

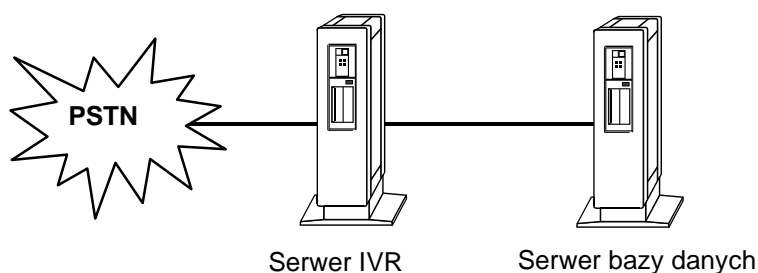
Bazy danych w zastosowaniu telekomunikacyjnym – Call Center i systemy audiotekstowe

Paweł Radziulis, prof. Czesław Jędrzejek
Instytut Technik Telekomunikacyjnych i Informatycznych Poznań
ul. Palacza 91A, 60-273 Poznań
e-mail: Pawel.Radziulis@itti.com.pl, Czeslaw.Jedrzejek@itti.com.pl

Abstrakt. W ostatnim czasie nastąpił szybki rozwój systemów automatycznych odpowiedzi głosowych (np. usługi telebankingu) oraz Call Center. Automatyczne funkcje wykonywane przez te systemy w większości przypadków opierają się na operacjach związanych z bazami danych. Bazy te przechowują parametry systemu wymagane do jego działania oraz tablice z danymi udostępnianymi bądź pobieranymi od klienta, w tym dane dźwiękowe. W obecnej chwili wdrażane są pierwsze internetowe Call Center, których głównym medium transmisji oraz pracy jest sieć IP. Bazy te cechują się wysoką niezawodnością oraz bardzo dużą wydajnością związaną z wymogami rzeczywistego czasu pracy powyższych systemów. W referacie przedstawione zostaną bazy danych w kontekście współdziałania z systemami telekomunikacyjnymi czasu rzeczywistego. Referat poparty jest doświadczeniem ITTI zdobytym podczas wdrażania powyższych systemów.

1. Wprowadzenie

W ostatnim czasie nastąpił szybki rozwój systemów automatycznej odpowiedzi głosowej (systemy audiotekstowe) oraz Call Center. Związane są one z usługami telekomunikacyjnymi na sieci inteligentnej, zwłaszcza z telebankingiem (na numery rozpoczynające się od 080x) oraz konkursami telefonicznymi (na numery rozpoczynające się od 0700). Są to systemy, charakteryzujące się tym, że w czasie całego połączenia klientem obsługiwany jest przez automat – bez ingerencji człowieka. Operacje wykonywane przez te systemy opierają się na interakcji z klientem. Poprzez klawiaturę swojego telefonu, klient steruje, oczywiście w udostępnionym zakresie, przebiegiem „rozmowy”. Na przykład po przejściu kolejnych menu oraz podanie liczbowego identyfikatora można odczytać stan konta w banku. Schemat systemu audiotekstowego przedstawia rysunek 1.

