

# Konfigurator urządzeń

Jakub Duda,  
Hydro-Eco-Invest, Gliwice

Dokumentacja użytkownika

<b>1.</b>	<b>Wprowadzenie</b>	<b>3</b>
<b>2.</b>	<b>Zastosowania i możliwości programu</b>	<b>3</b>
<b>3.</b>	<b>Praca z programem</b>	<b>4</b>
3.1	Sposób instalacji i uruchomienia	4
3.2	Tworzenie nowego projektu	4
3.3	Dodawanie urządzeń	5
3.4	Przeglądanie i edycja parametrów	6
3.5	Dodawanie i usuwanie rejestrów	7
3.6	Zapis projektu	8
3.7	Klient FTP	8
3.8	Projekty zdalne	9
3.9	Konfiguracja	9
<b>4.</b>	<b>Formaty plików konfiguracyjnych</b>	<b>10</b>
4.1	Plik application.xml	10
4.1	Plik device.xml	10
4.3	Pliki projektów	14

## **1. Wprowadzenie**

Program *Konfigurator urządzeń* jest elementem systemu e-FlowNet firmy Hydro-Eco-Invest. Jego zadaniem jest kompleksowe wspomaganie tworzenia interfejsów urządzeń dla serwera komunikacyjnego HI-02 – począwszy od wyboru urządzeń i ich rejestrów, poprzez konfigurację parametrów, aż do wysłania plików konfiguracyjnych do serwera HI-02. Intuicyjny i elastyczny interfejs umożliwi użytkownikowi szybkie przygotowanie interfejsów, bez konieczności pisania programów czy też żmudnej, ręcznej edycji plików konfiguracyjnych. Ponadto praca w trybie graficznym znacząco przyspiesza i upraszcza proces konfiguracji serwera komunikacyjnego i eliminuje wiele potencjalnych błędów.

Jako format przechowywania danych o interfejsach urządzeń wykorzystywany jest będący otwartym standardem język xml.

W dalszej części dokumentacji omówione zostaną kolejno zadania i możliwości programu (rozdział drugi), sposób instalacji, uruchomienia oraz pracy z programem (rozdział trzeci) i wreszcie pliki konfiguracyjne (rozdział czwarty).

## **2. Zastosowania i możliwości programu**

Jak wspomniano we wstępie, zadaniem programu jest łatwe i szybkie przygotowanie interfejsów urządzeń dla serwera komunikacyjnego HI-02. Zestaw interfejsów (nazywany dalej „drzewem urządzeń”) jest warstwą pośredniczącą między „przemysłową” częścią serwera (przyłączonymi do niego urządzeniami) a jego klientami (serwerem e-FlowNet lub programami wizualizacji i sterowania), komunikującymi się przy użyciu protokołu TCP/IP. Zawiera informację w jaki sposób połączone z serwerem urządzenia mają być widoczne dla klientów. Należy przy tym pamiętać że dodanie urządzenia do drzewa w programie oznacza tak naprawdę zdefiniowane interfejsu do niego, a wszystkie interfejsy muszą odpowiadać rzeczywistym urządzeniom i ich rejestrom.

Konfigurator urządzeń daje użytkownikowi dużą swobodę manipulacji danymi – dodawanie i usuwanie urządzeń, ich rejestrów, zmianę wartości parametrów, powielanie skonfigurowanych urządzeń przy pomocy schowka, itp.

Przygotowane drzewo urządzeń można zapisać na dysku (a więc w dowolnym momencie powrócić do niego i wprowadzić modyfikacje). Można również przy pomocy wbudowanego klienta FTP wysłać informacje do serwera komunikacyjnego, lub dowolnego innego serwera FTP. W przypadku wysłania danych do HI-02 nowa konfiguracja jest automatycznie stosowana przez serwer.

*Konfigurator urządzeń* jest aplikacją pracującą 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 *Konfigurator urządzeń* wykorzystuje standardowy program instalacyjny InstallShield. Aby rozpocząć instalację należy uruchomić (poprzez dwukrotne kliknięcie) znajdujący się w katalogu aplikacji 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 programu można uruchomić poprzez wybranie „Konfigurator e-FlowNet” w katalogu Programy menu startowego.

#### 3.2 Tworzenie nowego projektu

Aby utworzyć nowy projekt, należy zaznaczyć pozycję „Nowy” w menu „Plik” i wybrać opcję „Konfiguracja urządzenia” jako typ dokumentu. Wówczas pojawia się okno dialogowe pozwalające na podanie początkowych parametrów projektu.

