

Nowoczesne architektury systemów telemedycznych

Łukasz Czekerda, Joanna Kosińska, Paweł Rzepa

Katedra Informatyki EAIiE, Akademia Górniczo-Hutnicza

al. Mickiewicza 30, 30-059 Kraków, e-mail: {luke,kosinska,rzepa}@agh.edu.pl

STRESZCZENIE

Niniejszy artykuł podejmuje problem tworzenia nowoczesnych systemów telemedycznych. Przedstawia propozycję jego rozwiązania w oparciu o istniejące technologie związane z środowiskami udostępniania usług oraz systemami zorientowanymi na WWW. Tymi technologiami są m.in.: język Java, Jini, Jiro, Servlety, JSP, XML, EJB oraz wiele innych. Użycie ich zostało przedstawione w kontekście projektów zrealizowanych w Zespole Systemów Rozproszonych Katedry Informatyki AGH.

1 WSTĘP

Ogromny rozwój i upowszechnienie się w ostatnich latach technologii teleinformatycznych przyczyniły się do stworzenia możliwości wykorzystania globalnej infrastruktury wymiany informacji dla potrzeb medycznych. Zastosowanie sieci komputerowych oraz komórkowych do przekazu informacji umożliwiło zdalną wymianę danych pomiędzy personelem medycznym, a także zdalny dostęp pacjentów do informacji medycznej. Wymiana opinii pomiędzy specjalistami nabiera charakteru multimedialnego przybierając formę telekonsultacji. Co więcej w takich telekonsultacjach pojawiła się możliwość przesyłania różnego rodzaju potrzebnych obu stronom informacji w postaci statycznych obrazów (np. cyfrowych zdjęć RTG) oraz sekwencji wideo rejestrujących dynamicznie procesy życiowe pacjenta. Dostęp do tej informacji pozwala na stawianie diagnoz lekarzowi, który może znajdować się w ośrodku referencyjnym odległym o setki kilometrów. Z tego względu systemy telemedyczne muszą być oparte o model architektury rozproszonej.

Wprowadzenie właściwych rozwiązań w zakresie wyboru technologii oprogramowania z pewnością pozwoli na upowszechnienie profesjonalnych

systemów medycznych i spopularyzuje telemedycynę, a środki wydatkowane na nią będą mogły być traktowane jak długofalowa inwestycja przynosząca korzyści zarówno lekarzom jak i pacjentom.

Dalsza część niniejszego artykułu zarysowuje możliwości systemów telemedycznych, opisuje wymogi stawiane architekturom takich systemów oraz podaje przykłady aplikacji wykonanych przez autorów w ramach projektu badawczego 6WINIT¹ realizowanego w ramach 5 Programu Ramowego Unii Europejskiej.

2 MOŻLIWOŚCI NOWOCZESNEGO SYSTEMU TELEMEDYCZNEGO

Rozwój współczesnych systemów telemedycznych przebiega równocześnie w wielu kierunkach. Podstawowym wymogiem telemedycyny jest istnienie właściwych metod dla cyfrowej reprezentacji informacji medycznej oraz jej udostępniania.

Standardem cyfrowym szeroko stosowanym w bardzo wielu ośrodkach medycznych na całym świecie dla reprezentacji wyników badań radiologicznych jest standard DICOM². Plik zgodny z tym standardem oprócz cyfrowych obrazów o wysokiej, diagnostycznej rozdzielczości, zarówno statycznych, jak i złożonych z wielu klatek (np. badania hemodynamiczne), zawiera wiele dodatkowych informacji pomocnych personelowi medycznemu – takich m.in. jak dane pacjenta i parametry wykonywanego badania [11].

¹ Głównym celem projektu jest prowadzenie badań w zakresie stosowania protokołu IPv6 w środowiskach bezprzewodowych.

