



„NEMEZIS” – fundusz alimentacyjny

Dokumentacja techniczna

„INFO-R” Spółka Jawna – Pogórze 2013

**43-430 Pogórze, ul. Baziowa 29 , tel. (033) 479 93 29 (89), fax (033) 853 04 06
e-mail: admin@ops.stefa.pl**

Spis treści

1. Opis programu	3
2. Wymagania sprzętowe	3
3. Serwer bazy danych FireBird.....	4
4. Instalacja programu wersja jednostanowiskowa.....	6
4. Instalacja programu wersja sieciowa.....	14
6 .Konfiguracja sytemu w architekturze klient-serwer.....	15
7 .Struktura plików programu.....	16
8. Weryfikacja i naprawa pliku bazy danych.....	17
9. Jak wykonać kopię przyrostową bazy danych.....	21

1. Opis programu

Program Nemezis został opracowany przy użyciu środowiska programistycznego „C++Builder” wersja 6.00 firmy „Borland”®. Kod źródłowy w całości został stworzony w języku C++.

Jako „motor” bazy danych zastosowano system baz danych FireBird® firmy Borland® w wersji 2.00, rozpowszechniany na zasadach OpenSource. Użytkowanie bazy danych nie wymaga zakupienia dodatkowych licencji.

Serwer bazy danych może pracować lokalnie, na jednym komputerze, na którym zainstalowano program Nemezis, lub w sieci typu peer-to-peer oraz w sieciach z dedykowanym serwerem. W tym ostatnim przypadku serwer może być zarządzany przez oprogramowanie firm Microsoft® lub pracować pod kontrolą systemu Linux .

Aplikacja klienta pracuje wyłącznie w systemach operacyjnych firmy Microsoft®, począwszy od wersji Windows 95.

2. Wymagania sprzętowe

a) aplikacja klienta

	minimalne	zalecane
procesor	dowolny	Pentium® Celeron® 1 GB lub odpowiednik (lub nowsze)
pamięć	32 MB	128 MB
miejsce na dysku twardym	10 MB	50 MB
inne	CD-ROM	CD-ROM
system operacyjny	Microsoft® Windows od wersji 95	

b) serwer bazy danych

	minimalne	zalecane
procesor	dowolny	Pentium II ® 1 GB lub odpowiednik (lub nowsze)
pamięć	32 MB	128 MB
miejsce na dysku twardym	20 MB na potrzeby instalacji	100 MB
inne	CD-ROM	CD-ROM
system operacyjny	Microsoft® Windows Server™ 2003, Windows XP™, Windows® 2000 (SP2), ME, 98 SE, Windows NT® 4.0 (SP6a lub nowszy), Linux Wersja jądra 2.2 lub nowsza (zalecane dystrybucje Red Hat® Linux i SuSE® Linux)	

3. Serwer bazy danych FireBird

Baza danych FireBird to potężna, w pełni funkcjonalna, wbudowywalna baza danych, która pozwala zminimalizować koszty, a zarazem utrzymać duże tempo tworzenia skomplikowanych aplikacji biznesowych. Jest zaprojektowana pod kątem szerokiego zastosowania wszędzie tam, gdzie zasoby systemowe są niewystarczające, a obsługa informatyczna - niedostępna. FireBird to kluczowy składnik bazodanowy w pakiecie technologii przeznaczonych do zarządzania cyklem życia aplikacji. Ich zadaniem jest przyspieszenie tworzenia i wdrażania oprogramowania, przy jednoczesnym zmniejszeniu kosztu eksploatacji infrastruktury informatycznej. Baza FireBird jest niezależna od platformy, łatwa w użyciu i potrafi obsługiwać skomplikowane aplikacje biznesowe pisane z myślą o wielu jednoczesnych użytkownikach, a mimo to oferuje wysoką niezawodność, niski koszt wdrożenia i ekonomiczną eksploatację. Jest zoptymalizowana pod kątem niskiego całkowitego kosztu eksploatacji i szybkiego zwrotu z inwestycji.

Niski koszt i elastyczność potężnej, wbudowanej bazy danych.

Baza danych FireBird jest przeznaczona do obsługi aplikacji, które pełnią krytyczne funkcje w dużych środowiskach sieciowych lub autonomicznych, zwłaszcza takich, w których brak personelu informatycznego. FireBird to elastyczne rozwiązanie wdrożeniowe, które potrafi obsługiwać niezwykle wymagające aplikacje, a mimo to może być zainstalowane w sposób niewidoczny dla użytkowników końcowych. Baza jest zbudowana i zoptymalizowana pod kątem wbudowywania w wyspecjalizowane aplikacje, cieszy się dużą popularnością i jest stosowana w aplikacjach powszechnego użytku, z których korzystają tysiące użytkowników. Wiele korporacji używa jej jako wdrożeniowej bazy danych w złożonych aplikacjach biznesowych, które obsługują sieciowych użytkowników w rozproszonych oddziałach, zagranicznych zakładach produkcyjnych i kanałach dystrybucji. W największych przedsięwzięciach licencyjnych baza FireBird gromadzi dane z urzędzeń tak różnorodnych jak dystrybutory paliwowe, kasy w punktach sprzedaży i urządzenia do przygotowywania żywności, a następnie przekazuje je do centralnych serwerów w siedzibie firmy. Z biegiem lat baza FireBird dowiodła, że jest rozwiązaniem bardzo wydajnym, ekonomicznym i niezawodnym.

Bardzo niskie wymagania co do konserwacji i łatwa instalacja

Funkcje samodzielnej administracji, takie jak automatyczne odzyskiwanie sprawności po awarii, programowalne tworzenie kopii zapasowych online i zdolność do automatycznego dostrajania pomagają ograniczyć koszty konserwacji. Jeśli przedsiębiorstwo musi wdrożyć nową aplikacyjną bazę danych w tysiącach lokalizacji, w których brak wykwalifikowanego personelu technicznego i instalacyjnego, baza FireBird jest idealnym rozwiązaniem, ponieważ zawiera funkcje i interfejsy instalacyjne, które całkowicie automatyzują konfigurację i instalację. Co więcej, jest na tyle niewielka, że można ją przesłać do użytkowników końcowych przez telefoniczne połączenie internetowe.

Mniejszy całkowity koszt eksploatacji infrastruktury informatycznej

FireBird pozwala zmniejszyć koszty tworzenia aplikacji, pomagając zespołom programistów budować i wdrażać aplikacje w terminie i bez przekraczania budżetu. Ograniczenie kosztów wynika nie tylko z prostszych funkcji instalacyjnych, ale także ze skromnych wymagań systemowych i braku opłat licencyjnych. Ponieważ zaś FireBird nie musi być obsługiwana na miejscu przez administratora i wymaga minimalnej konserwacji, bieżące koszty administracyjne są bardzo niskie.

Zgodność ze standardami branżowymi

FireBird zachowuje ścisłą zgodność z ewoluującymi standardami branżowymi dla środowisk obliczeniowych typu klient-serwer, takich jak ANSI/SQL, Java™, Unicode®, Extensible Markup Language (XML) oraz External Data Representation (XDR). Dzięki stosowaniu się do tych najważniejszych norm technologicznych programiści mogą skrócić czas tworzenia, wdrażania i konserwowania aplikacji przeznaczonych dla wielu platform. Brak specyficznej składni i zgodność FireBird ze standardem SQL92 oznaczają, że programiści znający inne serwery baz danych SQL, takie jak Oracle®, IBM® DB2® i Sybase®, szybko osiągną pełną produktywność pracując z FireBird. Aplikacje zbudowane w oparciu o serwer FireBird można także łatwo przenieść do innych baz, gdyby tak nakazywały wymagania biznesowe.

Unikalna architektura

Sercem FireBird zbudowany jest w oparciu o wyrafinowaną, wielowersyjną architekturę, przynoszącą wielorakie korzyści w zakresie niezawodności, wydajności, produktywności programistów i łatwości w bieżącym utrzymaniu. Specyfika tej architektury pozwala programistom uwolnić się od wielu problemów związanych ze współbieżnością wykonywania operacji i efektywnym zarządzaniem pamięcią, a w przypadku awarii zapewnia automatycznie, natychmiastowe przywrócenie normalnych warunków pracy.