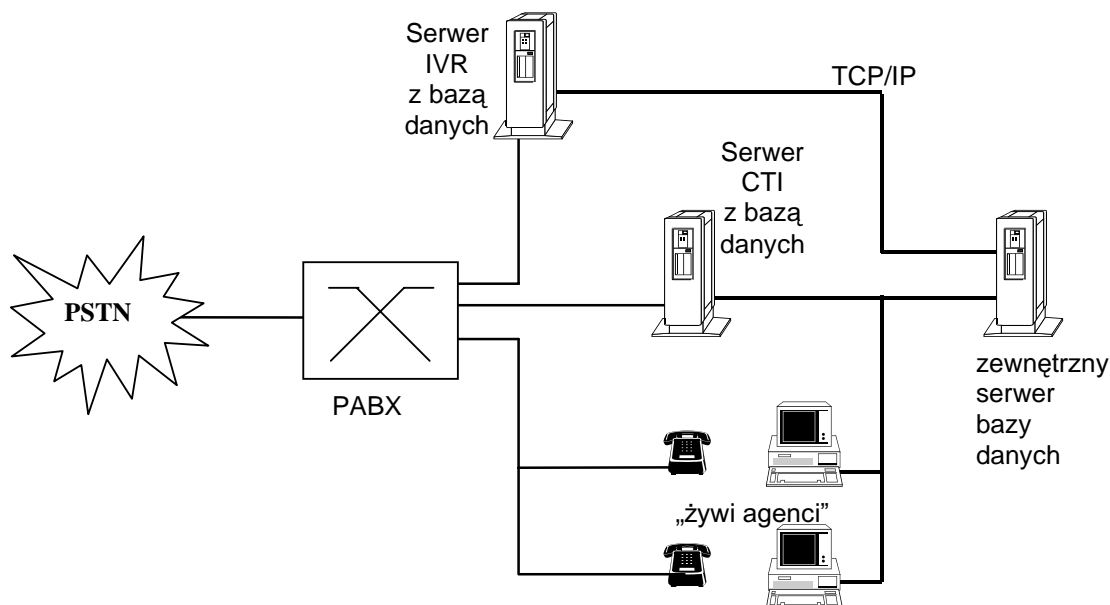


Rys. 1. Schemat systemu audiotekstowego

Rynek tych urządzeń podzielony jest na dwie grupy: urządzeń automatycznej odpowiedzi głosowej jako samodzielnych urządzeń oraz systemów budowanych w oparciu o karty komputerów klasy PC. Do pierwszej grupy zaliczają się produkty wszystkich najbardziej znaczących dostawców urządzeń telekomunikacyjnych jak Lucent (obecnie dział Call Center przechodzi do firmy Avaya Communications), Alcatel i Siemens. Najważniejszą dostawcą oferującą produkty drugiej grupy jest Dialogic (kontrolowany przez Intel).

Call Center to system, w którym podstawową jednostką nie jest już system automatycznych odpowiedzi, lecz tzw. „żywy agent”. „Żywy agent” to osoba obsługująca dzwoniącego klienta wyposażona w aparat telefoniczny oraz komputer, zintegrowane przez system CTI (*Computer*

Telephone Interface). Call Center zawiera w sobie system automatycznej odpowiedzi głosowej, który jest jego integralną częścią. System automatycznej odpowiedzi głosowej służy tu do wstępnej selekcji przychodzących połączeń i przekierowaniu ich do odpowiednich agentów. W tym samym czasie serwer CTI przesyła informacje do komputera agenta, do którego następuje przekierowanie, gdzie na ekranie pojawiają się dane o kliencie wraz z informacją o wybranej przez klienta usłudze



lub żądanej informacji.

Rys. 2. Schemat Call Center

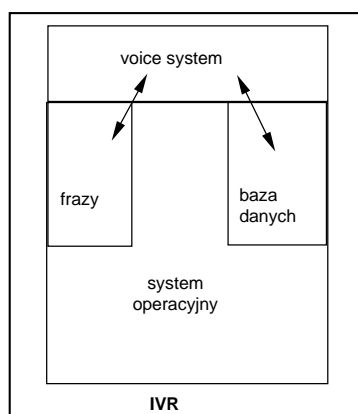
W najbardziej ogólnym przypadku przedstawionym na rysunku 2 systemy obsługi masowych wywołań mogą posiadać trzy typy baz danych:

- baza danych systemu automatycznej odpowiedzi głosowej na serwerze IVR;
- baza danych systemu zarządzania Call Center na serwerze CTI;
- zewnętrzna baza danych.

Najczęściej stosowane są bazy Oracle'a cieszące się wysoką niezawodnością wśród dostawców sprzętu (np. w przypadku produktów wiodących firm Altitude (Easy Phone) i Genesys (firma przejęta przez Alcatel)).

2.1. Baza danych systemu automatycznej odpowiedzi głosowej – serwer IVR

Baza danych systemu automatycznej odpowiedzi głosowej jest najbardziej „wyspecjalizowaną” bazą w omawianych systemach. Powiązana ściśle z systemem zarządzania głosem urządzenia IVR (*Interactive Voice Response*) stanowi jego integralną część. Jest to baza danych, której stawia się największe wymagania pod względem niezawodności i wydajności. Jest ona odpowiedzialna za odtwarzanie i zapis fraz (plików dźwiękowych) w systemie plików oraz przechowanie informacji o tych plikach w bazie danych. Architektura wewnętrzna IVR przedstawiona jest na rysunku 3. Wraz z zapisem plików baza koduje w swoich tablicach wskaźniki do nich, które umożliwiają późniejsze prawidłowe odczytywanie nagranych fraz. Oprócz zapisu dźwięku do bazy następuje zapis danych podanych przez klienta oraz danych dotyczących obsługiwanych połączeń, np.: godzina rozpoczęcia i zakończenia połączenia, nazwę uruchamianej usługi itp.

Rys. 3. Architektura wewnętrzna IVR (*Interactive Voice Response*)

Jak wcześniej wspomniano baza danych tworzy integralną całość z systemem głosowym. Podczas uruchomienia systemu głosowego nawiązywane są połączenia ze wszystkimi bazami danych, z których korzystają aplikacje. Jeżeli w czasie uruchamiania systemu głosowego baza danych nie będzie działała nie zostanie ona podłączona do systemu, a aplikacje z niej korzystające nie będą działały prawidłowo. Jedyną możliwością podłączenia takiej bazy danych jest ponowne uruchomienie systemu głosowego. System zachowuje się inaczej w przypadku, kiedy baza danych przestanie działać w trakcie jego pracy. Po pierwsze fakt ten zostanie wykryty tylko wtedy, gdy w czasie przestoju bazy danych, aplikacja zażąda do niej dostępu. W przypadku braku odpowiedzi aplikacja ponowi żądanie o dane po określonym czasie oczekiwania na odpowiedź. Jeśli aplikacja nadal nie uzyska odpowiedzi wygeneruje błąd.