² Digital Imaging and Communication in Medicine

Badania medyczne ze względu na swoją specyfikę wymagają ogromnych przestrzeni dyskowych (szacuje się, że średniej wielkości szpital w USA wymaga rocznie kilkunastu terabajtów pamięci masowej). Niezbędne jest w związku z tym stosowanie efektywnych metod ich wyszukiwania oraz przesyłania na odległość. Zadania te są realizowane przez bazy danych oraz protokoły transportu informacji medycznej (DICOM definiuje nie tylko format pliku lecz i protokoły komunikacji pomiędzy aplikacjami medycznymi).

Kolejnym aspektem związanym z telemedycyną są telekonsultacje wymagające w najprostszym podejściu jakości wideokonferencyjnej. Jednakże w wielu przypadkach konieczne jest przekazanie wysokiej jakości obrazu (np. chorego pacjenta) co wymaga z kolei wysokiej jakości sprzętu multimedialnego połączonego z urządzeniami diagnostycznymi a to z kolei wprowadza duże wymagania wobec przepustowości sieci oraz urządzeń uczestniczących w transmisji.

Jednym z kierunków, w jakim podąża współczesna telemedycyna jest zapewnienie dostępu do informacji medycznej w dowolnym czasie i z dowolnego miejsca (na przykład z miejsca wypadku lub z domu chorego) przy wykorzystaniu urządzeń przenośnych klasy laptop i PDA. Dlatego też pożądanym jest szybki rozwój sieci komórkowych przystosowanych do transmisji danych (standard GPRS, a docelowo UMTS).

3 ARCHITEKTURA I TECHNOLOGIE

Wymagania sprzętowe wobec systemu telemedycznego są ogromne, począwszy od nowoczesnych urządzeń, poprzez szerokopasmowy dostęp do sieci aż do dużych pamięci dyskowych. Niemniej ważne są jednak wymagania architektoniczne.

Spektrum zastosowań telemedycyny jest tak szerokie, że zdefiniowanie jednej architektury dla wszystkich systemów telemedycznych jest bardzo trudne. Można jednak przyjąć, że skoro w większości tego typu systemów wymagana jest komunikacja pomiędzy jego elementami lub zdalny dostęp do danych to korzystnie będzie zastosować architekturę rozproszoną. Ważnym jest również, aby w tej architekturze zastosować warstwową i modułarną budowę, która zwiększa przejrzystość systemów oraz możliwości ich rozwoju i zastosowań.

Właściwym dla większości nowoczesnych systemów jest używanie przy dostępie tzw. *lekkiego klienta* charakteryzującego się ograniczeniem funkcjonalności przede wszystkim do zadań prezentacyjnych. Logika przetwarzania aplikacji

jest przeniesiona do innych warstw systemu, przez co staje się osiągalna dla szerokiego zakresu klientów i dzięki temu dobrze wpisuje się w potrzebę ogólnej dostępności usług telemedycznych.

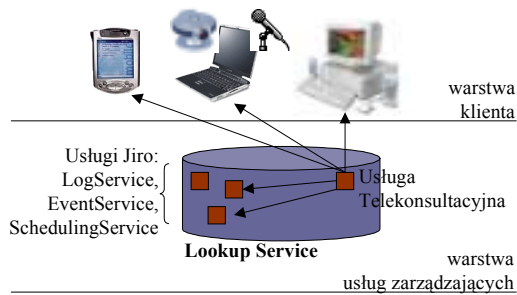
W zależności od stopnia skomplikowania systemu oprócz warstwy klienta wprowadza się dodatkowo jedną lub więcej warstw. Sposób podziału funkcjonalności na warstwy jest w znacznym stopniu uzależniony od zadań, jakie stoją przed aplikacjami. Dlatego też charakterystyka wielowarstwowych architektur rozproszonych zostanie omówiona w kontekście dwóch typów systemów — udostępniających usługi i systemów zorientowanych na WWW (ang. *web-centric*).