Minimalne administrowanie

FireBird stanowi idealne narzędzie do pracy w warunkach "bezobsługowych" - czyli przy minimalnym udziale administratora baz danych i wsparcia IT, a nawet przy zupełnym ich braku. Automatyczne przywracanie stanu po awarii, automatyzacja zarządzania użytkownikami, tworzenie kopii zapasowych na działającej bazie danych - to tylko niektóre mechanizmy FireBird minimalizujące konieczność interwencji administratora. Mechanizmy automatycznego dostrajania obejmują optymalną pod względem kosztów realizację żądań i automatyczne odzyskiwanie pamięci oraz innych zasobów. Dynamiczne równoważenie struktur indeksowych samoczynnie zwiększa efektywność pracy, co jeszcze bardziej ogranicza niezbędne czynności administracyjne.

Zgodność z SQL 92

Brak jakiegokolwiek uzależnienia FireBird od jakiegokolwiek nietypowej syntaktyki SQL powoduje, że polecenia języka SQL dla FireBird są czymś naturalnym dla programistów posiadających doświadczenie w pracy z innymi bazami SQL. Ponadto zgodność ze standardem SQL 92, ze względu na szerokie spektrum poleceń akceptowanych przezeń, minimalizuje prawdopodobieństwo powstawania rozmaitych barier przy tworzeniu specyficznych rozwiązań.

Łatwiejsza instalacja

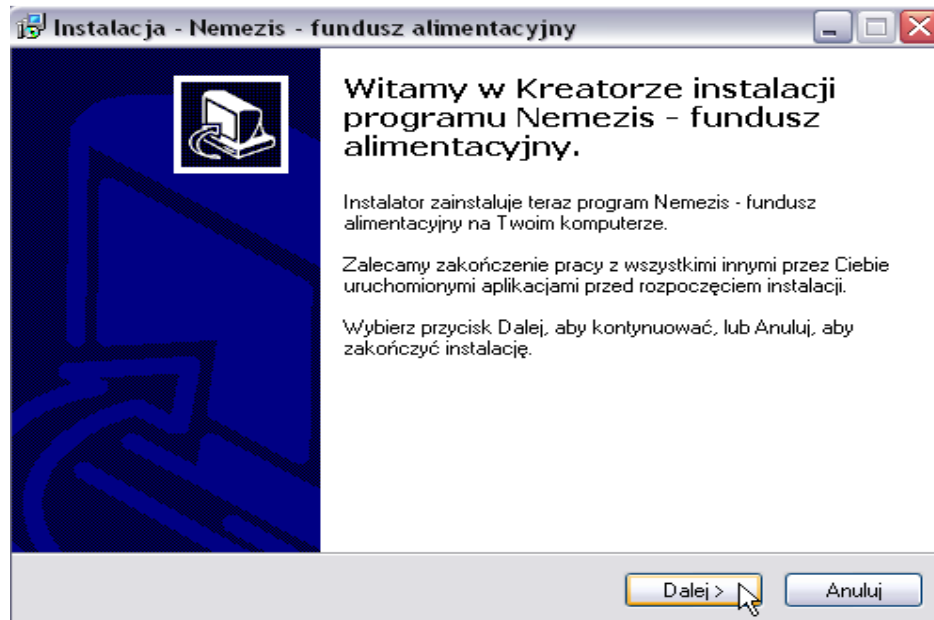
Stosunkowo niewielkie wymagania systemowe ze strony FireBird umożliwiają jego zastosowanie w systemach z niewielką pamięcią RAM i ograniczoną pojemnością pamięci dyskowych. Konfigurowalne komponenty instalacyjne umożliwiają "przezroczyste" integrowanie FireBird ze specyficznymi dla konkretnych aplikacji procedurami instalacyjnymi. Struktura licencjonowania FireBird ułatwia programistom tworzenie mechanizmów śledzących wykorzystywanie aplikacji, upraszczając tym samym udzielanie licencji na dostęp do tych aplikacji.

Niezależność od platformy

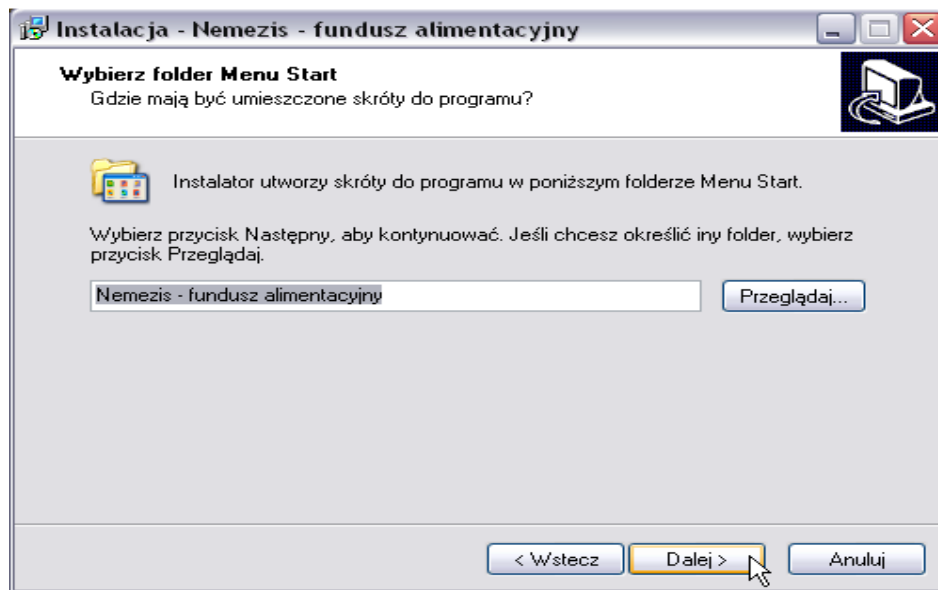
FireBird wolny jest od zagrożeń związanych z uzależnieniem użytkownika lub programisty od konkretnego języka programowania lub technologii konkretnego dostawcy, oferuje bowiem międzyplatformową zgodność z systemami Windows, Linux, Solaris oraz platformami Java i .NET - zapewniając przenośność pomiędzy różnymi mechanizmami zaplecza baz danych bez konieczności zmiany kodu źródłowego aplikacji.

4. Instalacja programu wersja jednostanowiskowa

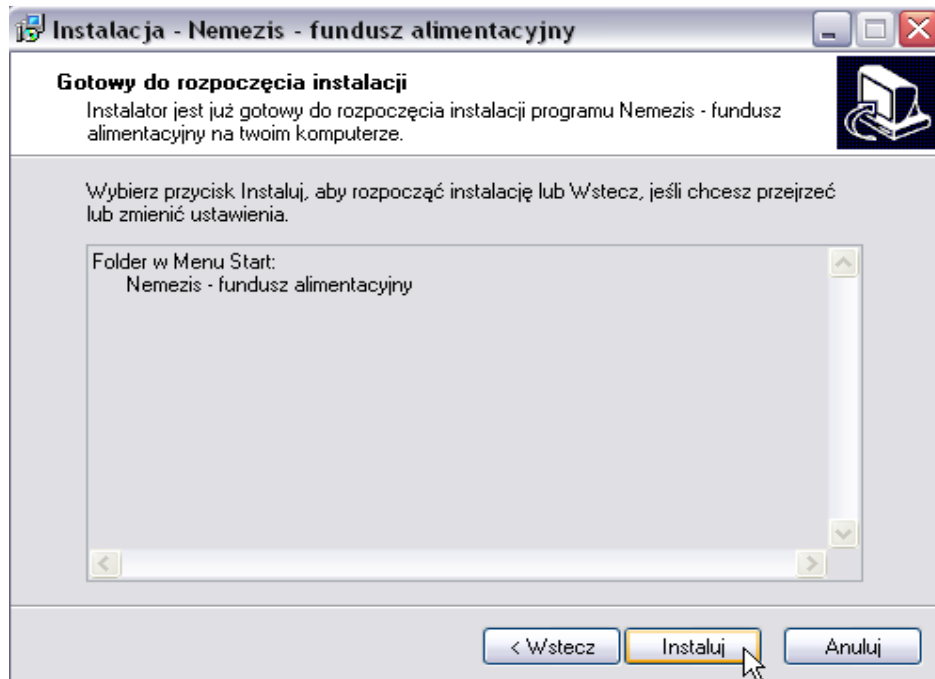
Po włożeniu do czytnika CD-ROM płyty z instalacją programu Nemesis, na ekranie pojawi się okno instalatora programu (jeżeli nie nastąpi automatyczne uruchomienie instalatora, to należy z płyty uruchomić plik **nemsetup.exe**).