Konfiguracja urządzenia		Połączenia FTP	
Nazwa		Serwer	157.158.13.187
		Katalog	./
		Użytkownik	hei
UPWD			
Adres UPWD	157.158.13.187	Port	9001
Edycja			
<input checked="" type="checkbox"/> Potwierdź dodawanie nowych urządzeń			
OK		Anuluj	

Rys. 3.1. Okno dialogowe z parametrami nowego projektu.

Widoczne na rysunku 3.1 parametry przykładowego projektu 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 projektu. Nazwa projektu określa nazwy plików projektu oraz katalogu w którym zostaną zapisane, dlatego istotne jest by była poprawna w rozumieniu systemu plików.

Z kolei ramka „Połączenia FTP” pozwala na skonfigurowanie adresu, katalogu i użytkownika serwera FTP na który zostanie wysłana konfiguracja urządzenia (tj. pliki dev lub bdv).

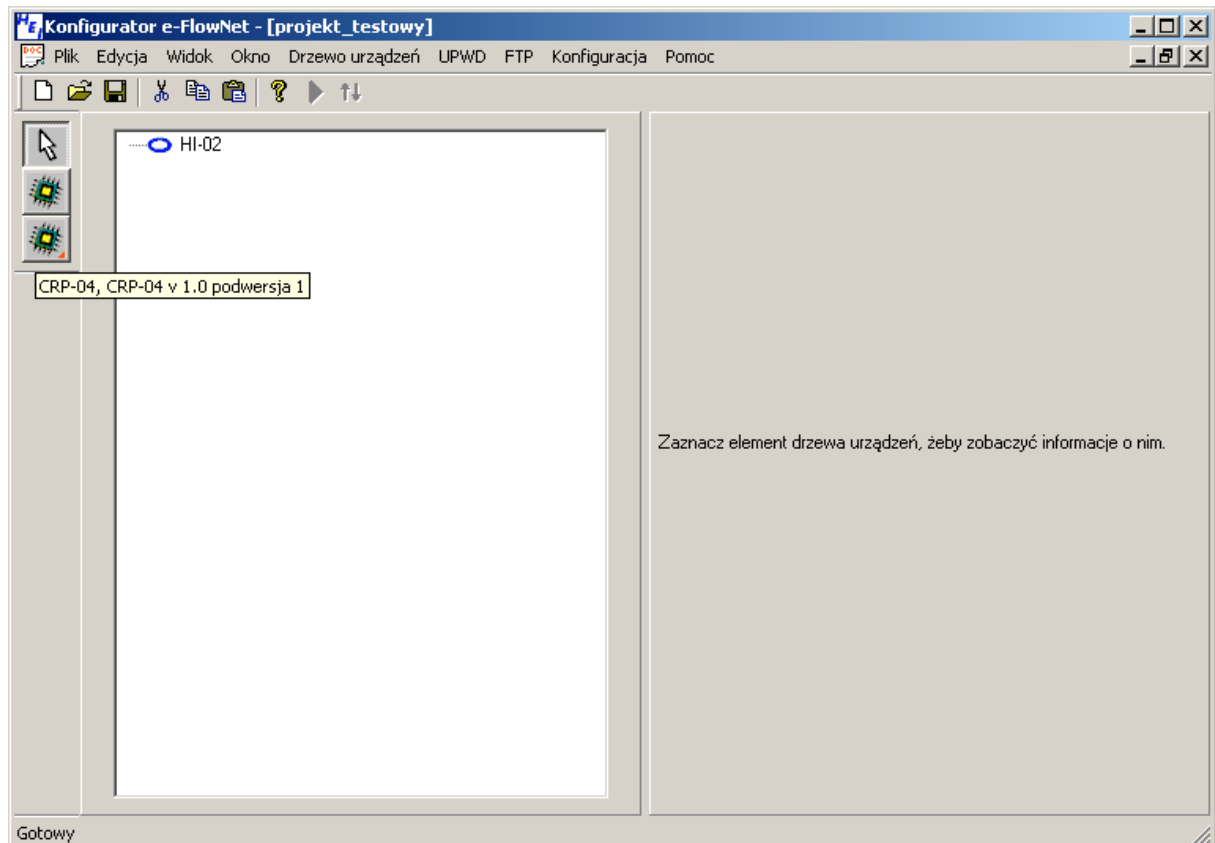
Adres i port serwera UPWD (Uniwersalny Protokół Wymiany Danych – rozwiązanie firmy Hydro-Eco-Invest) określają adres serwera HI-02 wykorzystywany przy połączeniach UPWD informujących serwer o nowej konfiguracji. Zazwyczaj adres UPWD jest taki sam jak adres FTP.

