawnego działania podgląd lokalny wymaga maszyny wirtualnej firmy Sun – we wszystkich innych przypadkach wystarcza maszyna Javy firmy Microsoft domyślnie obecna w większości wersji przeglądarki Internet Explorer (więcej informacji na ten temat w Dodatku A).



Rys. 3.9. Projekt apletu w edytorze.



Rys. 3.10. Aplet działający w oparciu o przygotowany projekt.

Na rysunkach 3.9 i 3.10 przedstawiony jest ten sam aplet – za pierwszym razem jako projekt w edytorze, a za drugim już jako uruchomiony w przeglądarce program (wykorzystano funkcję podglądu lokalnego). Warto zwrócić uwagę na zainicjowane pomiary

– np. kontrolki w projekcie oznaczone „wskazanie cyfrowe”, w działającym aplecie prezentują cyfrowe pomiary. W przypadku pomp, zielony kolor zaworu i jednej z pomp oznacza poprawną pracę urządzenia, natomiast czerwony (widoczny na drugiej pompie) sygnalizuje awarię.

W Dodatku B opisano sposób konfiguracji środowiska e-FlowNet umożliwiający testowanie systemu na pojedynczym komputerze.

4. Formaty plików

Zarówno pliki wejściowe, jak i wyjściowe mają swoje miejsce w strukturze katalogów programu. Dlatego do pełnego zrozumienia ich roli i sposobów wykorzystania przydatna jest znajomość tej struktury.

Tabela 4.1 przedstawia podkatalogi katalogu programu.

Katalog	Zawarte pliki
config	- pliki konfiguracyjne a także ikony kontrolek wykorzystywane w programie
help	- pliki pomocy w formacie html
images	- pliki graficzne wykorzystywane do projektowania apletów – m.in. zbiór bitmap tła dla apletu
local	- pliki wykorzystywane przy lokalnym podglądzie apletu – plik html, bitmapy a także plik jar zawierający kod apletu
temp	- pliki tymczasowe – np.: pobrane z sieci projekty zdalne
templates	- wzorce HTML

Aby rozpocząć pracę program potrzebuje danych zawartych w dwóch plikach wejściowych: „application.xml” i „controls.xml”. Podobnie jak wykorzystywane w programie bitmapy (np. ikony kontrolek) znajdują się one w podkatalogu config. Pierwszy z nich zawiera parametry aplikacji – przede wszystkim domyślne wartości wykorzystywane przy tworzeniu nowych projektów. Drugi (najważniejszy), zawiera definicje parametrów apletu i kontrolek dostępnych w trakcie pracy z programem. Na jego podstawie przy starcie dynamicznie tworzony jest pasek narzędzi z kontrolkami.

Poza wspomnianymi plikami xml, do poprawnego zapisania parametrów apletu wymagana jest obecność przynajmniej jednego pliku zawierającego wzorzec html w podkatalogu templates.

Będący efektem działania programu projekt apletu zapisywany jest w dwóch formatach: html i xml. Plik html, główny efekt pracy programu, to strona internetowa zawierająca aplet (definicje jego parametrów). Natomiast plik xml zapisywany jest na użytek programu i pozwala na późniejszą modyfikację projektu – jest dokumentem który aplikacja może odczytywać i zapisywać.

4.1 Plik „application.xml”

Przykładowy plik „application.xml” wygląda następująco:

```
<?xml version="1.0" encoding="windows-1250"?>
<!DOCTYPE application SYSTEM "application.dtd">
<application version="1.0">
  <ftp.host>157.158.1.3</ftp.host>
  <ftp.directory>./public_html</ftp.directory>
  <ftp.user>kubad</ftp.user>
  <preview.page>http://157.158.1.3</preview.page>
  <upwd.host>157.158.11.3</upwd.host>
  <upwd.port>9001</upwd.port>
  <html.template>template.htm</html.template>
  <html.editor>notepad.exe</html.editor>
  <html.browser>opera.exe</html.browser>
</application>
```