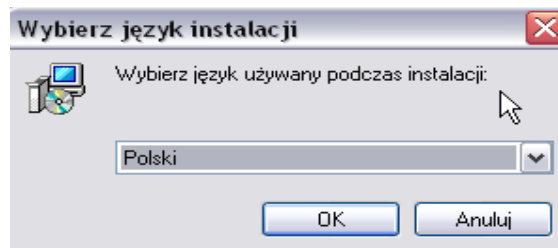
Po wybraniu przycisku **Dalej** pojawi się okno z prośbą o podanie nazwy folderu Menu Start (domyślnie Nemesis).



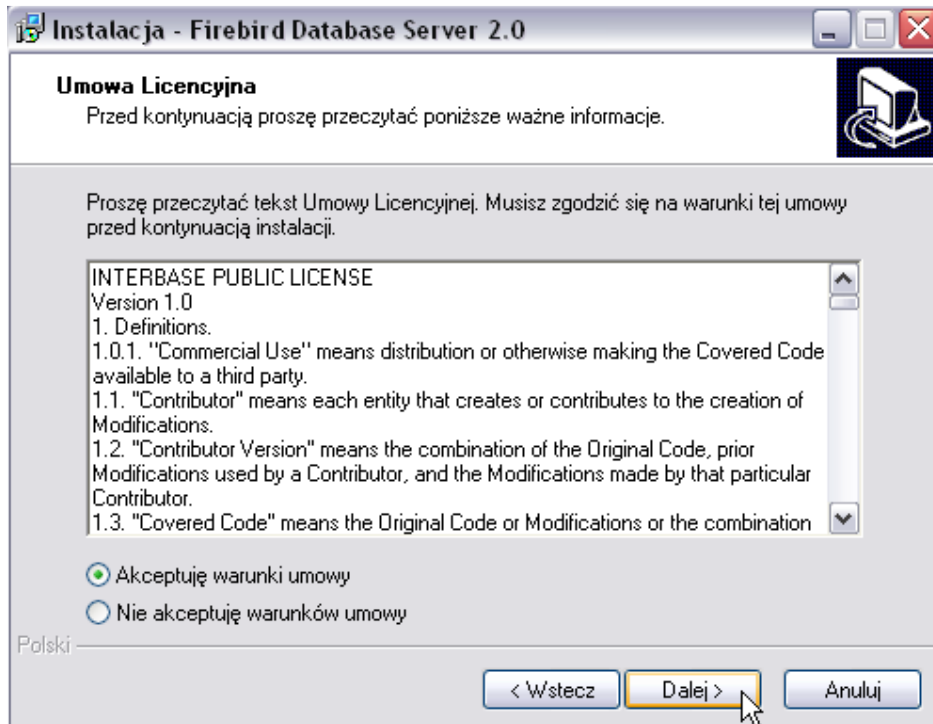
W następnym oknie w celu rozpoczęcia instalacji należy wybrać przycisk **Instaluj**.



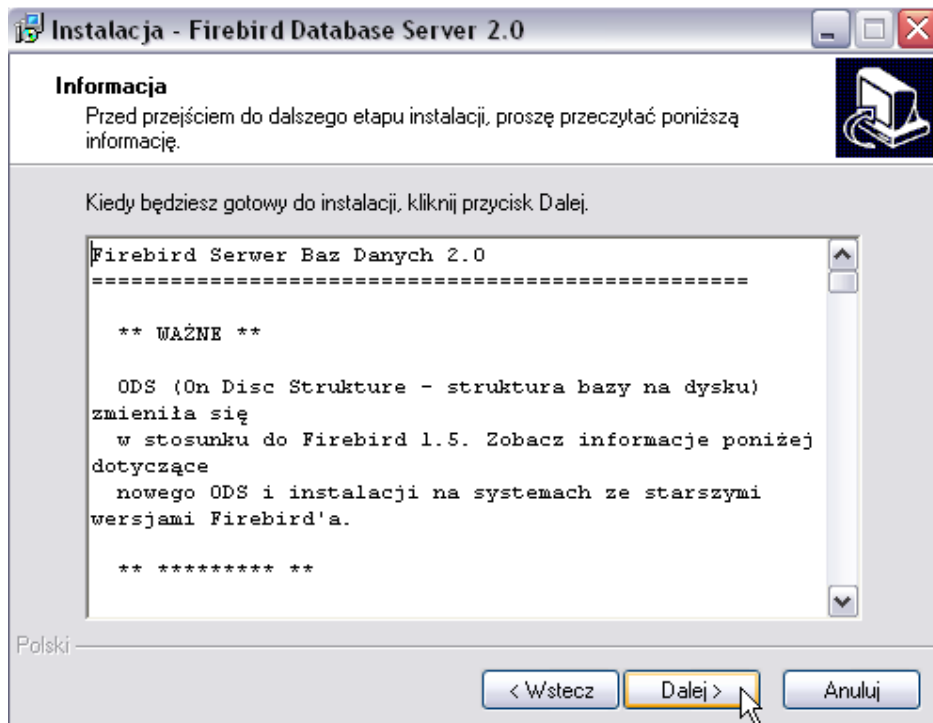
Program rozpocznie instalację, w jej trakcie nastąpi uruchomienie instalacji serwera bazy danych FireBird.



w tym miejscu wybieramy ikonę DALEJ a na ekranie pojawi się kolejne okno instalatora