Opcja „Potwierdź dodawanie nowych urządzeń” decyduje o tym, czy program będzie pytał użytkownika przy każdej próbie dodania urządzenia, uniemożliwiając zrobienie tego przypadkowo.

Jeśli nazwa projektu została podana poprawnie, po naciśnięciu OK. zostanie utworzony nowy projekt.

### 3.3 Dodawanie urządzeń

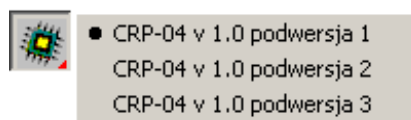
Po utworzeniu nowego projektu kolejnym krokiem jest dodanie do niego urządzeń.



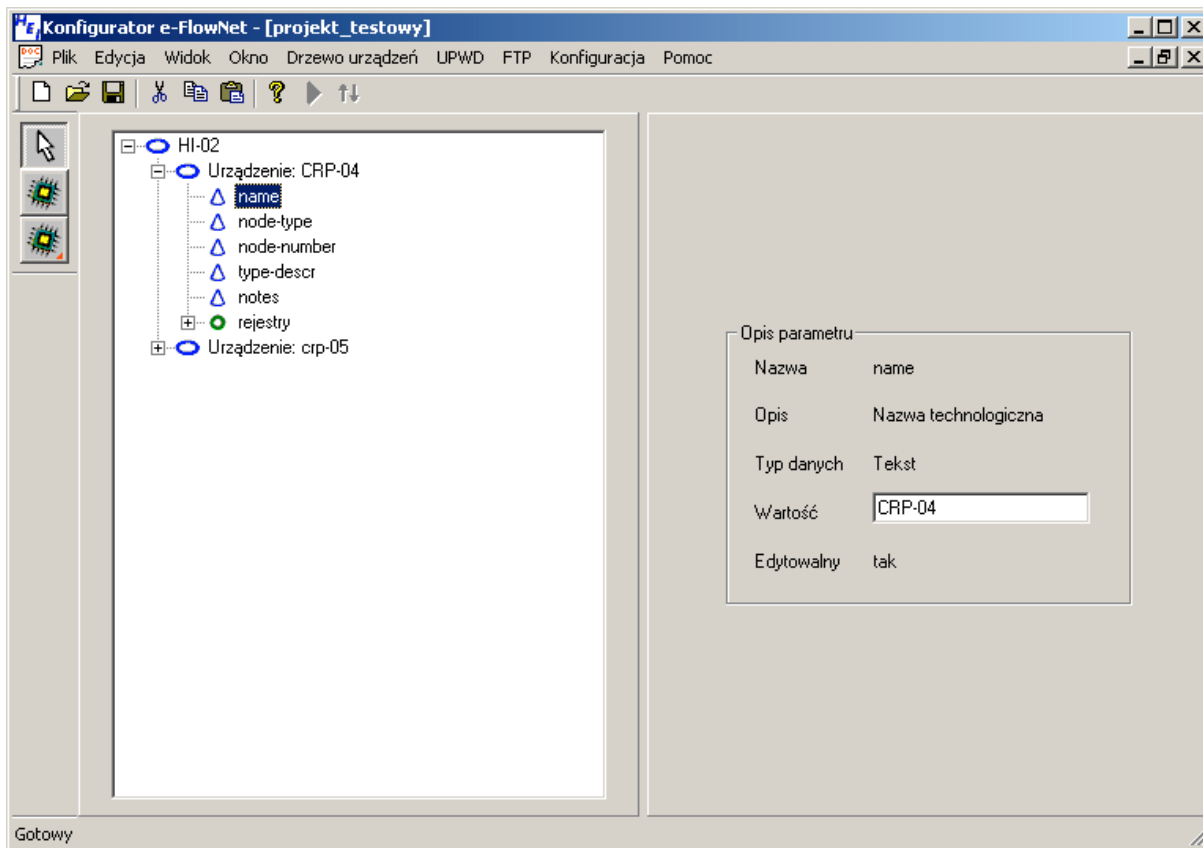
Rys. 3.2. Nowo utworzony projekt.