Każdej aplikacji udostępniany jest proces interfejsu do bazy danych DIP (*Database Interface Process*), który zostanie omówiony nieco później. DIP jest jedynym interfejsem dostępnym dla aplikacji, dzięki któremu może połączyć się z bazą danych. Podczas procesu tworzenia aplikacji definiuje się tablice, do których dana aplikacja będzie miała dostęp. Dla każdej tablicy pojedynczej aplikacji tworzony jest w bazie danych kursor, przy pomocy którego aplikacja może zapisywać i odczytywać dane. Kursor ten istnieje tak długo jaka długo uruchomiona jest aplikacja i zajęty jest przez nią kanał rozmówny. System Oracle'a instalowany na urządzeniach IVR umożliwia otwarcie do 500 kursorów jednocześnie. Jest to ograniczenie wprowadzone dla utrzymania odpowiedniej wydajności systemu. W momencie, gdy limit ten zostanie przekroczony transakcje dokonywane na bazie danych mogą się nie powieść. Można wyznaczać maksymalną liczbę używanych kursorów, bądź ilość kursorów przypadających na jedną aplikację. Na przykład dla 100 kanałów (czyli maksymalnie uruchomionych 100 jednakowych aplikacji jednocześnie) maksymalna ilość możliwych do użycia tablic wynosi: $500 / 100 = 5$ tablic.

Najczęściej spotykane na rynku urządzenia IVR są samodzielnymi komputerami przemysłowymi opartymi na platformach UnixWare (wersja Unix'a) i posiadają ograniczone zasoby systemowe (szczególnie dyskowe). Obecnie wprowadzane są na rynek urządzenia pracujące pod systemem Windows posiadające architekturę umożliwiającą dalszą rozbudowę. Artykuł opiera się o doświadczenia zebrane podczas pracy na urządzeniach IVR pracujących na systemie UnixWare.

Systemy te udostępniają dwie podstawowe grupy kontroli zasobów bazy danych Oracle:

- zmiana parametrów zasobów dyskowych (*storage size*)

System głosowy urządzenia IVR umożliwia wykonanie dwóch czynności. Pierwsza to zmiana ilości gromadzonych danych dotyczących realizowanych połączeń. Polega ona na zmniejszeniu do minimum informacji zapisywanych przez aplikację i jest jednym z kroków projektowania aplikacji dla systemu. Druga to skrócenie okresu, przez który przechowywane są dane. Dokonuje się tego przez ustawienia odpowiednich zmiennych. Na przykład dla

systemu Conversant firmy Lucent zmienne te powinny wyglądać następująco [1]:

```
$VSUTIL/cdhsu
```

```
$VSUTIL/cdhdel -x
```

```
$VSUTIL/ccasu
```

```
$VSUTIL/ccadel -x ,
```

gdzie x jest liczbą dni, przez które dane mają być przechowywane.

- operacje strojenia bazy danych, takie jak zmiana ilości pamięci współdzielonej (*shared pool size*) oraz operacje na segmencie wycofań (*rollback segment*).

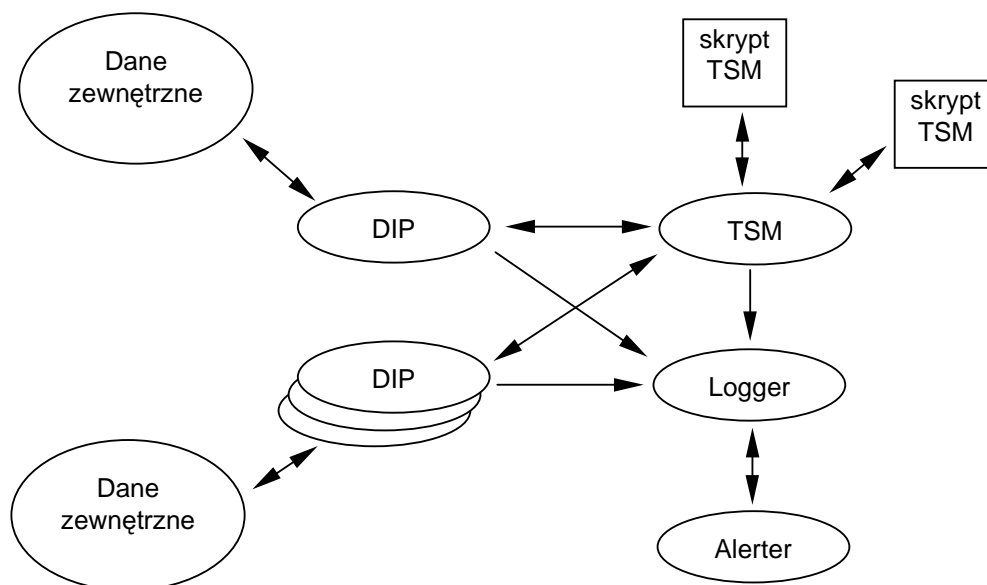
W systemach IVR liczba jednocześnie uruchomionych aplikacji równa jest ilości kanałów rozmównych, jakie mogą zostać obsłużone przez urządzenie. Posiadając informacje o operacjach, liczbie poszczególnych operacji wykonywanych przez aplikacje oraz określoną liczbę maksymalnie uruchomionych aplikacji, administrator bazy danych może dość dokładnie określić minimalne wartości parametrów dla bazy danych.