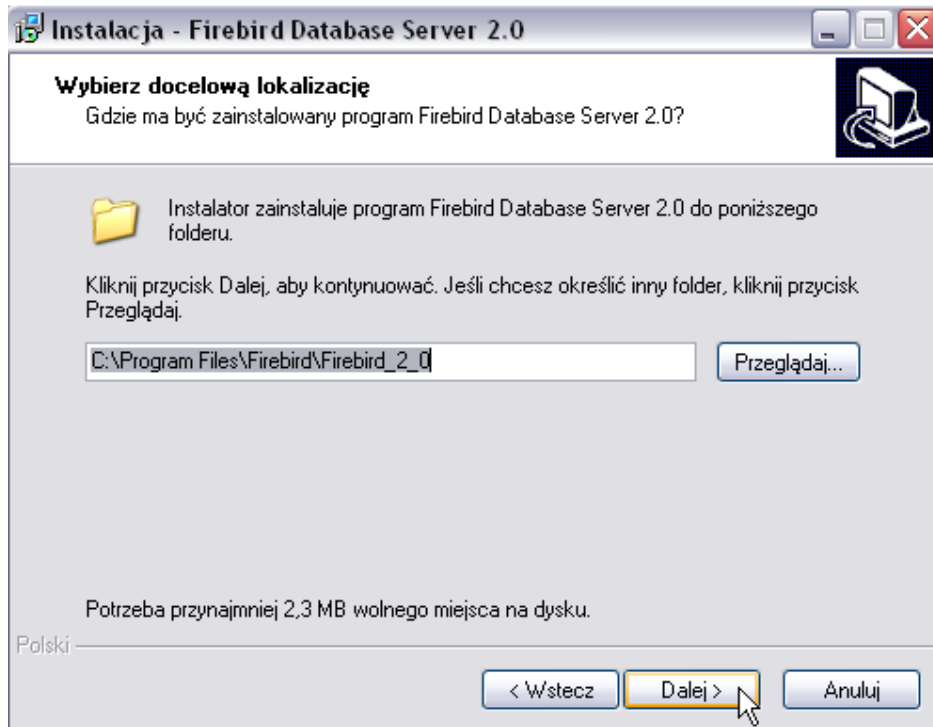


W tym miejscu wybieramy AKCEPTUJĘ WARUNKI UMOWY a następnie ikonę DALEJ

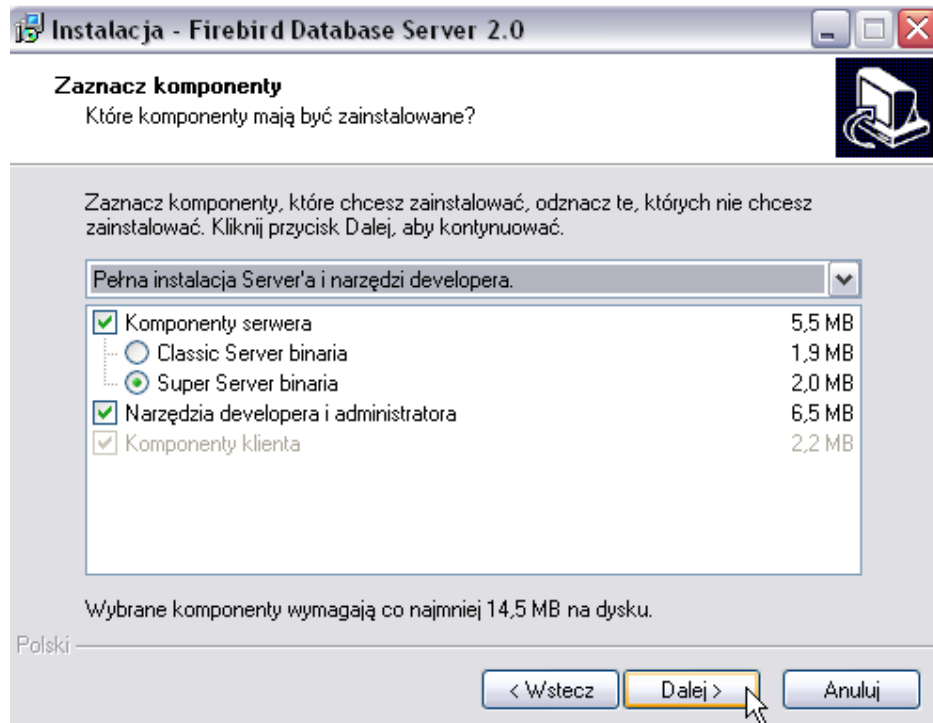


ponownie wybieramy ikonę DALEJ

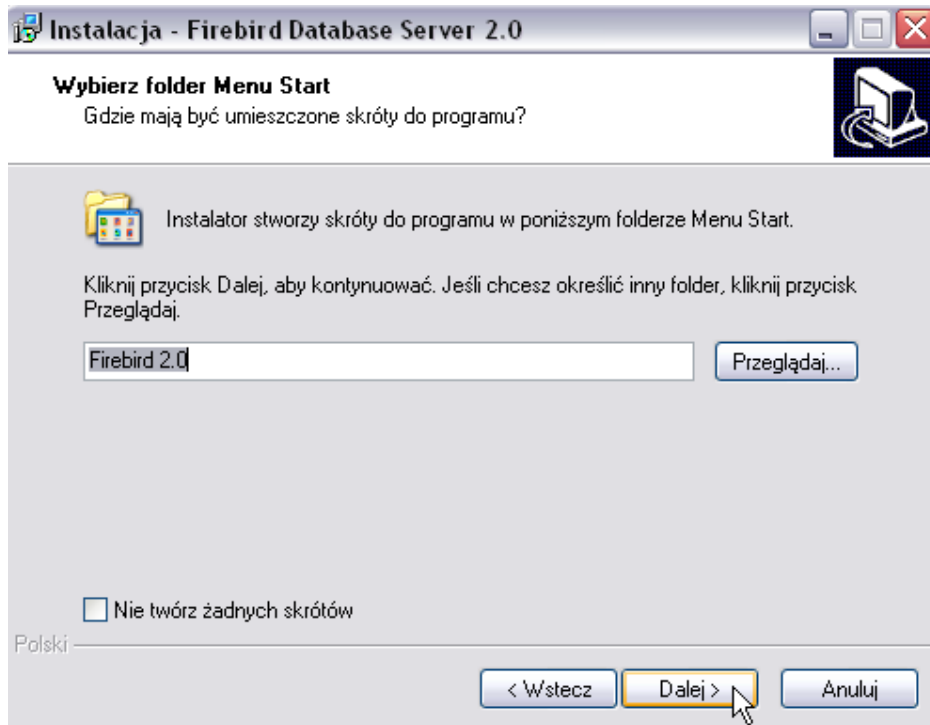
na ekranie pojawi się okno informujące o lokalizacji instalacji plików – wybieramy NEXT :



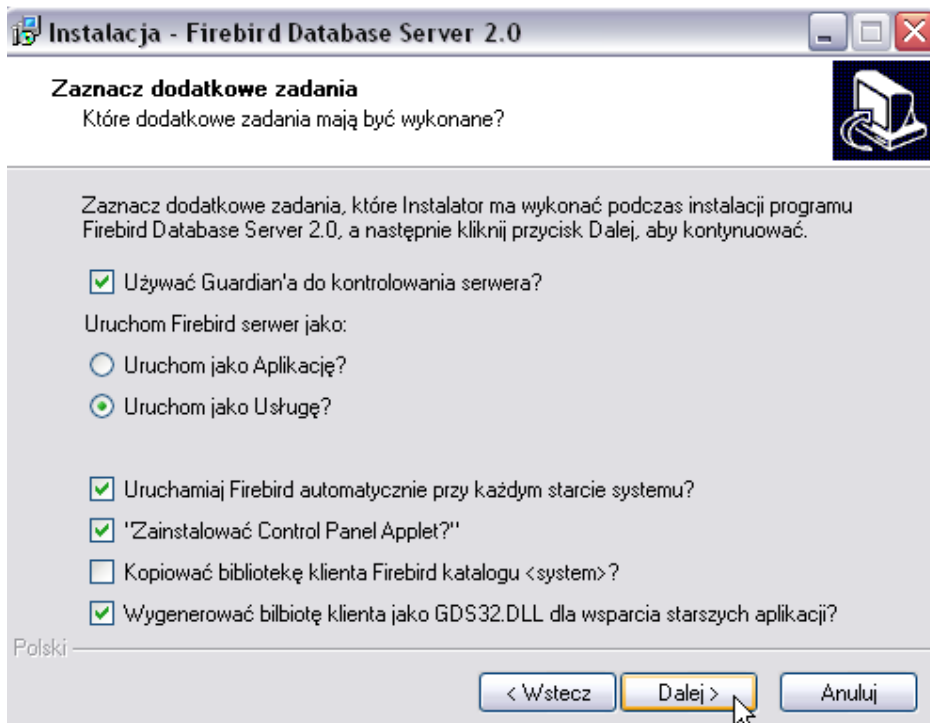
w kolejnym oknie wybieramy DALEJ – program podpowie nam jakie komponenty dodać:



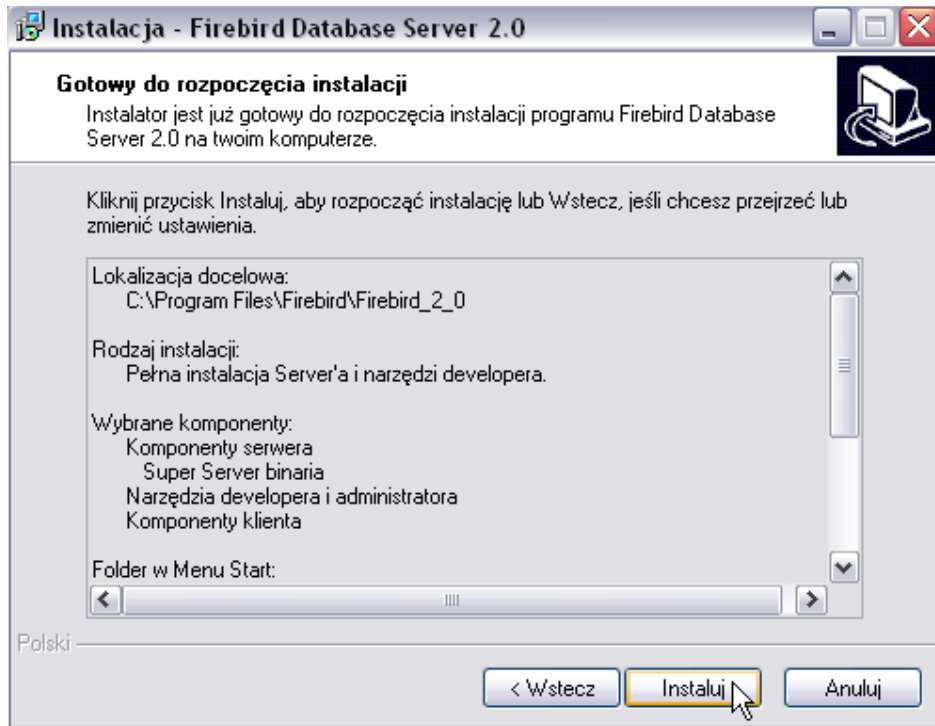
Program instalacyjny poinformuje użytkownika gdzie utworzy skróty:



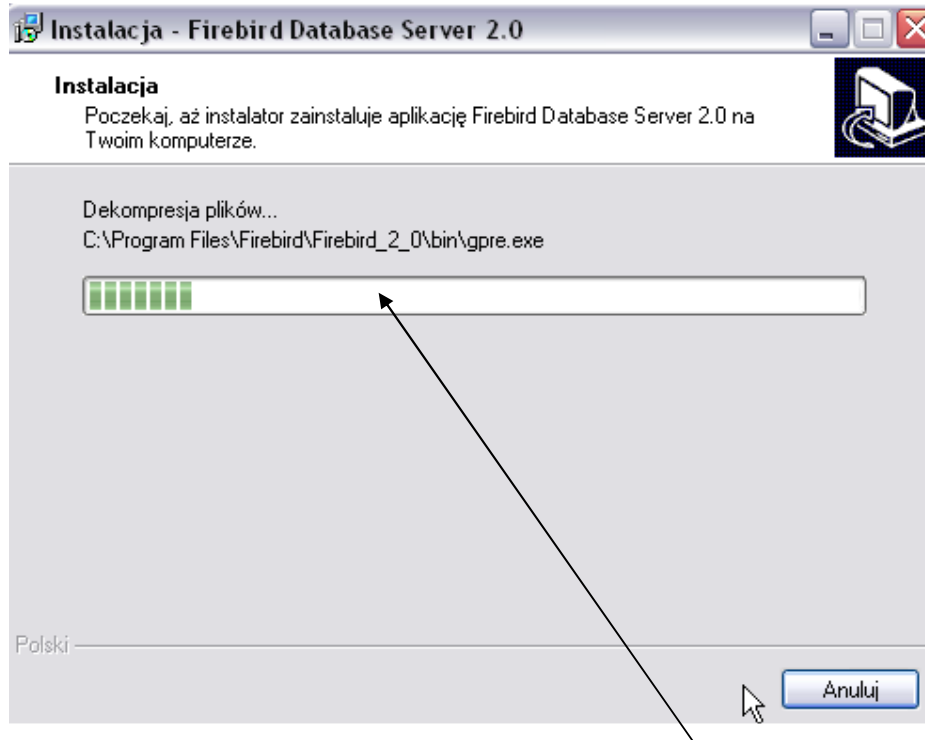
Wybieramy kolejny raz ikonę DALEJ – program zaproponuje domyślne dodatkowe zadania:



nie zmieniając żadnych ustawień wybieramy ikonę DALEJ:

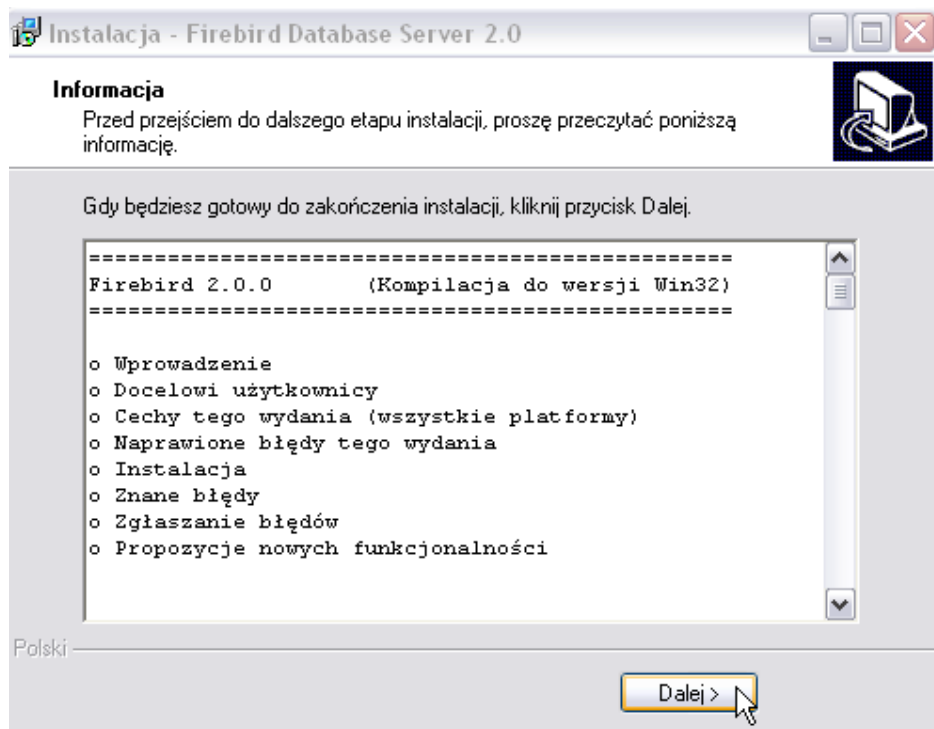


W tym momencie przystępujemy do instalacji FireBird poprzez wybranie ikony INSTALUJ:



Postęp instalacji będzie widoczny na ekranie – na pasku postępu instalacji.

Po zainstalowaniu bazy FireBird program instalacyjny wyświetli następujący komunikat:

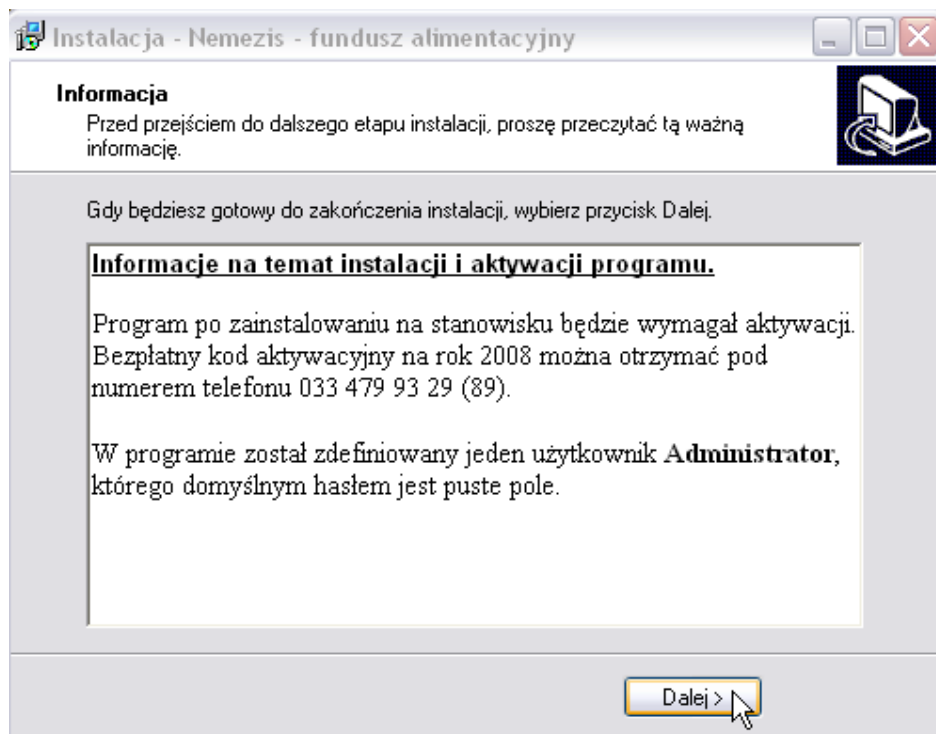


wyberamy ikonę DALEJ:

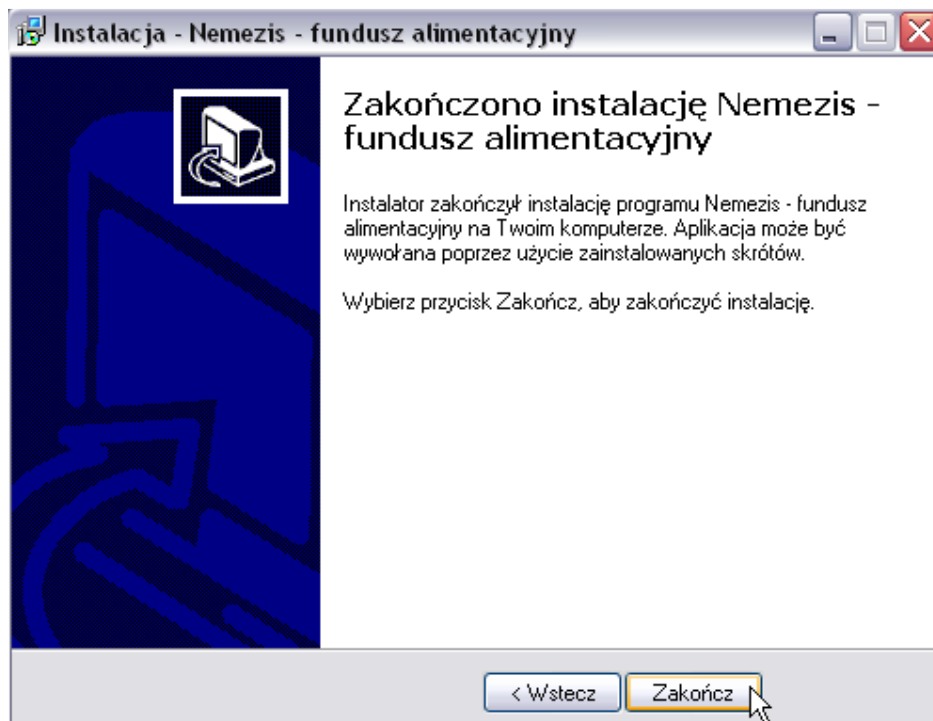


i zakańczamy instalację poprzez wybranie ikony ZAKOŃCZ.

Po zakończeniu instalacji pojawi się okno z informacją na temat aktywacji programu.



W kolejnym oknie program poinformuje o zakończeniu instalacji, należy go zamknąć przyciskiem **Zakończ**.



5 . Instalacja programu wersja sieciowa

Instalację na każdym ze stanowisk przeprowadzamy w sposób analogiczny jak w **pkt 4** (instalacja wersji jednostanowiskowej).

Po zainstalowaniu programu użytkownik nie musi wykonywać żadnych dodatkowych działań i operacji. Jedynie w przypadku instalacji programu na stanowisku roboczym, które będzie pracowało w sieci, korzystając z bazy danych uruchomionej na innym komputerze należy :

- w katalogu **c:\program files\info-r\Nemesis** otworzyć do edycji plik **Nemesis.ini**
- zmodyfikować zapisy

[Baza]

Nemesis=C:\Program files\Info-r\Nemesis\Baza\Nemesis.gdb

Dokument=C:\Program files\Info-r\Nemesis\Baza\Dokument.gdb

należy zmienić na :

[Baza]

Nemesis=**Komputer**:C:\Program files\Info-r\Nemesis\Baza\Nemesis.gdb

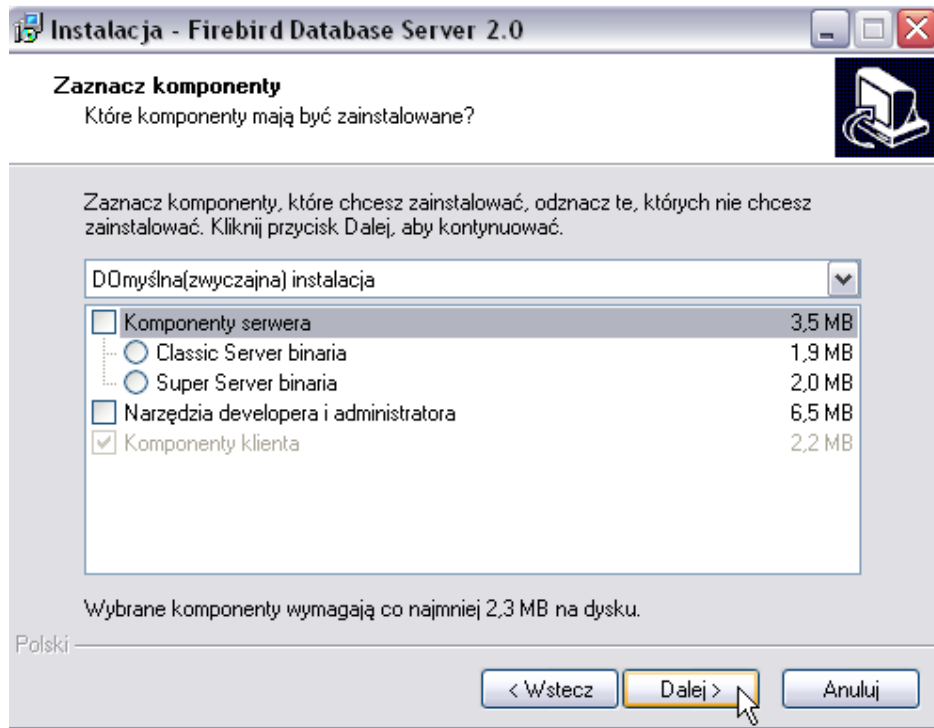
Dokument=**Komputer**:C:\Program files\Info-r\Nemesis02\Baza\Dokument.gdb

gdzie **Komputer** oznacza nazwę komputera na którym pracuje baza danych lub jego adres IP pod którym jest widoczny w sieci.

- po dokonaniu modyfikacji należy zapisać plik **Nemesis.ini**.

6. Konfiguracja sytemu w architekturze klient-serwer

Na komputerze pełniącym rolę serwera instalujemy program analogicznie jak w **pkt 4**. Na komputerze będącym klientem, podczas instalacji motoru bazy danych Firebird, wybieramy następującą opcję konfiguracyjną :



Wybór takiej opcji instalacji bazy danych Firebird, umożliwi uruchomienie programu Nemesis w architekturze „ cienkiego klienta”. Całość operacji związanych z przetwarzaniem danych odbywać się będzie po stronie serwera, z wykorzystaniem procedur składowych zdefiniowanych w pliku bazy danych. Komputer będący klientem będzie jedynie prezentował dane, umożliwił ich wprowadzanie oraz wstępną walidację.

Po zakończeniu instalacji na komputerze klienta należy zmodyfikować plik **Nemesis.ini** w sposób opisany w **pkt 5**.

7. Struktura plików programu

Program instaluje się w katalogu **c:\program files\info-r\Nemesis02**. W katalogu głównym programu znajdują się pliki aplikacji oraz narzędzi FireBird wykorzystywanych przez program. W tym katalogu znajduje się również plik konfiguracyjny **Nemesis02.ini**.

W katalogu **Baza** znajdują się pliki baz danych, plik **Nemesis02.gdb** zawiera główną bazę programu, plik **dokument.gdb** zawiera dokumenty przygotowane przez użytkownika.

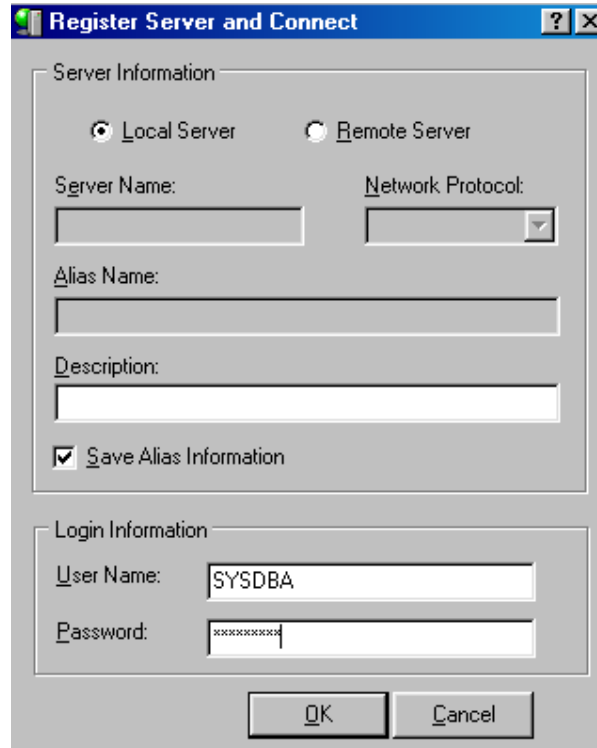
W katalogu **Dokumentacja** znajdują się plik z instrukcją obsługi ,inne informacje techniczne oraz akty prawne, rozporządzenia itp.