Przy rozważaniu architektury systemu należy mieć także na uwadze ogromną rangę kwestii zachowania poufności danych medycznych i zapewnienia dostępu do nich jedynie przez uprawnione osoby. Wykorzystać tu można z powodzeniem mechanizmy zapewniania bezpieczeństwa stosowane np. w bankowości elektronicznej, łącząc szyfrowanie transmisji z opartymi na infrastrukturze klucza publicznego metodami autentykacji.

Udostępnianie usług

Typowe rozwiązania multimedialne oferowane przez takie urządzenia jak terminale H.323 implementowane w popularnych klientach (np. NetMeeting), mogą być wykorzystywane do przeprowadzania telekonsultacji. Posiadają one jednak pewne niedogodności, są zależne od platformy programowej jak i sprzętowej a dodatkowo współpracują tylko i wyłącznie z rozwiązaniami jednego producenta. Alternatywnym podejściem jest możliwość oparcia systemu wideokonferencyjnego o technologie Jini [6], Jiro [4]. Podejście to gwarantuje bogate możliwości zarządzania systemem, oprogramowaniem oraz sesjami audio-wideo. Aplikacje wykorzystujące tą technologię są dodatkowo niezależne od platformy (gdyż wykorzystują język Java) i mogą być używane na większej liczbie systemów operacyjnych. Przykładami typowych usług zrealizowanych w tym systemie mogą być przeprowadzanie wideokonferencji między lekarzami czy współdzielenie wirtualnej tablicy np. służącej do konsultacji zdjęć medycznych.

Umiejętne zastosowanie nowoczesnych rozwiązań technologicznych przyczynia się do znacznych uproszczeń w modelu architektury systemów telekonsultacyjnych. Przykładowa architektura została przedstawiona na rysunku 1.



Rysunek 1 Model systemu telekonsultacyjnego

Warstwę najniższą stanowią usługi, które pozwalają zarządzać przebiegiem poszczególnych telekonsultacji. Główną ideą systemu jest wykorzystanie przenośności języka Java oraz opartej na nim architektury rozproszonej Jini. Podstawy Jini skupiają się wokół nowych koncepcji – protokoły wyszukiwania usług (ang. *discovery protocols*), repozytorium usług (ang. *Lookup Service*), rezerwacja zasobów (ang. *leasing*), zdalne zdarzenia (ang. *remote events*) [5]. Najważniejszym komponentem w systemie Jini jest *Lookup Service*, w którym muszą być zarejestrowane wszystkie usługi dostępne w danym rozwiązaniu, w szczególności usługa telekonsultacyjna. Rejestracja polega na przesłaniu obiektu *proxy* (czyli obiektu, który pośredniczy w komunikacji między daną usługą a klientem chcącym ją wykorzystać) do *Lookup Service*, skąd będzie on ładowalny na żądanie klienta. Klienci wyposażeni jedynie w specjalne aplikacje umożliwiające przeglądanie zainstalowanych w *Lookup Service* usług (ang. *Lookup Service Browser* [9]) mogą pobierać za pomocą HTTP kod (*proxy*) będący interfejsem do usługi. Ważną zaletą systemu telekonferencyjnego opartego o środowisko Jini jest to, iż lekarze nie muszą mieć zainstalowanego na swoich urządzeniach oprogramowania przechwytyjącego obraz i dźwięk, jest ono ładowane wraz z kodem *proxy*. Oprogramowanie to, w zależności od jego funkcjonalności może być stworzone przy użyciu *Java Media Framework (JMF) API* [10], które wspiera obsługę urządzeń audio-wideo, kodowanie i transmisję strumieni multimedialnych, czy też *Java Shared Data Toolkit (JSDT)* przeznaczone dla współpracujących aplikacji współdzielących dane (tzw. „whiteboard”).

Oprogramowanie to, czyli obsługa urządzeń audio-wideo, kodowanie i transmisja strumieni multimedialnych stworzone jest przy użyciu *Java Media Framework (JMF) API*.