Większość zawartych w nim parametrów to domyślne wartości inicjujące tworzony projekt:

ftp.host – nazwa serwera ftp dla nowego projektu

ftp.directory – katalog ftp dla nowego projektu

ftp.user – nazwa użytkownika ftp

preview.page – adres http strony podglądu

upwd.host – adres serwera upwd

upwd.port – port serwera upwd

html.template – wzorzec strony wykorzystywany do zapisu parametrów apletu

html.browser – nazwa przeglądarki wykorzystywanej przez program (m.in. do podglądu apletów)

html.editor – określa, jakiego edytora ma używać program do edycji wzorców html

Ze względów bezpieczeństwa program nie przechowuje haseł ftp.

4.2 Plik „controls.xml”

Plik ten definiuje aplet i kontrolki dostępne w trakcie pracy z programem. Zarówno w pierwszym, jak i drugim przypadku definicja to lista parametrów reprezentowanych przez znaczniki i atrybuty xml:

```
<controls>
  <applet>
    <!-- lista parametrów -->
  </applet>
  <control>
    <!-- lista parametrów -->
  </control>
  <!--...
  (więcej definicji kontrolek)
  ...-->
</controls>
```

4.2.1 Parametry

Wszystkie parametry apletu i kontrolek definiowane są w jednolity sposób, tak by uprościć odczyt plików oraz dodawanie lub usuwanie parametrów. Gramatyka DTD opisująca postać znacznika param wygląda następująco:

```
<!ELEMENT param (name | default.value | file.type |
file.extension | values) +>
<!ATTLIST param type (text | int | color | file | choice |
float | register | readonly | hidden | measure) #REQUIRED>
<!ATTLIST param visibility (visible|hidden) #IMPLIED>
<!ELEMENT name (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT default.value (#PCDATA)>
<!ELEMENT file.type (#PCDATA)>
<!ELEMENT file.extension (#PCDATA)>
<!ELEMENT values (value+)>
<!ELEMENT value (#PCDATA)>
```

Jak widać, znacznik param posiada obowiązkowy (informuje o tym #REQUIRED) atrybut type informujący o typie definiowanego parametru. Informacja o typie parametru pozwala programowi na kontrolę wprowadzonych przez użytkownika wartości (np. zakresu liczb typu integer), a także na odpowiednią reprezentację typów parametrów w listach właściwości. Dopuszczalne typy parametrów przedstawia tabela 4.2.

Dopuszczalne typy parametrów.

Tabela 4.2

Typ parametru	Opis
text	Dane tekstowe (np. nazwa urządzenia).
int	Liczba typu integer.
float	Liczba rzeczywista typu float.
color	Definicja koloru (np. tła apletu). Parametry tego typu są reprezentowane w listach właściwości jako dialog wyboru koloru.
file	<p>Plik (np. bitmapa tła). Dla parametrów typu file wewnątrz znacznika param mogą pojawić się znaczniki file.type i file.extension definiujące odpowiedni słowny opis typu pliku (np.: „Bitmapy”) i dopuszczalne rozszerzenia pliku (np. „*.jpg; *.gif; *.png”)</p> <p>Parametry tego typu są reprezentowane w listach właściwości jako dialog wyboru pliku zainicjowany wartościami file.type i file.extension.</p>
choice	<p>Wartość parametru może być wybrana spośród jednej z kilku podanych wartości – np. dla parametru „typ magistrali” dopuszczalne wartości to:</p> <p>RS-232 RS-485</p> <p>Wewnątrz znacznika param dopuszczalne wartości dla typu choice umieszczone są wewnątrz znacznika <values></p> <pre><values> <value>RS-232</value> <value>RS-485</value> ... </values></pre> <p>Parametr tego typu jest reprezentowany w listach właściwości jako pole wyboru „tzw. „combo box”).</p>
register	Wartością parametru może być nazwa rejestru zapisywalnego urządzenia. W listach właściwości reprezentowane są w dwojaki sposób – jeśli użytkownik wcześniej przy pomocy protokołu upwd pobrał z serwera drzewo pomiarów, wyświetlany jest dialog drzewa pomiarów, a w przeciwnym przypadku pole tekstowe.
measure	Zastosowanie tego typu jest bardzo podobne (można powiedzieć symetryczne) do typu register. Wartością parametru może być nazwa pomiaru udostępnianego przez urządzenie. Parametry tego typu pozwalają na powiązanie kontrolki wizualizujących dane z pomiarami udostępnianymi przez urządzenie. Podobnie jak dla typu register w listach właściwości reprezentowane są jako drzewo pomiarów lub pole tekstowe.
readonly	Ten typ definiuje parametry tylko do odczytu – są widoczne w listach właściwości, ale użytkownik nie może zmienić ich wartości (np. opis kontrolki)
hidden	Ten typ definiuje parametry niewidoczne w listach właściwości.