W katalogu **Szablon** znajdują się pliki szablonów dokumentów (przygotowanych przez producenta) wykorzystywanych w programie, zapisane w formacie RTF.

W katalogu **Kopia**, znajdują się pliki z kopią bazy danych, wykonywaną okresowo przez program po potwierdzeniu przez użytkownika. Uwaga! Kopia bazy danych powinna być również przechowywana na innym nośniku np. CD-ROM lub innym komputerze oraz należy ją wykonywać zawsze (!) na serwerze.

8. Weryfikacja i naprawa pliku bazy danych

1. Uruchomić program **IBConsole** przy użyciu polecenia **Start->Programy->InterBase->IBConsole**.
2. Jeżeli nie ma jeszcze zarejestrowanego serwera bazy danych to należy podświetlić na liście pozycje **InterBase Servers** i wybrać z menu pozycje **Server->Register ...** . Na ekranie pojawi się okno, które należy wypełnić jak na rysunku poniżej (w polu **Password** wpisać **masterkey**) i wybrać przycisk **OK**.



The screenshot shows a dialog box titled "Register Server and Connect". It has two main sections: "Server Information" and "Login Information".

Server Information:

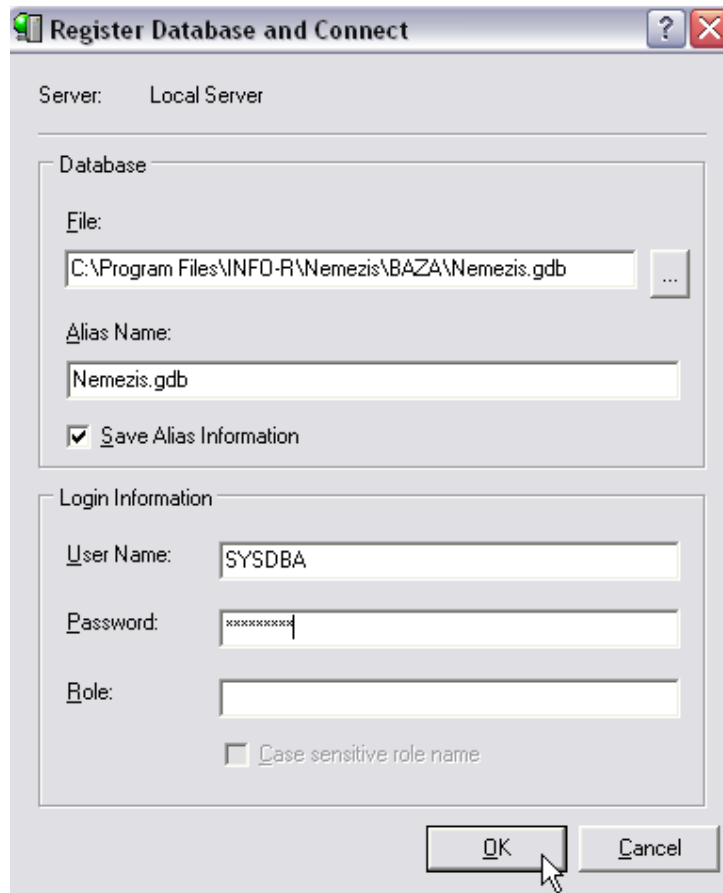
- Radio buttons for "Local Server" (selected) and "Remote Server".
- "Server Name:" text box (empty).
- "Network Protocol:" dropdown menu (set to "TCP/IP").
- "Alias Name:" text box (empty).
- "Description:" text box (empty).
- Checked checkbox for "Save Alias Information".

Login Information:

- "User Name:" text box containing "SYSDBA".
- "Password:" text box containing "masterkey".

Buttons for "OK" and "Cancel" are located at the bottom of the dialog.

3. Jeżeli mamy już na liście zarejestrowany serwer **Local Server** to należy się do niego zalogować poprzez dwukrotne kliknięcie jego nazwy lub podświetlenie jego nazwy i wybranie z menu pozycji **Server->Login ...** . W polu **User name** powinno być wpisane **SYSDBA**, w polu **Password** wpisać **masterkey** i wybrać przycisk **Login**.
4. Rozwinąć listę obiektów serwera poprzez kliknięcie znaku **+** obok jego nazwy.
5. Jeżeli nie mamy jeszcze na liście zarejestrowanej bazy danych, to należy podświetlić pozycje **Databases** i wybrać z menu polecenie **Database->Register ...** . Na ekranie pojawi się okno rejestracji bazy danych.

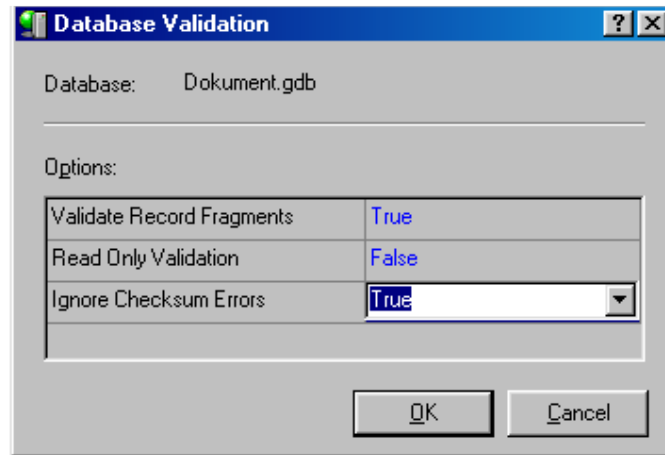


W polu **File** należy wpisać:

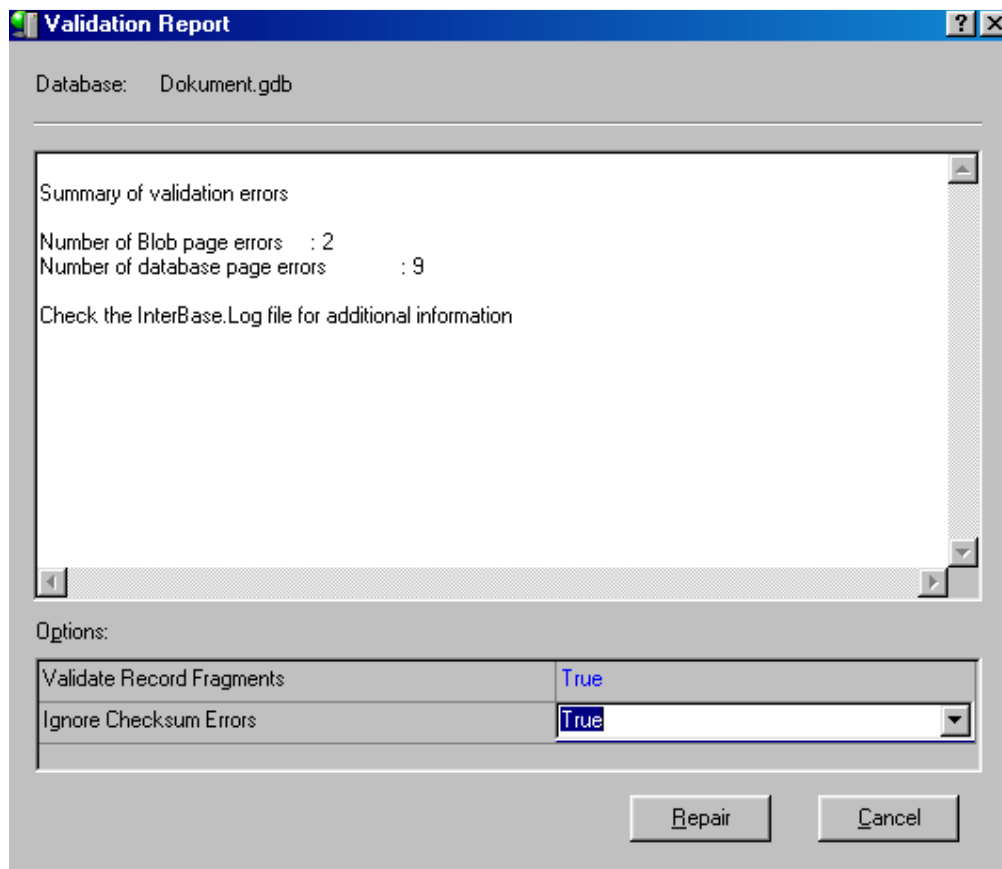
- *c:\Program Files\Info-r\Nemezis\Baza\Nemezis.gdb* dla bazy głównej programu
- *c:\Program Files\Info-r\Nemezis\Baza\Dokument.gdb* dla bazy dokumentów programu

W polu **Password** wpisać **masterkey** i wybrać przycisk **OK**.