Realizacja funkcjonalności związanej z zarządzaniem wszystkimi komponentami oraz występującymi w systemie telekonsultacyjnym zdarzeniami jest oparta o technologię Jiro. Jiro to rozszerzenie możliwości Jini o aspekty związane z zarządzaniem. Dostarczone gotowe usługi wspierają m.in. archiwizację różnych istotnych informacji

(*Log Service*), dystrybuowanie generowanych zdarzeń (*Event Service*), uruchamianie zadań w określonych momentach czasu (*Scheduling Service*).

Systemy zorientowane na WWW

Do budowy systemów zorientowanych na WWW stosuje się model trójwarstwowy, który dzieli funkcjonalność pomiędzy warstwę przechowywania danych, przetwarzania danych oraz klienta.



Rysunek 2 Model systemu zorientowanego na WWW

Zadaniem warstwy klienta jest korzystanie z funkcjonalności udostępnianej przez warstwę przetwarzania danych, co zwykle ogranicza się do interakcji z użytkownikiem poprzez prezentowanie mu danych oraz reagowanie na jego polecenia. Najczęściej jako sposób komunikacji z użytkownikiem wybiera się aktywne, dynamiczne strony WWW. Zaletą takiego rozwiązania jest fakt, iż większość systemów operacyjnych posiada wbudowane przeglądarki WWW a więc nie jest potrzebne instalowanie na urządzeniach końcowych żadnego dodatkowego oprogramowania. Dodatkowo w takim rozwiązaniu zoptymalizowany jest sposób zarządzania systemem, gdyż modyfikacja oprogramowania nie wymaga prac instalacyjnych na urządzeniach klienckich. Ze względu na ograniczenia, jakie narzuca HTML nie jest to jednak rozwiązanie, które można zastosować w każdym przypadku. Jeśli potrzebny jest klient będący samodzielną aplikacją, to korzystnie jest wtedy zastosować omówioną wcześniej technologię Jini.

Warstwa przetwarzania danych koncentruje się na logice działania systemu. Musi ona obejmować usługi takie jak równoważenie obciążenia, zapewnienie ciągłości dostępu czy bezpieczeństwo. Ponieważ jest to warstwa najbardziej skomplikowana, w przypadku bardziej złożonych systemów dzieli się ją na podwarstwy przechodząc w ten sposób z systemów trójwarstwowych na wielowarstwowe.

Jedną z bardziej zaawansowanych rozwiązań stosowanych w tej warstwie jest technologia EJB (ang. *Enterprise JavaBeans*) [1] stanowiąca model komponentowy dedykowany dla warstwy

serwerowej aplikacji. W modelu tym logika aplikacji podzielona jest pomiędzy komponenty języka Java umieszczone w *serwerze aplikacji*. Serwer aplikacji tworzy infrastrukturę zapewniającą tym komponentom szereg usług takich jak np. bezpieczeństwo i transakcyjność oraz dba o skalowalność, niezawodność, równoważenie obciążenia lub też zoptymalizowany dostęp do warstwy przechowywania danych. Polityka bezpieczeństwa, sposób zapewnienia niezawodności czy konfiguracja transakcyjnych parametrów aplikacji jest definiowana w sposób opisowy przy wykorzystaniu języka XML. W ten sposób można łatwo dostosować aplikację do zmieniających się wymagań systemów.