Znaczniki dopuszczalne wewnątrz znacznika param przedstawia tabela 4.3.

Znaczniki wewnątrz <param>.

Tabela 4.3

Znacznik	Opis
name	Nazwa parametru.
description	Opis parametru.
default.value	Domyślna wartość.
file.type	Typ pliku (tylko dla parametrów typu file).
file.extension	Rozszerzenie pliku (tylko dla parametrów typu file).
values	Lista wartości parametru – tylko dla typu choice. Wewnątrz tego znacznika znajduje się jeden lub więcej znaczników <value> definiujących dopuszczalne wartości parametru.

Przyjęty format zapisu parametrów może wydać się nadmiarowy, jednak dzięki temu pliki mogą być stosunkowo prosto odczytane przez człowieka – można je nazwać „samodokumentującymi”.

4.2.2 Atrybuty

W kilku przypadkach dla przedstawienia właściwości apletu i kontrolki zamiast opisanego powyżej znacznika param zastosowano atrybuty xml. Wynika to po pierwsze z faktu, że pewne właściwości znaczeniowo lepiej pasują do pojęcia atrybutu, a po drugie, z konieczności by pewne wartości (np. rozmiar) były znane zanim nastąpi odczyt pozostałych parametrów.

Dla apletu domyślną szerokość i wysokość zdefiniowano w następujący sposób:

```
<applet width="550" height="350">
```

co w pliku DTD zdefiniowano następująco:

```
<!ATTLIST applet width CDATA #IMPLIED>  
<!ATTLIST applet height CDATA #IMPLIED>
```

Słowo kluczowe „#IMPLIED” oznacza, że w przypadku braku atrybutu program powinien dostarczyć swoją wartość.

Dla kontrolki plik DTD definiuje następujące atrybuty:

```
<!ATTLIST control icon CDATA #IMPLIED>  
<!ATTLIST control width CDATA #IMPLIED>  
<!ATTLIST control height CDATA #IMPLIED>  
<!ATTLIST control border (0|1) #IMPLIED>  
<!ATTLIST control typeinfo (0|1) #IMPLIED>  
<!ATTLIST control size (fixed|scalable) #IMPLIED>
```

Znaczenie tych atrybutów jest następujące:

- icon - ikona reprezentująca kontrolkę w pasku narzędzi (kontrolki muszą być plikami png rozmiarze 24x24)
- width - szerokość kontrolki
- height - wysokość kontrolki
- border - informacja, czy kontrolka ma być wyświetlana z obwódką czy bez niej - może przyjmować wartości „0” lub „1” (domyślna to „1” - granica jest rysowana)
- type.info - informacja, czy na kontrolce ma być wyświetlana informacja o jej typie - może przyjmować wartości „0” lub „1” (domyślna to „1” - informacja jest wyświetlana)
- size - informacja czy użytkownik może zmieniać wielkość kontrolki - może przyjmować wartości „fixed” lub „scalable” (domyślna to fixed - nie można zmienić wielkości kontrolki); dla kontroltek „fixed” nie są wyświetlane węzły umożliwiające zmianę wielkości

4.3 Wzorce html

Wzorce html to pliki do których zapisywane są parametry apletu. Plik wzorca kopiowany jest do katalogu projektu, przy czym umieszczony w nim (mający postać komentarza) znacznik `<!--applet mark>` zastępowany jest definicją apletu – więcej informacji znajduje się w opisie danych wyjściowych.