Systemy powyższe pracują jako systemy czasu rzeczywistego. Klient podczas rozmowy nie odczuwa żadnych opóźnień w dostarczaniu mu danych głosowych – rozmowa jest płynna. Czasy odpowiedzi z bazy danych są rzędu milisekund. Dla zapewnienia takiej pracy, nawet dla 100 rozmów jednocześnie, systemom IVR wystarcza następująca konfiguracja (przy wykorzystaniu kart z jednym procesorem DSP na 30 kanałów rozmównych):

- procesor główny Pentium 200
- 512MB pamięci RAM
- system dysków SCSI.

2.1.1. DIP – Data Interface Process

Każda aplikacja stworzona dla systemu IVR jest interpretowana i wykonywana przez TSM (*Transaction State Machine*) w postaci skryptów. Skrypty te kontrolują cały proces obsługi połączenia poprzez takie funkcje jak odbiór nadchodzącego telefonu bądź zbieranie wprowadzanych cyfr przez klawiaturę numeryczną telefonu. Jednakże nie potrafią one same wykonać funkcji związanych z dostępem do danych zewnętrznych w bazie i plikach lub skomplikowanych obliczeń numerycznych. Funkcje te dostarczane są do TSM przez DIP. Procesy DIP udostępniają wszystkie funkcjonalności możliwe do oprogramowania w języku C, w którym są tworzone oraz funkcjonalności systemu operacyjnego możliwe do oprogramowania pod tym językiem.



Rys. 4. Architektura Data Interface Process [1]

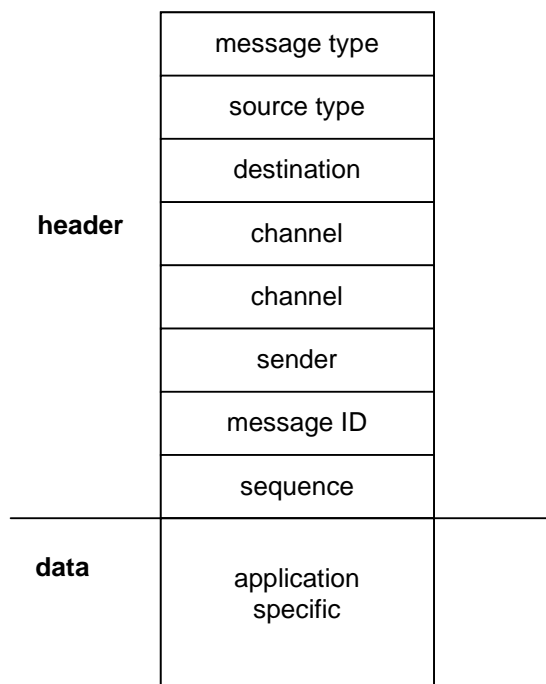
Rysunek 4 przedstawia sposób powiązania procesów interfejsów danych z TSM. Proces taki oprócz interakcji z TSM jest powiązany z zewnętrznymi zasobami (bazy danych, pliki danych), modułem tworzenia dziennika zdarzeń, logów (Logger) oraz alarmów (Alerter).

DIP jest procesem przetwarzania wiadomości. Oznacza to, że wykonuje on akcje związane ze swoją funkcjonalnością dopiero po otrzymaniu wiadomości z prośbą o jej wykonanie. Po jej otrzymaniu wiadomość jest przetwarzana, a wynik operacji przekazywany jest do skryptu TSM, który zażądał wcześniej przetworzenia danych zewnętrznych. Liczba procesów DIP w systemie jest z góry określona przez jego zasoby (zbyt duża ich ilość może doprowadzić do spowolnienia działania systemu), a dostęp do nich odbywa się przez kolejki wiadomości (*Message Queues*). Każda taka kolejka działa jak skrzynka pocztowa przechowując żądania oraz odsyłając odpowiedzi do odpowiednich skryptów TSM.