Komponenty EJB udostępniają swoją funkcjonalność warstwie klienta poprzez wywołania *Java RMI*. Taki sposób komunikacji jest akceptowalny dla samodzielnych aplikacji klienckich. Aby można było jako klienta użyć przeglądarki WWW należy natomiast oprogramować dodatkowo stronę serwera WWW, który będzie wysyłał odpowiednie informacje do przeglądarki. Technologią, która jest do tego celu wykorzystywana w architekturze EJB są *serwlety* [2] i/lub *JSP* [3]. Serwlety to programy napisane w języku Java, których zadaniem jest rozszerzenie funkcjonalności serwera WWW. Serwer WWW otrzymując żądanie od przeglądarki może przekazać je do serwleta, który przetwarza żądanie (na przykład komunikując się z komponentami EJB) i generuje odpowiedź w formie dokumentu HTML zwracaną przez serwer WWW do przeglądarki. Do generacji odpowiedzi może być użyta technologia *JSP*, która pozwala na dynamiczne generowanie stron WWW. Systemy dostosowane do różnorodnych urządzeń końcowych stosują do zapisu stron język XML, który w przeciwieństwie do języka HTML opisującego sposób prezentacji dokumentu pozwala definiować strukturę dokumentu nadając poszczególnym jego elementom odpowiednie znaczenie. Sposób prezentacji dokumentu XML na urządzeniu końcowym jest definiowany przez język *XSL*.

Wymienione technologie warstwy przetwarzania danych nie muszą występować w każdym systemie jednocześnie. Istnieją systemy, w których zastosowanie np. serwletów pozwala na zrealizowanie całej wymaganej dla tego systemu funkcjonalności. Ostatnia warstwa, czyli przechowywania danych, jest odpowiedzialna za przechowywanie i udostępnianie danych. W warstwie tej zwykle stosuje się relacyjne lub obiektowe bazy danych.

4 PRZYKŁADOWE IMPLEMENTACJE

W oparciu o technologie opisane w poprzednim punkcie zaprojektowano i zaimplementowano aplikacje telemedyczne.

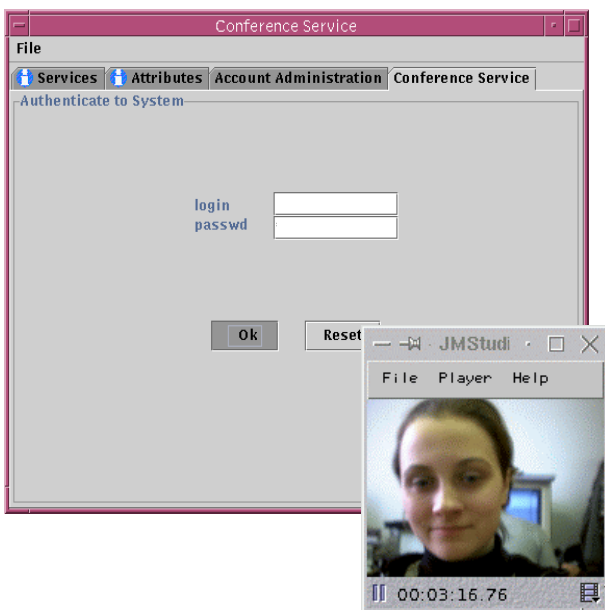
System telekonsultacyjny

Celem systemu telekonsultacyjnego [8] jest udostępnienie użytkownikom usług umożliwiających przeprowadzanie i zarządzanie wideokonferencjami, które mogą być wykorzystane do wymiany opinii pomiędzy lekarzami znajdującymi się w różnych miejscach.

Oprócz funkcjonalności związanej z obsługą strumieni multimedialnych, usługa telekonsultacyjna wprowadza dodatkowo mechanizmy ułatwiające zarządzanie wideokonferencjami. Z punktu widzenia osoby zarządzającej systemem ważna jest możliwość np. zbierania statystyk obecności użytkowników czy generowanego przez nich ruchu. W systemie jest również kontrolowany dostęp do poszczególnych informacji, oraz istnieją mechanizmy związane z autentykacją użytkowników. Podstawowym elementem pracy systemu telekonsultacyjnego jest poprawne obsłużenie zdarzeń generowanych przez składowe systemu. Typowe zdarzenia to np. pojawienie się nowego lekarza, rozpoczęcie nowej sesji konferencyjnej, negocjacja parametrów transmisji. Zdarzenia te są dystrybuowane do odpowiednich (zainteresowanych) odbiorców z wykorzystaniem *Event Service*. W ten sposób wszyscy użytkownicy na bieżąco są informowani o zmianach w systemie. Zdarzenia te są równocześnie przetwarzane przez *Log Service*, dzięki czemu możliwa będzie ich późniejsza analiza i przetwarzanie w celu wygenerowania odpowiednich statystyk.