Wzorce opisują stronę html zawierającą aplet – definiują „dekorację” apletu. W ten sposób możliwa jest szybka i prosta zmiana wyglądu stron – np. dostosowanie ich do potrzeb klienta.

Prosty plik wzorca może wyglądać następująco:

```
<html>
  <head><title>Pomiary oczyszczalni</title></head>
  <body>

  Pomiary oczyszczalni
  <!--applet mark-->
  <!--powyzszy komentarz zostanie zastapiony definicja
  apletu-->

  </body>
</html>
```

Jeśli zdefiniowany zostanie drugi plik wzorca:

```
<html>
  <head><title>HEI</title></head>
  <body>

  Pomiary Hydro - Eco - Invest
  <!--applet mark-->
  <!--powyzszy komentarz zostanie zastapiony definicja
  apletu-->

  </body>
</html>
```

poprzez zmianę aktywnego wzorca projektu można błyskawicznie zmienić wygląd wynikowej strony html.

Dodatkowo pliki wzorca zazwyczaj zawierają kod w języku javascript sprawdzający obecność maszyny wirtualnej Javy na komputerze. W przypadku, gdy jest ona nieobecna lub wyłączona (ustawienie poziomu bezpieczeństwa przeglądarki jako „wysoki” zwykle wyłącza maszynę wirtualną – patrz Dodatek A), następuje przekierowanie na stronę z instrukcją instalacji maszyny wirtualnej, inną dla każdej przeglądarki.

4.4 Plik projektu w formacie xml

Pliki wyjściowe xml zawierają zapis projektu apletu który edytor potrafi później odczytać. Definicja apletu zapisywana jest w podkatalogu projektu dwóch plikach: nazwa_projektu.xml oraz „application.xml”.

Pierwszy z nich zawiera parametry apletu i umieszczonych na nim kontrolki. Można pokusić się o stwierdzenie, że jeśli plik wejściowy controls.xml definiuje klasy kontrolki apletu, to wyjściowy plik projektu stanowi zapis obiektów tych klas.

Format zapisu, będący uproszczoną i nieco zmodyfikowaną formą stosowane w pliku „controls.xml” jest następujący:

```
<controls>
  <applet>
    <!-- lista parametrów -->
  </applet>
  <control type=typ_kontrolki>
    <!-- lista parametrów -->
  </control>
</controls>
```

Atrybut type znacznika control stanowi powiązanie między definicją kontrolki (obiektem) a definicją typu kontrolki (klasą). Jeśli plik „controls.xml” definiuje typ o nazwie „zawór”, kontrolka tego typu rozpoczynałaby się od znacznika <control type=„zawór”> (możliwe jest używanie „polskich znaków”).

Parametry zapisywane są w następujący sposób:

```
<param>
  <name>nazwa parametru</name>
  <value>wartość parametru</value>
</param>
```

Jeśli kontrolka poprzez atrybut type zostanie skojarzona z typem kontrolki, jej parametry będą poprzez nazwę kojarzone z typami parametrów.

Przykładowo, jeśli w pliku controls.xml znajduje się (pomijając elementy nieistotne dla przykładu) wpis:

```
<control>
  <name>wskazanie cyfrowe</name>
  <param type=„choice”>
    <name>proto</name>
```

```

        <default.value>RS-485-Modbus-RTU</default.value>
        <values>
            <value>RS-485-Modbus-RTU</value>
            <value>RS-232</value>
        </values>
    </param>
</control>

```

a w pliku projektu wpis:

```

<control type="wskazanie cyfrowe">
<param>
    <name>proto</name>
    <value>RS-485-Modbus-RTU</value>
</param>
</control>

```

to po odczytaniu pliku projektu program rozpozna, parametr „proto”, jako parametr typu choice o dopuszczalnych wartościach „RS-485-Modbus-RTU” i „RS-232”.