Na rysunku 3.2 widoczny jest nowoutworzony projekt konfiguracji urządzenia. Jak widać okno aplikacji podzielone jest na dwa panele. Lewy przedstawia drzewo urządzeń (w nowym projekcie jest to „pusty” serwer HI-02) a prawy jest wykorzystywany do przeglądania i edycji parametrów.

Wszystkie dostępne urządzenia wyświetlane są na pasku narzędzi po lewej stronie ekranu. Gdy kursor znajdzie się nad ikoną wyświetlana jest informacja o typie pomiaru. Aby dodać urządzenie do projektu należy zaznaczyć symbolizującą go ikonę a następnie kliknąć na lewy panel (poza dodanymi wcześniej urządzeniami). Trzeba pamiętać, że niektóre urządzenia posiadają podwersje o różnych parametrach (np. przystosowane do pracy z magistralą RS-232 lub RS-485). Urządzenia takie zaznaczone są czerwoną strzałką w rogu ikony.







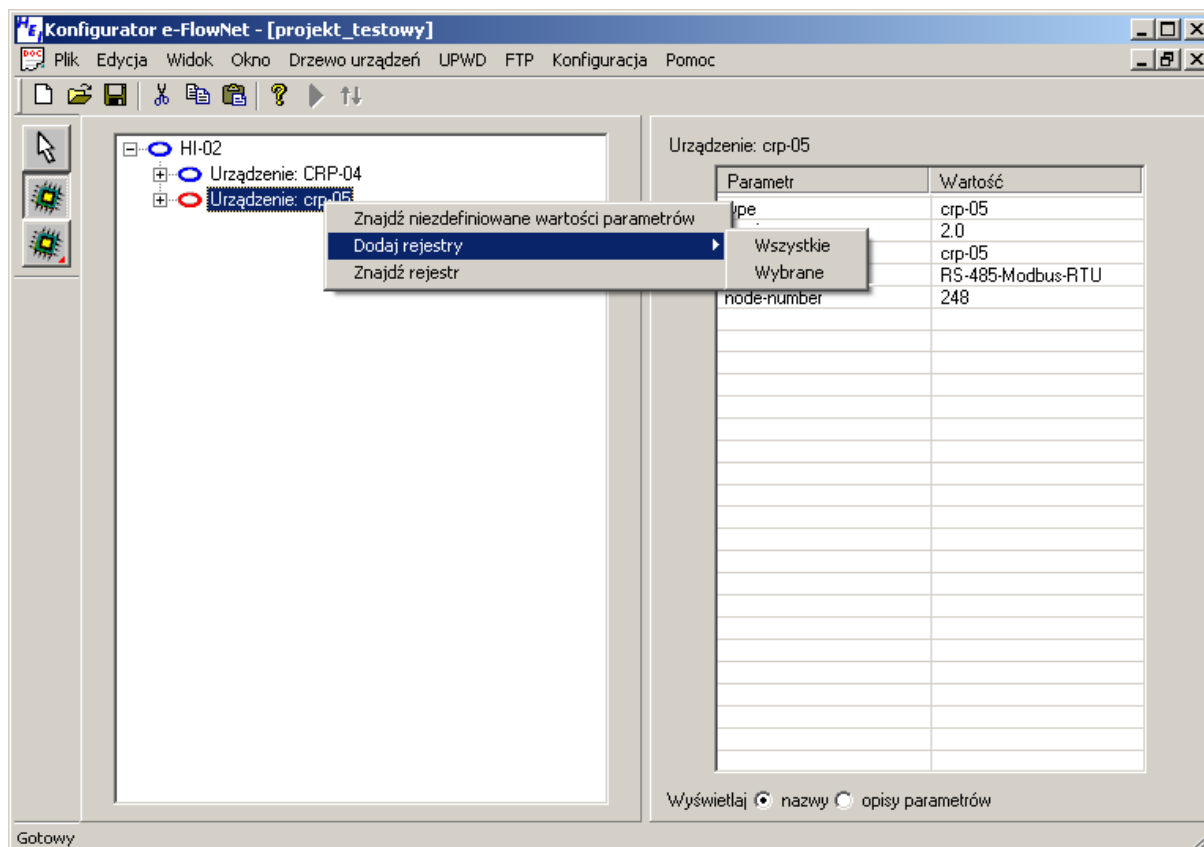
Rys. 3.5. Szczegółowe informacje o parametrze.