Systemy telekonsultacyjne posiadają jednak duże wymagania sprzętowe, szczególnie jeśli pożądana jest bardzo dobra jakość przesyłanego obrazu i dźwięku. Odpowiednio szybki procesor oraz duża ilość pamięci operacyjnej to tylko niektóre z tych wymagań koniecznych przy transmisjach dwukierunkowych, w których każda ze stron w odpowiednim krótkim czasie powinna nadążać z kodowaniem wysyłanego obrazu i dźwięku oraz dekodowaniem odbieranego strumienia. Przeciętny komputer (500MHz) jest w stanie uzyskać obraz do 10klatek/s przy rozdzielczości 320x240.

Korzystanie z aplikacji telekonsultacyjnej nie sprawia wiele trudności, gdyż aplikacja ma zdefiniowane parametry domyślne (kodowanie obrazu i dźwięku, ilość klatek/sekundę, rozdzielczość, itp), wystarczające do przeprowadzenia telekonsultacji o akceptowalnej jakości. Co więcej nawet wybór adresów docelowych może być automatyczny.



Rysunek 3 System telekonsultacyjny

Dostęp do radiologicznej bazy danych

Przechowywane w bazie danych wyniki badań radiologicznych mogą być udostępniane w różny sposób. Stacje diagnostyczne o dużej rozdzielczości są zazwyczaj wyposażone w oprogramowanie pozwalające na ich zaawansowane przetwarzanie. Dostęp do tych samych danych na urządzeniu przenośnym dysponującym niewielkim jedynie wyświetlaczem musi być jednak realizowany w inny sposób. Decyduje o tym wiele czynników, wśród których należy wymienić przede wszystkim niewielką rozdzielczość wyświetlacza, mniejszą moc obliczeniową i przepustowość sieci bezprzewodowej.



Rysunek 4 Dostęp do radiologicznej bazy danych z PDA.

Zaprojektowane i zaimplementowane oprogramowanie pozwala na dostęp do radiologicznej bazy danych szpitala Jana Pawła II w Krakowie. Wyniki badań pacjentów zapamiętywane są w formacie DICOM. Jako język programowania wykorzystano język Java, którego przenośność pozwala na uruchomienie aplikacji na wszystkich urządzeniach wyposażonych w jego maszynę wirtualną. Główną

platformą sprzętową uwzględnianą w czasie projektowania aplikacji były urządzenia klasy PDA. Znalazło to oczywiste odbicie w konstrukcji interfejsu graficznego i zaimplementowanych funkcjach, dopasowując aplikację do rozmiaru wyświetlacza i sposobu interakcji z urządzeniem (minimalizując konieczność wpisywania znaków z klawiatury na rzecz użycia wskaźnika graficznego). Także od strony funkcjonalnej aplikacja została dopasowana do zastosowania celowo implementując jedynie najniezbędniejsze funkcje, co oczywiście ogranicza jej użyteczność w celach diagnostycznych. Celem aplikacji było jednak dostarczenie lekarzom narzędzia pozwalającego w dowolnym miejscu (na przykład w czasie wizyty domowej u chorego) przejrzeć wyniki badań pacjenta. Specjalista wyposażony w taki sprzęt przebywający poza zasięgiem stacjonarnej sieci komputerowej może w nagłych wypadkach po oglądnięciu przesłanych mu danych służyć radą innym lekarzom.