Drugi z plików projektu, „application.xml”, ma format identyczny, jak plik o tej samej nazwie przechowywany w podkatalogu config (z wyjątkiem html.editor i html.browser, których wartości nie można zmieniać) i przechowuje takie same parametry:

ftp.host – nazwa serwera ftp dla nowego projektu,
ftp.directory – katalog ftp dla nowego projektu,
ftp.user – nazwa użytkownika ftp,
preview.page – adres http strony podglądu,
upwd.host - adres serwera upwd,
upwd.port – port serwera upwd,
html.template – wzorzec strony wykorzystywany do zapisu parametrów apletu.

Tak więc kiedy tworzony jest nowy projekt, jego parametry inicjowane są wartościami pliku wejściowego, które użytkownik może zmienić. Przy zapisie trafiają one do pliku w podkatalogu projektu. Dzięki temu każdy projekt może posiadać własny zestaw adresów sieciowych (ftp, upwd, http) z którymi jest związany, a równocześnie użytkownik nie musi ich pamiętać i wpisywać za każdym razem.

4.5 Plik html inicjujący aplet

Jest to wzorzec html, do którego zapisano definicję apletu. Zawiera te same informacje co plik xml opisujący projekt apletu, zapisane zgodnie ze specyfikacją html, a więc jako znacznik aplet i zbiór znaczników param.

Tak więc parametry apletu zapisane w języku xml:

```

<project>
    <applet>
        <param>
            <name>name</name>
            <value>test</value>

```

```

    </param>
    <param>
      <name>width</name>
      <value>770</value>
    </param>
    <param>
      <name>height</name>
      <value>494</value>
    </param>
    <param>
      <name>backX</name>
      <value>5</value>
    </param>
    <param>
      <name>code</name>
      <value>Diagnostyka.class</value>
    </param>
    <param>
      <name>archive</name>
      <value>synopt.jar</value>
    </param>
    <param>
      <name>datasourceip</name>
      <value>127.0.0.1</value>
    </param>
    <param>
      <name>datasourceport</name>
      <value>1111</value>
    </param>
    <param>
      <name>obrazTla</name>
      <value>.\test\218.gif</value>
    </param>
    ...
  </applet>
</project>

```

zapisane w języku html wyglądają następująco:

```

<Applet id="test" style="WIDTH: 770px; HEIGHT: 494px"
code="Diagnostyka.class"
width="770" height="494" MAYSCRIPT VIEWASTEXT
archive="synopt.jar">
<PARAM NAME="code" VALUE="Diagnostyka.class">
<PARAM NAME="archive" VALUE="synopt.jar">
<PARAM NAME="datasourceip" VALUE="127.0.0.1">
<PARAM NAME="datasourceport" VALUE="1111">
<PARAM NAME="obrazTla" VALUE="218.gif">
...

```

Dodatek A – Instalacja i konfiguracja środowiska Javy w przeglądarce Internet Explorer

A.1 – Poziom bezpieczeństwa

W wersjach 4 i 5 IE maszyna wirtualna javy (w wersji firmy Microsoft) jest standardowo obecna i włączona. Jeśli jednak poziom bezpieczeństwa przeglądarki został ustawiony jako „wysoki”, obsługa Javy zostaje wyłączona. Aby ją włączyć należy wybrać z menu „Narzędzia” pozycję „Opcje internetowe”. Następnie trzeba kliknąć na zakładkę zabezpieczenia, zaznaczyć grupę „Internet” i nacisnąć przycisk „Poziom niestandardowy”. Wyświetlona zostanie lista opcji związanych z bezpieczeństwem przeglądarki. Należy odszukać pozycję „Microsoft VM” i wybrać opcję inną niż „Wyłącz Java” – np. „Wysokie bezpieczeństwo”.

A.2 Instalacja maszyny wirtualnej firmy Sun.