Większość parametrów posiada wartości domyślne, jednak w niektórych przypadkach (np. dla adresów sieciowych) istnienie takich wartości byłoby szkodliwe. Takie parametry oznaczone są jako „niezdefiniowane” (napis #niezdefiniowany# w polu wartości). Przy każdej próbie zapisania projektu posiadającego parametry o niezdefiniowanych wartościach użytkownik jest o tym informowany. Dostępna w menu „Drzewo urządzeń” funkcja „Znajdź niezdefiniowane wartości parametrów automatycznie wyszukuje i zaznacza pierwszy taki parametr zaznaczonego urządzenia (urządzeń).

Istnieją też tzw. parametry złożone. Zmiana wartości takiego parametru pociąga za sobą usunięcie z projektu wszystkich parametrów zależnych od tej wartości. Dla przykładu zmiana parametru urządzenia definiującego typ obsługującej je magistrali z wartości Modbus na RS-232 pociągnie za sobą usunięcie parametrów definiujących połączenie Modbus i dodanie parametrów definiujących połączenie RS-232.

### 3.5 Dodawanie i usuwanie rejestrów

W wielu przypadkach nie wszystkie rejestry udostępniane przez fizyczne urządzenie mają znaczenie dla konkretnego projektu. Rejestry takie domyślnie nie pojawiają się w nowotworzonych interfejsach urządzeń, choć mogą być doń dodane poprzez wybór w menu „Drzewo urządzeń” opcji „Dodaj rejestry”. Rejestry dodawane tylko są do zaznaczonego w drzewie urządzenia (urządzeń). Zaznaczenie serwera komunikacyjnego umożliwia dodanie rejestrów do wszystkich urządzeń.



Rys. 3.6. Dodawanie rejestrów do urządzenia.

W zależności od tego, czy użytkownik wybierze opcję „Wszystkie” czy „Wybrane”, do urządzenia (urządzeń) dodane zostaną wszystkie dostępne rejestry, lub wyświetlone zostanie okno dialogowe pozwalające na ich wybór.

Użytkownik może w każdej chwili usunąć rejestr z urządzenia naciskając klawisz delete.

### 3.6 Zapis projektu

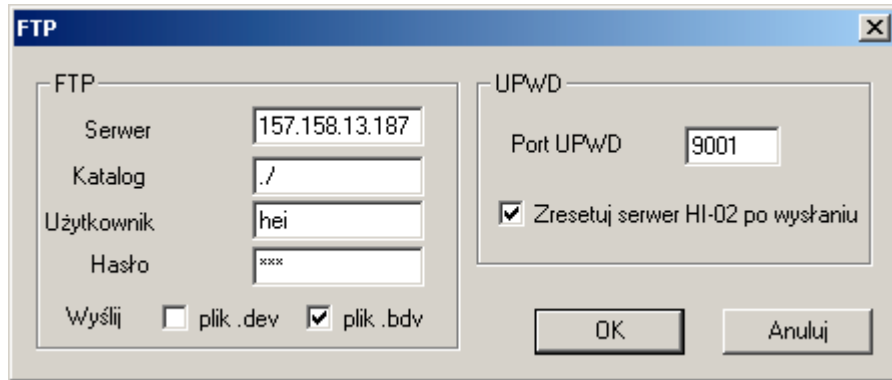
Konfiguracji urządzeń może być w każdej chwili zapisana (a później odczytana). Zapisanie projektu (funkcja Zapisz w menu Plik) powoduje utworzenie katalogu o nazwie odpowiadającej nazwie projektu i zapisanie w nim trzech plików:

- nazwa\_projektu.dev – plik xml opisujący zdefiniowane drzewo urządzeń
- nazwa\_projektu.bdv – plik xml zapisany w postaci binarnej, również zawierający definicję drzewa urządzeń, przeznaczony dla serwera HI-02
- application.xml – parametry programu związane z konkretnym projektem (adresy FTP, i UPWD)

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

### 3.7 Klient FTP

Wbudowany klient FTP umożliwia wysyłanie konfiguracji drzewa urządzeń do serwera komunikacyjnego lub dowolnego innego serwera FTP. Jest dostępny poprzez pozycję menu „FTP”. Po jej wybraniu pojawia się przedstawione na rysunku 3.6 okno dialogowe pozwalające określić parametry połączenia.