W systemie występują dwa typy procesów DIP: predefiniowane oraz dynamiczne (DynaDIP). Różnią się one sposobem przypisania procesowi kolejki wiadomości, zaszytą w ich kodzie. Procesy DIP odwołujące się do tej samej kolejki wiadomości mogą wysyłać informacje oraz współdziałać ze sobą. Zanim zaczną się wykonywać procesy dynamiczne DIP żądają od systemu przydzielenia wolnej kolejki wiadomości od systemu głosowego. Procesy te nie znają numeru obsługiwanej skrzynki wiadomości do momentu uruchomienia systemu głosowego IVR i „żyją” tak długo jak włączony jest system głosowy.

Ponieważ procesy DIP są procesami przetwarzania wiadomości muszą one być w stanie wysłać i odebrać wiadomość z TSM oraz wysłać wiadomości o błędach do systemu logów. Struktura i działanie predefiniowanych oraz dynamicznych procesów DIP jest identyczna. Wiadomości pomiędzy nimi a skryptami TSM przesyłane są w postaci pakietów.

Pakiet wiadomości dzieli się na dwie części: nagłówek i dane. Obie te części posiadają swoją specyfikację w języku C. Nagłówek zawiera informacje o adresacie oraz nadawcy wiadomości, dane dotyczące kanałów rozmównych systemu IVR, numery wiadomości. Schemat pakietu wiadomości przedstawiony jest na rysunku 5.

Rys. 5. Komponenty pakietu wiadomości **mbhdr** [1]

Struktura nagłówka wiadomości posiada następującą postać [1]:

```
struct mbhdr {
    long    mtype; /* Message type */
    short   irType; /* Source type for message */
    ir_key_t irWhoTo; /*Destination queue or channel owner */
    long    irChan; /* Channel number */
    long    mchan; /*Channel number */
    short   morig; /*Sender's Qkey */
    short   mcont; /* Message id */
    unsigned short mseqno; /*Message sequence number */
};
```

Kolejne pola oznaczają:

- **mtype** – pole wykorzystywane w przypadku przesyłania wiadomości pomiędzy procesami DIP działających na tej samej kolejce zadań;
- **irType** – pole definiujące źródło wiadomości, ustawiane jest przez funkcję wysyłającą wiadomość;
- **irWhoTo** - pole określające kolejkę wiadomości, do której ma być przesłana odpowiedź;
- **irChan** – pole oznaczające numer kanału rozmównego systemu IVR;
- **mchan** – pole wypełniane przez TSM oznaczające numer kanału rozmównego systemu IVR wykorzystywanego przez aplikację uruchamiającą proces DIP;
- **morig** – pole określające numer kolejki wiadomości, z której wysłana jest wiadomość;
- **mcont** – pole informujące o typie danych przekazywanych pomiędzy TSM a procesem DIP;
- **mseqno** – pole do dodatkowego oznaczania wiadomości unikalnym identyfikatorem.

Część zawierająca dane następuje tuż po nagłówku. Nie posiada ona ograniczonej z góry wielkości a jej format definiowany jest przez nagłówek (pole *Message Id*). Przykładowa struktura wiadomości przedstawiona jest poniżej:

```
struct stockInfo {
    struct mbhdr hd;
    char callerName [30];
    int callerId;
};
```

2.2. Baza danych systemu CTI – serwer CTI

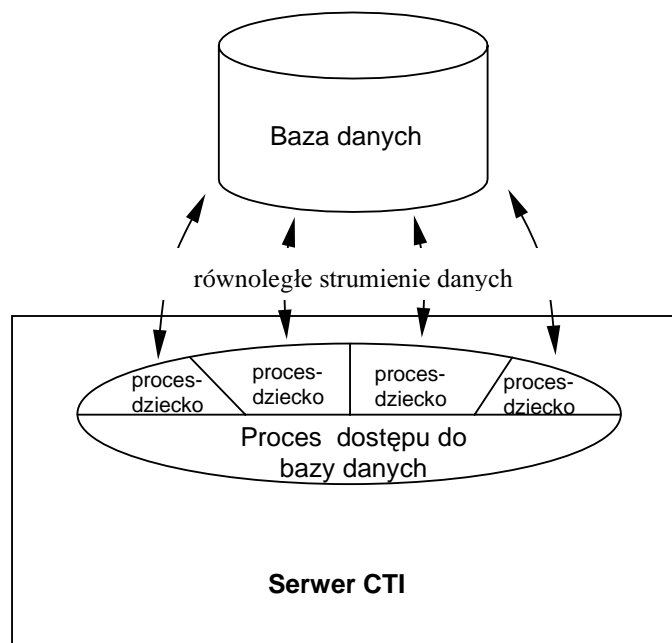
Baza danych systemu CTI (*Computer Telephony Interface*) przechowuje informacje konieczne do prawidłowego działania Call Center wliczając w to:

- zmienne konfiguracyjne,
- informacje o kampaniach,
- dane o agentach i klientach,
- konfigurację centrali telefonicznej,
- informacje o obsłużonych połączeniach,
- statusy aktualnie obsługiwanych połączeń,
- dane gromadzone podczas rozmowy z klientem (system CTI umożliwia transfer tych danych wraz z rozmową pomiędzy agentami).