Komunikacja z serwerem odbywa się w oparciu jedynie o standardowy port protokołu HTTP, używany zarówno do przesyłania zdjęć radiologicznych jak i pomocniczej informacji zawartej w bazie danych. Jako że aplikacja realizuje dostęp do bazy danych, do komunikacji z serwerem używany jest język SQL, którego zapytania umieszczone są w żądaniu protokołu HTTP i kierowane do pełniacego rolę pośrednika *serwleta* komunikującego się w jej imieniu z bazą danych i zwracającego odpowiedź w postaci obiektu języka Java przesyłanego z powrotem także przy użyciu protokołu HTTP.

Rejestracja na badania

System rejestracji na badania pozwala pacjentom na zarejestrowanie się na określoną godzinę do wybranego lekarza korzystając ze zdalnego dostępu do bazy zawierającej informację o badaniach i lekarzach, którzy mają uprawnienia do ich wykonywania. Korzystając z systemu pacjent może zarządzać informacjami o rejestracjach, zmieniać ich warunki i odwoływać je. Osobą współpracującą ze strony szpitala z pacjentami jest rejestrator, który koordynuje żądania pacjentów i możliwość ich realizacji przez szpital.

System jest dostępny przez przeglądarkę WWW, ponieważ właściwości języka HTML w zupełności wystarczają do udostępnienia pełnej funkcjonalności. Jego interfejs graficzny został dopasowany do urządzeń charakteryzujących się różnymi właściwościami wyświetlaczy (laptopy, PDA). Wewnętrzna architektura systemu oparta jest na technologii EJB w połączeniu z *serwletami* i JSP. Taka budowa sprawia, że system jest skalowalny i otwarty na nowe metody dostępu. Istnieje również

możliwość skorzystania z systemu poprzez telefony komórkowe wyposażone w obsługę protokołu WAP, jednak ze względu na ograniczone możliwości tych urządzeń system udostępnia tą drogą tylko funkcje informacyjne.



Rysunek 5 System rejestracji CAS

Użytkownicy korzystający z systemu są uwierzytelniani za pomocą nazw i haseł zawartych w bazie danych a następnie autoryzowani przed wykonaniem każdej operacji, zaś ich sesja jest szyfrowana.

5 PODSUMOWANIE

Budowa nowoczesnych systemów telemedycznych wymaga użycia odpowiednich środowisk programowych w celu zapewnienia możliwości ich adaptacji do przyszłych potrzeb, skalowalności i dostosowania do różnych terminali dostępowych. Obecnie obserwuje się poważne opóźnienia we wdrażaniu nowych technologii, a większość systemów pracujących w ośrodkach medycznych w Polsce nie spełnia wymagań współczesnych architektur rozproszonych.

Podejście zaprezentowane w niniejszym artykule wskazuje na spektrum rozwiązań, które zdaniem autorów powinny być używane i wykazuje, że z powodzeniem można je zastosować w budowie nowoczesnych systemów telemedycznych.

6 BIBLIOGRAFIA

- [1] E. Roman, "Mastering Enterprise JavaBeans, 2nd Edition", John Wiley & Sons, 2001
- [2] H. Bergsten, "Java Server Pages", O'Reilly, 2000
- [3] J. Hunter, "Java Servlet Programming, 2nd Edition", O'Reilly, 2001
- [4] J. Miller, K. Zorgdrager, "What is Jiro Technology? Series I, Part I", Sun Microsystems
- [5] W. K. Edwards, "Core Jini", The Sun Microsystems Press, 1999

[6] Sun Microsystems, "Jini(TM) Architectural Overview", Technical White Paper, Sun Microsystems

[7] Sun Microsystems, "Why Jini Technology Now?" 1999

[8] J. Chlebek, "Jiro based Multimedia Conference System design and implementation", praca magisterska, 2001

[9] <http://www.okilab.com/jini/finder>

[10] Java Media Framework API 2.0 API Guide

[11] "Digital Imaging and Communications in Medicine (DICOM)", National Electrical Manufacturers Association, 2000