Rys. 3.6. Dialog klienta FTP.

Ono dialogowe domyślnie wykorzystuje wartości zdefiniowane przy tworzeniu nowego projektu. Standardowa konfiguracja („Wyslij plik .bdv”, „Zresetuj serwer HI-02”) wysyła plik bdv (plik binarny z definicją drzewa urządzeń przeznaczony dla serwera HI-02), a po zakończeniu transmisji korzystając z protokołu UPWD próbuje uruchomić ponownie serwer komunikacyjny (zakładając, że serwer FTP to HI-02), by odczytał nową konfigurację. Prócz pliku binarnego możliwe jest również wysłanie pliku dev (pliku xml), jednak ze względu na jego duże (z punktu widzenia serwera komunikacyjnego) rozmiary, nie należy wysyłać go do serwera HI-02.

### 3.8 Projekty zdalne

Konfigurator urządzeń 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 dev – patrz rozdział czwarty). Wybrany plik zostanie pobrany z sieci i otwarty w aplikacji.

### 3.9 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 4).

## 4. Formaty plików konfiguracyjnych

Aby rozpocząć pracę program potrzebuje danych zawartych w dwóch plikach wejściowych: „application.xml” i „device.xml”. 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 urządzeń dostępnych w trakcie pracy z programem. Na jego podstawie przy starcie tworzony jest pasek narzędzi z kontrolkami.

### 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>157.158.13.187</ftp.host>
<ftp.directory>./ </ftp.directory>
<ftp.user>hei</ftp.user>
<upwd.host>157.158.13.187</upwd.host>
<upwd.port>9001</upwd.port>
<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

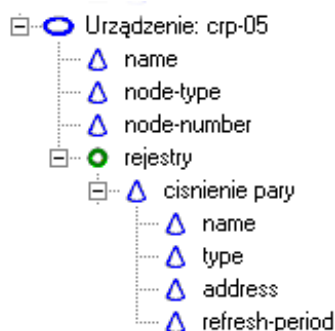
upwd.host – adres serwera upwd

upwd.port – port serwera upwd

html.browser – nazwa przeglądarki wykorzystywanej przez program (m.in. do wyświetlania plików pomocy)

### 4.1 Plik device.xml

Plik „device.xml” definiuje urządzenia, ich rejestry i parametry dostępne w trakcie pracy z programem. Ponieważ urządzenie ma postać drzewa:



naturalnym wyborem zapisu informacji o urządzeniach jest język xml pozwalający dokładnie odwzorować taką strukturę:

```
<device type="crp-05" version="2.0">
  <param>
    <name>name</name>
    <value>crp-05</value>
  </param>
  <param>
    <name>node-type</name>
    <value>RS-485-Modbus-RTU</value>
  </param>
  <param>
    <name>node-number</name>
    <value>248</value>
  </param>
  <register type="cisnienie pary" writable="yes">
    <param>
      <name>name</name>
      <value>cisnienie pary</value>
    </param>
    <param>
      <name>type</name>
      <value>float</value>
    </param>
    <param>
      <name>address</name>
      <value><undefined/></value>
    </param>
    <param>
      <name>refresh-period</name>
      <value>1000</value>
    </param>
    <param>
      <name>refresh-timeout</name>
      <value>3000</value>
    </param>
  </register>
</device>
```

Ogólna struktura pliku device.xml wygląda następująco:

```
<device.project> <!--korzeń drzewa-->
  <hub><!--serwer komunikacyjny-->
    <device><!--urządzenie-->
      <param> <!--definicja parametru urządzenia-->
        <!-- ...-->
      </param>

      <!-- ...-->
      <!-- więcej definicji parametrów -->
      <!-- ...-->

      <register> <!--definicja rejestrów-->
        <param> <!--definicja parametru rejestru-->
          <!-- ...-->
        </param>
        <!-- ...-->
        <!--więcej definicji parametrów -->
        <!-- ...-->
      </register>
    </device>
  </hub>
</device.project>
```

```

        </register> <!--koniec definicji rejestru-->

        <!-- ...-->
        <!--więcej definicji rejestrów -->
        <!-- ...-->

    </device><!--koniec definicji urządzenia-->

    <!-- ...-->
    <!--więcej definicji urządzeń -->
    <!-- ...-->

</hub>
<device.project><!--koniec definicji urządzeń-->

```

Wszystkie parametry urządzeń i rejestrów definiowane są w jednolity sposób, tak by uprościć odczyt plików oraz dodawanie lub usuwanie parametrów. Znacznik param może posiadać następujące elementy i atrybuty:

Atrybuty znacznika param:

type – typ parametru; dopuszczalne typy przedstawia tabela 4.1

Dopuszczalne typy parametrów.