Serwer CTI działa jako łącznik pomiędzy centralą obsługującą transmisję dźwięku oraz bazą danych i komputerami agentów obsługujących transmisję danych. Dzięki zawartości bazy danych serwer CTI umożliwia wyświetlanie na ekranie komputerów agentów danych klienta, z którym prowadzona jest rozmowa. Cała aplikacja, przy pomocy której agent obsługuje połączenie znajduje się na komputerze lokalnym agenta. Dane o kliencie zaś pobierane są z bazy danych w wyniku rozpoznania numeru telefonu osoby dzwoniącej (warunkiem tego jest przekazywanie rozmowy pomiędzy klientem a centralą Call Center przez cyfrowe centrale tranzytowe).

Serwer ten musi przechowywać w swojej bazie danych nie tylko informacje o połączeniach, ale także o elementach Call Center takich jak kampanie, agenci, aktualnie dzwoniący klienci. Średnio dla przechowania 1000 kontaktów wymagane jest 360 kB na dane osobowe klientów oraz po 1 kB na każdego klienta do przechowywania danych biznesowych (związanych z historią obsługi klienta przez Call Center). Do tego należy doliczyć około 1MB na każde tysiąc przeprowadzonych rozmów, co jest związane z rejestracją tych rozmów. Z tego powodu serwery CTI wraz z bazą danych wymagają mocniejszych maszyn oraz większych pamięci dyskowych niż systemy automatycznej odpowiedzi głosowej. Typowa konfiguracja dla 200 agentów to serwer z podwójnym procesorem klasy Pentium III 500, 512 RAM oraz kilkanaście GB pamięci dyskowej SCSI.

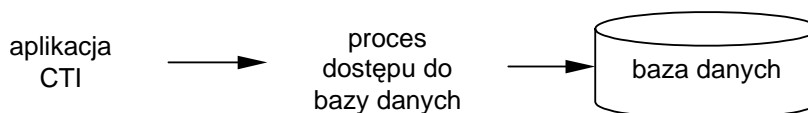
Serwery CTI są w stanie obsługiwać ponad 1000 agentów jednocześnie. Wąskim gardłem może w tym przypadku okazać się szybki i równoczesny dostęp wielu agentów do zasobów zgromadzonych w bazie danych. W celu przyspieszenia oraz zrównoleglenia dostępu do baz danych na serwerze uruchamiany jest osobny proces (serwer aplikacji) zarządzający połączeniami do bazy danych. Dla każdej transakcji dokonywanej na bazie danych tworzy on osobny proces-dziecko, jak na rysunku 6. Procesy te mogą być uruchamiane na komputerach innych niż ten, na którym jest zainstalowany system CTI. Osiągamy wówczas wzrost wydajności systemu poprzez zaangażowanie większych zasobów zewnętrznych.



Rys. 6. Dostęp do baz danych serwera CTI

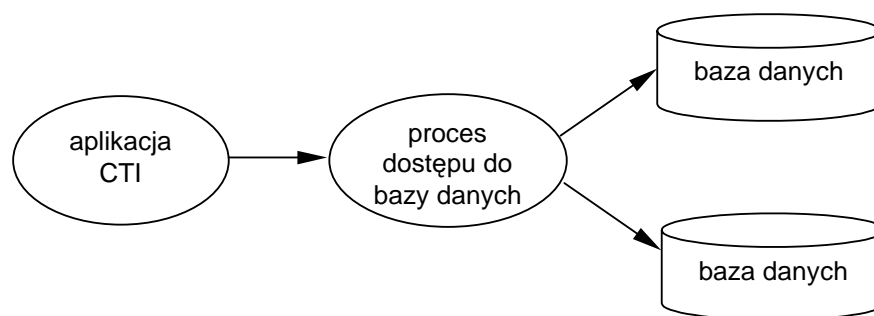
Baza danych przekazuje dane do procesu-dziecka jak do osobnych klientów umożliwiających jednoczesną obsługę wielu strumieni danych. W efekcie działania tego rozwiązania podnoszona jest szybkość przetwarzania danych oraz eliminuje się opóźnienia powstałe w wyniku szeregowej transmisji dużych ilości danych. W przeciwieństwie do systemu automatycznej odpowiedzi głosowej, rozmowa prowadzona jest tu przez agenta, który odczytuje tekst wyświetlany na ekranie. Baza danych wykorzystywana jest przez agenta tylko do pobierania z niej informacji o kliencie i informacji o kampanii, którą prowadzi, np. liczby jednostek danego produktu znajdującego się w danej chwili na stanie magazynu. Zatem agent wykonuje rzadkie transakcje na bazie danych, dla których odpowiedź może posiadać nawet sekundowe opóźnienia. W tym przypadku najważniejsze stają się zagwarantowanie równoległego dostępu do bazy danych. Opierając się na kryterium szybkiego i równoległego dostępu do danych, posiadanej mocy obliczeniowej komputerów oraz liczbie i różnorodności zewnętrznych i wewnętrznych baz danych, serwer dostępu do danych można skonfigurować na kilka sposobów:

- połączyć aplikację z jedną bazą danych poprzez jeden proces dostępu do bazy danych;



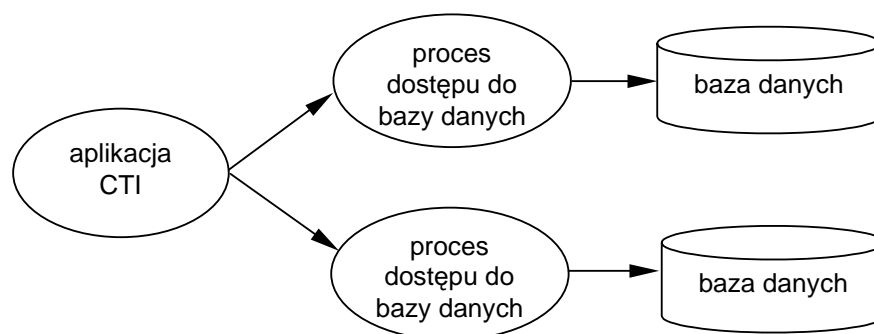
Rys. 7. Dostęp do bazy danych

- połączyć aplikację z kilkoma bazami danych poprzez jeden proces dostępu do bazy danych;



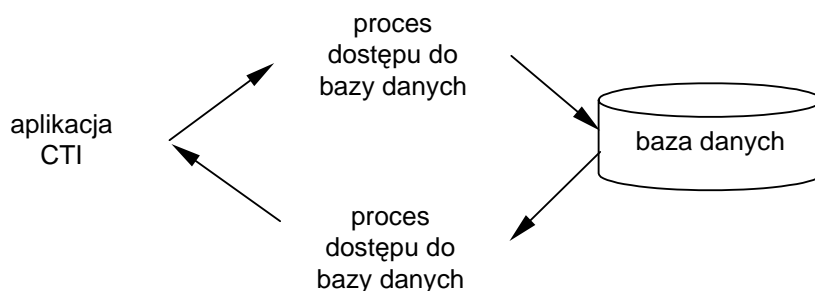
Rys. 8. Dostęp do baz danych przez jeden proces dostępu

- połączyć aplikację z kilkoma bazami danych poprzez kilka procesów dostępu do bazy danych;



Rys. 9. Dostęp do baz danych poprzez kilka procesów dostępu

- połączyć aplikację z jedną bazą danych poprzez kilka procesów dostępu do bazy danych.