6. W celu weryfikacji i naprawy pliku bazy danych należy podświetlić na liście nazwę bazy danych np. *Nemezis.gdb* i wybrać z menu polecenie **Database->Maintenance->Validation ...** (gdy polecenie jest nieaktywne należy najpierw wybrać z menu opcje **Database->Disconnect** i potwierdzić przyciskiem **Yes**). Po wybraniu opcji **Validation** na ekranie pojawi się następujące okno:

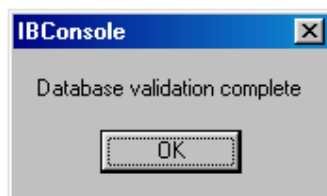


W opcji pierwszej i trzeciej ustawiamy wartość **True** poprzez wybranie jej z rozwijanej listy {patrz rysunek powyżej} i wybieramy przycisk **OK**. Na ekranie pojawi się kolejne okno :



Jeżeli plik bazy danych zawiera błędy to zostaną one wypisane w górnej części okna i aktywny będzie przycisk **Repair**. Jeżeli program nie stwierdzi błędów to zamykamy okno przyciskiem **OK** i kończymy pracę programu **IBConsole**.

Jeżeli plik jest uszkodzony to obu opcjom w dolnej części okna ustawiamy wartość **True** poprzez wybranie z rozwijanej listy i wybieramy przycisk **Repair**. Po udanym zakończeniu operacji na ekranie pojawi się następujący komunikat :



Zamykamy go przyciskiem **OK** i kończymy pracę programu **IBConsole**.

9. Jak wykonać kopię przyrostową bazy danych .

Tworzenie kopii przyrostowej polega na tym, że w kolejnych plikach archiwum zapisywane są tylko te rekordy, które zostały zmodyfikowane od momentu utworzenia ostatniej kopii zapasowej na określonym poziomie.

Planując tworzenie przyrostowych kopii zapasowych, określa się poziomy, na których będą tworzone kopie zapasowe. Kopie poziomu zerowego (najniższego) zawierają wszystkie rekordy w bazie danych. Kopie kolejnych poziomów (1, 2, ...) zawierają tylko te rekordy, które zostały zmienione od momentu wykonania ostatniej kopii zapasowej na poziomie niższym. Na przykład archiwum poziomu 1 zawiera tylko rekordy, które zmieniły się od czasu wykonania ostatniego archiwum poziomu 0.

Kopie przyrostowe umożliwiają częstszą archiwizację danych przy mniejszym zużyciu zasobów komputera. Przykładowy plan tworzenia kopii przyrostowych może uwzględniać:

- kopie poziomu **0** tworzone co tydzień;
- kopie poziomu **1** tworzone codziennie;
- kopie poziomu **2** tworzone co godzinę.

Wykonywanie kopii przyrostowych może odbywać się w czasie, gdy użytkownicy korzystają z bazy danych (w czasie pracy na bazie). Wykonywanie archiwum w czasie pracy nie zapewnia jednak 100% integralności danych w bazie. Prawdopodobieństwo utraty integralności danych jest bardzo niskie. Mimo to zalecane jest wykonywanie kopii w czasie, gdy aktywność użytkowników jest minimalna (np. o północy, w niedziele itd).

Do wykonywania przyrostowych kopii zapasowych służy program nbackup.exe, znajdujący się w folderze /bin firebirda. Kopie można wykonywać ręcznie z linii komend lub korzystając z programów typu harmonogram zadań.

Tworzenie kopii przyrostowej zalecane jest jedynie w przypadku bardzo dużej bazy danych, dla której tworzenie pełnej kopii wymaga bardzo długiego czasu.

Przed wykonaniem pierwszej kopii przyrostowej należy wykonać pełną kopię bazy danych.

Wykonanie pełnej kopii

Składnia polecenia dla wykonania pełnej kopii bazy danych :

```
nbackup -U <użytkownik> -P <hasło> -B 0 <baza> [<plik_archiwum>]
```

Przykład:

```
nbackup -U SYSDBA -P masterkey -B 0 C:\Program Files\INFO-  
R\Nemesis\Baza\Nemesis.gdb D:\Nemesis\Baza\Kopia\kopia_2012_01_01.nbk
```

-B 0 oznacza, że wykonujemy kopię poziomu 0, czyli jest to pełna kopia;

Przywracanie pełnej kopii

Składnia polecenia dla przywrócenia pełnej kopii bazy danych :

```
nbackup -R <baza> [<plik_archiwum>]
```

Przykład:

```
nbackup -R C:\Program Files\INFO-R\Nemezis\Baza\Nemezis.gdb  
D:\Nemezis\Baza\Kopia\kopia_2012_01_01.nbk
```

Wykonanie kopii przyrostowej

Aby wykonać kopię przyrostową, należy przy wykonywaniu kopii podać numer poziomu większy od zera. Kopia na poziomie N zawiera wszystkie zmiany od ostatniej kopii na poziomie N - 1 (nie można wykonać kopii na poziomie N, jeśli nie wykonało się kopii na poziomie N - 1).

Przykład:

Dzień po wykonaniu kopii poziomu 0 wykonujemy kopię poziomu 1:

```
nbackup -U SYSDBA -P masterkey -B 1 C:\Program Files\INFO-R\Nemezis\Baza\Nemezis.gdb  
D:\Nemezis\Baza\Kopia\kopia_2012_01_02.nbk
```

Kopia ta zawiera wszystkie zmiany z tego jednego dnia. Dzień później wykonujemy kolejną kopię poziomu 1:

```
nbackup -U SYSDBA -P masterkey -B 1 C:\Program Files\INFO-R\Nemezis\Baza\Nemezis.gdb  
D:\Nemezis\Baza\Kopia\kopia_2012_01_03.nbk
```

Kopia ta zawiera wszystkie zmiany z ostatnich dwóch dni. Tego samego dnia wieczorem wykonujemy kopię na poziomie 2:

```
nbackup -U SYSDBA -P masterkey -B 2 C:\Program Files\INFO-R\Nemezis\Baza\Nemezis.gdb  
D:\Nemezis\Baza\Kopia\kopia_2012_01_03_wieczorem.nbk
```

Kopia ta będzie zawierać tylko zmiany od utworzenia ostatniej kopii na poziomie 1 (czyli w tym przypadku z ostatnich kilku godzin).

Przykładowy plik bat zawierający wywołanie polecenia nbackup, wpisując e aktualną datę do nazwy pliku:

```
set d=%data:~0,4%%date:~5,2%%date:~8,2%_%time:~0,2%%time:~3,2%  
set katalog_archiwum=c:\Bazy\Archiwum\0  
set baza=c:\bazy\baza_danych.fdb  
"C:\Program Files\Firebird\Firebird_2_1\bin\nbackup.exe" -B 0 %baza%  
"%katalog_archiwum%\poziom0_%data%.nbk" -user SYSDBA -password masterkey
```

Przywracanie kopii przyrostowej

Przy przywracaniu kopii przyrostowej należy podać cały łańcuch przyrostowych kopii, zaczynając od poziomu 0, aż do poziomu, który chcemy odzyskać. Składnia wygląda następująco:

```
nbackup -R <baza> [<plik_archiwum_poziom_0> [<plik_archiwum_poziom_1>...]]
```

Dla przywrócenia kopii z poprzedniego przykładu, aby odzyskać dane na wieczór 2012-01-03 polecenie wyglądałoby następująco:

```
nbackup -R C:\Program Files\INFO-R\Nemezis\Baza\Nemezis.gdb  
D:\Nemezis\Baza\Kopia\kopia_2012_01_01.nbk  
D:\Nemezis\Baza\Kopia\kopia_2012_01_02.nbk  
D:\Nemezis\Baza\Kopia\kopia_2012_01_03.nbk  
D:\Nemezis\Baza\Kopia\kopia_2012_01_03_wieczorem.nbk
```