Tabela 4.1

Typ parametru	Opis
text	Dane tekstowe (np. nazwa urządzenia).
int	Liczba typu integer.
float	Liczba rzeczywista typu float.
choice	<p>Wartość parametru może być wybrana spośród jednej z kilku podanych – np. dla parametru „typ magistrali” dopuszczalne wartości to:            RS-232            RS-485</p> <p>Wewnątrz znacznika param dopuszczalne wartości dla typu choice umieszczone są wewnątrz znacznika &lt;values&gt;</p> <pre> &lt;values&gt; &lt;value&gt;RS-232&lt;/value&gt; &lt;value&gt;RS-485&lt;/value&gt; ... &lt;/values&gt; </pre> <p>Parametr tego typu jest reprezentowany w listach właściwości jako pole wyboru „tzw. „combo box”).</p>
complex	<p>Parametr złożony. Podobnie jak dla typu choice, wartość parametru może być wybrana spośród jednej z kilku, ale zmiana jej wartości pociąga za sobą zmianę parametrów zależnych (więcej informacji na ten temat w punkcie 3.4 i poniżej)</p>

writable – atrybut informuje czy parametr może zostać zmodyfikowany przez użytkownika

visible – atrybut informuje czy parametr jest widoczny dla użytkownika; w ten sposób niektóre parametry pozostają niewidoczne dla użytkownika;

dependency – określa od jakiej wartości parametru złożonego zależy dany parametr; ten atrybut ma sens jedynie dla parametru zagnieżdżonego w parametrze złożonym

W przykładowym parametrze złożonym:

```
<param type="complex">
  <name>node-type</name>
  <default.value>RS-485-Modbus-RTU</default.value>
  <values>
    <value>RS-485-Modbus-RTU</value>
    <value>RS-232C-FlowNet</value>
    <value>UPWD-Server</value>
  </values>
  <param type="int" dependency="RS-232C-FlowNet">
    <name>parametr zależny 1</name>
    <default.value>1</default.value>
  </param>
  <param type="int" dependency="UPWD-Server">
    <name>parametr zależny 2</name>
    <default.value>2</default.value>
</param>
```

parametr zależny 1 będzie zdefiniowany jeśli dla parametru `node-type` zostanie wybrana wartość `RS-232C-FlowNet`, natomiast parametr zależny 2 będzie zdefiniowany jeśli dla wartości `UPWD-Server`

Znaczniki dopuszczalne wewnątrz znacznika `param` przedstawia tabela 4.2.

Znacznik	Opis
<code>name</code>	Nazwa parametru.
<code>description</code>	Opis parametru.
<code>default.value</code>	Domyślna wartość.
<code>values</code>	Lista wartości parametru – tylko dla typu <code>choice</code> . Wewnątrz tego znacznika znajduje się jeden lub więcej znaczników <code>&lt;value&gt;</code> definiujących dopuszczalne wartości parametru.
<code>param</code>	Parametr zależny zagnieżdżony w parametrze złożonym.

Definicja rejestru sprowadza się do podania listy jego parametrów oraz określenia wartości atrybutów. Rejestr posiada następujące atrybuty:

`name` – nazwa rejestru

`visible` – informacja czy rejestr jest domyślnie widoczny; rejestry zdefiniowane jako niewidoczne lub usunięte z urządzenia można ponownie do niego dodać (więcej na ten temat w punkcie 3.5); jeśli atrybut nie jest zdefiniowany, rejestr domyślnie jest widoczny;

`writable` – informacja o tym, czy definiowany rejestr jest rejestrem zapisywalnym, czy też rejestrem tylko do odczytu (pomiarom); jeśli atrybut nie jest zdefiniowany, rejestr domyślnie jest zapisywalny;

Dodatkowo rejestr powinien posiadać znacznik `description` opisujący jego zastosowanie, np.:

```
<register name="Y0" writable="yes">
    <description>Opis: ciśnienie pary</description>
...