W wersji 6 przeglądarki Internet Explorer maszyna wirtualna Javy jest nieobecna. Natomiast wersja dołączana do wcześniejszych wydań (wersje 4 i 5) nie obsługuje poprawnie funkcji lokalnego podglądu przygotowywanych apletów (nie potrafi połączyć się z adresem 127.0.0.1). W takich przypadkach należy samodzielnie zainstalować maszynę wirtualną javy, najlepiej w wersji firmy Sun. Jest ona dołączona do płyty e-FlowNet (w katalogu j2re1.4.1) lub pobrać ze strony <http://www.javasoft.com>.

Dodatek B – przygotowania środowiska testowego systemu e-FlowNet

W przypadku testów systemu, bardzo przydatna jest możliwość zestawienia środowiska testowego w oparciu o programowy emulator serwera komunikacyjnego HI-02.

Program ten jest dołączony do serwera e-FlowNet (obecnego na płycie z oprogramowaniem) i domyślnie instalowany razem z nim, tak więc pierwszym krokiem jest instalacja serwera e-FlowNet.



Rys. B.1. Instalacja serwera e-FlowNet

Domyślna konfiguracja serwera e-FlowNet jest przygotowana do współpracy właśnie z emulatorem HI-02 – adres źródła danych (serwera komunikacyjnego) ustawiono na 127.0.0.1 (adres lokalny), port 9001. Natomiast sam serwer e-FlowNet jest oczywiście dostępny pod adresem komputera na którym został zainstalowany (w przypadku środowiska testowego można założyć 127.0.0.1) i wykorzystuje port 9005.

Po zainstalowaniu należy najpierw uruchomić (korzystając ze skrótów w menu Start) Emulator HI-02 a następnie serwer e-FlowNet. Prawidłowo zainstalowany serwer powinien połączyć się z Emulatorem HI-02 a następnie pobrać i wyświetlić drzewo urządzeń. Od tej chwili oprogramowanie klienckie wykorzystujące protokół UPWD (*Edytor stron WWW* i *Oprogramowanie wizualizacyjne e-FlowNet*) może łączyć się z serwer e-FlowNet (127.0.0.1:9005) lub bezpośrednio z Emulatorem HI-02 (adres 127.0.0.1:9001). Serwer OPC ma własny system testowania (emulujący HI-02), wbudowany w wersję demo.

Dodatek C – Testowy projekt apletu

Przygotowanie testowego projektu w programie Edytor Stron WWW

Aby móc przetestować aplet wizualizacji w środowisku testowym należy najpierw przygotować je zgodnie z opisem zamieszczonym w dodatku B.

W standardowej instalacji *Edytora stron WWW* dostępne są dwa testowe projekty apletów (o nazwach „testowy” i „testowy_2”), gotowe do uruchomienia w programie. Poniżej krok po kroku opisana została procedura przygotowania apletu odpowiadającemu projektowi „testowy”.

C.1 Instalacja i uruchomienie programu „Konfigurator urządzeń”

Należy wybrać opcję „Konfigurator „e-FlowNet” w menu płyty lub ręcznie uruchomić (poprzez dwukrotne kliknięcie) znajdujący się w katalogu „Konfigurator e-FlowNet” program setup.exe. Po pojawieniu się okna programu instalacyjnego nie trzeba zmieniać żadnych opcji, wystarczy klikać na przycisk „Next” na kolejnych ekranach. Po zainstalowaniu program można uruchomić poprzez wybranie „Konfigurator e-FlowNet” w katalogu „Programy” menu startowego. Aby móc testować komunikację (pobieranie informacji o dostępnych urządzeniach) oraz pracę przygotowanego apletu, należy również uruchomić środowisko testowe – najpierw emulator serwera komunikacyjnego a następnie serwer e-FlowNet. Są one dostępne poprzez pozycję Serwer e-FlowNet menu startowego.

C.2 Utworzenie nowego projektu apletu

C.3 Rozmieszczenie kontroltek

C.4 Pobranie z serwera informacji o dostępnych pomiarach i rejestrach

C.5 Edycja parametrów kontroltek

C.6 Uruchomienie gotowego projektu