Rys. 10. Dostęp do bazy danych poprzez kilka procesów dostępu

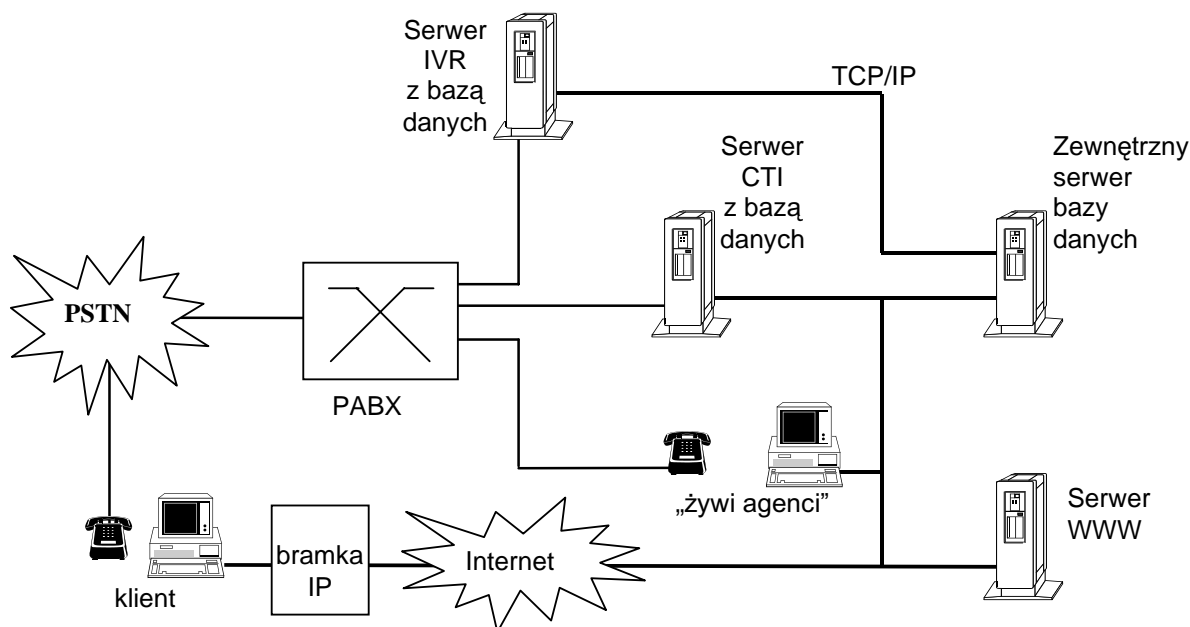
3. Podsumowanie

W artykule przedstawiono dwa typy baz danych występujących w systemach automatycznych odpowiedzi głosowych i Call Center: bazę danych systemu automatycznej odpowiedzi głosowej IVR oraz bazę danych serwera CTI zarządzającego Call Center. Pierwsza z nich to baza zarządzająca systemem głosowym. Odtwarzanie nagranych fraz dźwiękowych, ich zapis oraz

zarządzanie nimi stawiają przed nią wysokie wymagania niezawodności oraz szybkości działania. Jest to typowa baza OLTP (OnLine Transaction Processing). Druga baza serwera CTI wykorzystywana jest nie tylko do przechowywania danych, ale także do zarządzania całym Call Center. Przed tą bazą stawiane są wymogi dużej mocy obliczeniowej, koniecznej do przetwarzania jednocześnie informacji zarządzających oraz zapytań do bazy danych generowanych przez agentów.

W obu systemach audiotekstowym oraz Call Center spotykane są także zewnętrzne bazy danych, należące do systemu, które działają na osobnym serwerze bazy danych. Są to bazy danych dedykowane do skomplikowanych obliczeń oraz do przechowywania dużych ilości danych (tablic o milionach rekordów, np. jak baza klientów w banku). Najczęściej wykorzystywane są one do przygotowywania oraz udostępniania gotowych informacji dla Call Center bądź systemu audiotekstowego. Skomplikowane operacje na danych i ich analiza oraz zapytania na dużych ilościach danych są operacjami potrzebującymi dużych zasobów systemowych. Wielkość zasobów do wykonania tych operacji może okazać się tak duża, że spowolni to pracę całego systemu odpowiedzi głosowych, bądź wykonanie tych operacji będzie trwało przez czas nie akceptowalny w systemie czasu rzeczywistego. Dlatego też w przypadkach takiego zagrożenia dla systemu czasu rzeczywistego stosuje się zewnętrzne bazy danych.

W roku 2000 wdrożone zostały pierwsze Internetowe Call Center, których głównym medium transmisji oraz pracy jest sieć IP (patrz rysunek 11).



Rys. 11. Schemat internetowego Call Center

W systemach tym oprócz transmisji głosu do i od klienta prowadzona jest także transmisja danych. Transmisja danych odbywa się poprzez Internet lub prywatną sieć IP. Dzięki temu agenci zyskują możliwość wizualizacji informacji na komputerze klienta, co uatrakcyjnia oferowane usługi oraz znacznie ułatwia pracę agentom. Wśród tych informacji znajdują się także dane gromadzone na innych serwerach bazy danych znajdujących się w systemie. Porównując ze zwykłym Call Center, dla takiego przypadku serwery baz danych muszą być przygotowane do przyjęcia zwiększonej liczby zapytań.

Jak wcześniej wspomniano systemy automatycznej odpowiedzi głosowej są systemami o architekturze zamkniętej. Producenci tego sprzętu zadbali o to, by przy swoich ograniczeniach spełniał on swoje funkcje należycie. Służą temu między innymi specjalne procesy interfejsów dostępu do baz danych DIP (*Data Interface Process*) bądź programowe serwery aplikacji dostępu do baz danych. Systemy audiotekstowe oraz Call Center spełniają coraz większe wymagania. Są one

zasadniczym elementem systemów obsługi klientów CRM (Customer Relationship Management). Cena najbardziej zaawansowanych systemów CRM opartych na Call Center np. firmy Siebel sięga 1 mln USD.

Bibliografia

1. Dokumentacja: Intuity™ Conversant® System Documents – Lucent, maj 2000
2. Dokumentacja firm Lucent, Alcatel, Dialogic oraz Siemens.