```

Definicja urządzenia to lista jego parametrów rejestrów oraz znacznik `description` zawierający opis. Ponadto urządzenie posiada następujące atrybuty:

`type` – typ urządzenia (np.: `crp-05`);

`version` – wersja urządzenia (np. 1.0); typ i wersja razem określają urządzenie. Jeśli jednak w definicji urządzeń (pliku `device.xml`) pojawią się dwa urządzenia o tym samym typie i wersji, są uznawane za podwersje tego samego urządzenia, a w aplikacji grupowane pod jedną ikoną (więcej na ten temat w punkcie 3.3);

`icon` – nazwa ikony urządzenia reprezentującej je na pasku narzędzi (np. `device.png`); ikony przechowywane są w katalogu `config` aplikacji;

### 4.3 Pliki projektów

Na początku należy zaznaczyć różnicę między plikiem konfiguracyjnym `device.xml` a plikiem projektu. Plik `device.xml` definiuje dostępne typy urządzeń – określa jakie parametry i rejestry posiadają urządzenia (ewentualnie podaje domyślne wartości tych parametrów). Natomiast plik projektu opisuje konkretne egzemplarze urządzeń podłączone do serwera i wartości ich parametrów. Można pokusić się o stwierdzenie, że jeśli plik wejściowy `device.xml` definiuje klasy urządzeń, to wyjściowy plik projektu stanowi zapis obiektów tych klas.

Projekt drzewa urządzenia zapisywany jest w dwóch formatach: `xml` i `binary xml`. Plik `xml` (posiadający rozszerzenie `dev`) zapisywany jest na użytek programu i pozwala na późniejszą modyfikację projektu – jest dokumentem który aplikacja może odczytywać i zapisywać. Natomiast `binary xml` (będący adaptacją opracowanego na potrzeby telefonii komórkowej standardu WAP Binary XML) jest przeznaczony dla serwera HI-02. Zakodowany binarnie `xml` może być ok. dziesięciu razy mniejszy od przenoszącego tę samą informację zwykłego pliku `xml`.

Format pliku projektu, będący uproszczoną i nieco zmodyfikowaną formą stosowanego w pliku „`device.xml`” jest następujący:

```
<device.project> <!--korzeń drzewa-->
    <hub><!--serwer komunikacyjny-->
        <param> <!--definicja parametru urządzenia-->
            <!-- ...-->
        </param>

        <!-- ...-->
            <!-- więcej definicji parametrów -->
        <!-- ...-->

        <device><!--urządzenie-->
            <param> <!--definicja parametru urządzenia-->
                <!-- ...-->
            </param>
        </device>
    </hub>
</device.project>

```

```

        </param>

        <!-- ...-->
        <!-- więcej definicji parametrów -->
        <!-- ...-->

        <register> <!--definicja rejestrów-->
            <param> <!--definicja parametru rejestru-->
                <!-- ...-->
            </param>
            <!-- ...-->
            <!--więcej definicji parametrów -->
            <!-- ...-->
        </register> <!--koniec definicji rejestru-->

        <!-- ...-->
        <!--więcej definicji rejestrów -->
        <!-- ...-->

    </device><!--koniec definicji urządzenia-->

    <!-- ...-->
    <!--więcej definicji urządzeń -->
    <!-- ...-->

</hub>
<device.project><!--koniec definicji urządzeń-->

```

Zasadnicza różnica między plikiem projektu dotyczy sposobu zapisu parametrów. Parametry zapisywane są w następujący sposób:

```

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

```

A więc zapisywane są tylko nazwa i wartość parametru.

Atrybut `type` znacznika `device` stanowi powiązanie między definicją urządzenia (obiektem) a definicją typu urządzenia (klasą). Jeśli plik „device.xml” definiuje typ o nazwie „crp-05”, urządzenie tego typu rozpoczynałoby się od znacznika `<device type="crp-05">` Analogicznie atrybut `type` znacznika `register` wiąże typ rejestru (z pliku device.xml) z jego wystąpieniem (z pliku projektu).

Jeśli urządzenie (lub rejestr) poprzez atrybut `type` zostanie skojarzone z typem urządzenia, jego parametry będą poprzez nazwę kojarzone z typami parametrów. Program nie pozwala na otwarcie projektów definiujących urządzenia lub rejestry nie posiadające typów w device.xml.

Dodatkowo w katalogu projektu zapisywany jest plik application.xml. Jego format jest identyczny jak pliku o tej samej nazwie z katalogu config.

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) z którymi jest związany, a równocześnie użytkownik nie musi ich pamiętać i wpisywać za każdym razem.