

**Wiosenna Szkoła PTI**  
**Świnoujście'96**

**Szybkie wytwarzanie aplikacji**  
**(Rapid Application Development)**  
*Organizacja, metody, narzędzia*

Organizowana przez  
POLSKIE TOWARZYSTWO INFORMATYCZNE

*Świnoujście 15-17 maja 1996*



**Wiosenna Szkoła PTI  
Świnoujście'96**

**Szybkie wytwarzanie aplikacji  
(Rapid Application Development)  
*Organizacja, metody, narzędzia***

Organizowana przez

**POLSKIE TOWARZYSTWO INFORMATYCZNE**

*Świnoujście 13-17 maja 1996*



## SPIS TREŚCI

**Zdzisław Szyjewski - Polskie Towarzystwo Informatyczne**

Szybkie wytwarzanie aplikacji (RAD) metoda wytwarzania  
oprogramowania

**Witold Staniszkis - RODAN-SYSTEM**

System zarządzania przepływem pracy, jako narzędzie tworzenia  
rozproszonych aplikacji

**Jerzy Sitarz, Jarosław Kaczorowski - Computer Systems For  
Business International**

Szybkie tworzenie aplikacji w wersji 8 Progressa

**Maciej Kwiatkowski - SYSTEM 2000**

Narzędzia firmy SAPIENS International jako środek realizacji metodologii  
RAD

**Grzegorz Prochowski - POLNET Technologies International**

UNIFACE Six - sterowane modelem środowisko do szybkiego  
tworzenia aplikacji klient/serwer drugiej generacji

**Grzegorz Wojarnik, Mariusz Gibultowski - Uniwersytet Szczeciński**

DELPHI 2.0 - narzędzie do graficznego tworzenia aplikacji

**Krzysztof Chelpiński - IBM POLSKA**

IBM Polska poszukuje partnerów do realizacji aplikacji

Wkładki reklamowe sponsorów i informacje



## Szybkie wytwarzanie aplikacji (RAD) metoda wytwarzania oprogramowania

Zdzisław Szyjewski  
Uniwersytet Szczeciński  
71-101 Szczecin ul. Mickiewicza 64  
tel. (91) 878080, e-mail zszyjew@uoo.univ.szczecin.pl

Czas jaki jest potrzebny na wytworzenie aplikacji informatycznej jest kluczowym czynnikiem sukcesu stosowania informatyki. Zadania stawiane przed zespołem realizującym systemy informatyczne najczęściej definiowane są niezbyt precyzyjnie ale warunek szybkiej realizacji rozwiązania jest najczęściej stawianym przez użytkownika. Dla sprostania tym oczekiwaniom, zespoły informatyczne powinny posiadać odpowiednie narzędzia wytwarzania, metody pracy oraz organizować pracę zespołu, tak aby bez szkody dla jakości wytworzonego produktu spełnić wymagania klienta. Kryterium szybkiej realizacji połączone jest równocześnie z wymaganiami wysokiej jakości produktu.

Na licznych konferencjach naukowych prezentowane są nowe metody i sposoby wytwarzania systemów informatycznych, które pozwalają wytwarzać więcej, lepiej i szybciej. Największe firmy komputerowe bardzo często prezentują nowe produkty, które również pozwalają więcej, lepiej i szybciej wytwarzać systemy informatyczne. Prezentowane metody, oferowane produkty, określane są chwytliwymi marketingowo, zmieniającymi się w zależności od mody przymiotnikami. Modnym ostatnio przymiotnikiem jest *obiektowy*. Wszystkie te techniki, metody, narzędzia lub organizacja pracy, stanowią nowe ramowe sposoby wytwarzania wykorzystujące postęp technologiczny

Metody generowania oprogramowania, programy szkieletowe, prototypowanie, języki 4GL czy wreszcie narzędzia programowania obiektowego, to kolejne etapy rozwoju oprogramowania i środowiska pracy programisty. Działania te mają na celu sprostanie oczekiwaniom użytkownika i zagwarantowanie wysokiej jakości wytwarzanych systemów informatycznych. *Szybkie wytwarzanie aplikacji (Rapid Application Development)* to metodyka wytwarzania systemu, która łączy nowoczesne narzędzia, sprawną organizację procesu wytwarzania i włącza aktywnie użytkownika w proces wytwarzania systemu

Ukierunkowanie prac pod kątem spełnienia oczekiwań przyszłego użytkownika i dbanie o jakość wytwarzanego produktu to zasadnicze zmiany w akcentach stosowanej metodyki wytwarzania. Metodyka RAD charakteryzuje się:

- gwarantowaną wysoką jakością rozwiązania,
- ewolucyjnością procesu wytwarzania systemu,
- stosowaniem inżynierskich technik wytwarzania w pełnym cyklu realizacji,
- wytwarzaniem przez profesjonalne zespoły wykonawcze, dedykowane do określonego zadania,
- używaniem profesjonalnych metod zarządzania zespołem,
- stosowaniem silnych narzędzi wspomagających wytwarzanie oprogramowania,
- gwarantowaną produktywność wytwarzanego systemu

Stosowanie metodyki RAD pozwala na wytwarzania dużych i skomplikowanych systemów informatycznych w sensownym czasie i zgodnie z wymaganiami użytkownika. Osiągnięcie założonych celów często jest związane z koniecznością integracji nowego rozwiązania z wcześniej wytworzonymi i eksploatowanymi systemami informatycznymi. Integracja taka jest trudna do osiągnięcia z wielu bardzo istotnych powodów. Stare rozwiązania metodyczne wykorzystujące nienowoczesne technologie i sprzęt komputerowy muszą współdziałać z najnowszymi osiągnięciami informatyki. Każdy informatyk wie jakie problemy powstają w takich sytuacjach i jak trudno zintegrować dwa rozwiązania z różnych epok nie burząc dotychczasowych, eksploatowanych systemów. Ewolucyjne przeprowadzenie użytkownika w nowoczesną technologię informatyczną to nie tylko problem inżynierski, to wiele złożonych i skomplikowanych działań koniecznych do wspólnego z użytkownikiem wykonania.

Bardzo często produkty wytwarzane z wykorzystaniem metodyki RAD mają z założenia czasowy termin użytkowania, czyli są to rozwiązania prowizoryczne pozwalające na bezbolesne przeprowadzenie użytkownika w nowoczesną technologię informatyczną. Zachowanie dotychczasowego dorobku, przyzwyczajzeń, metod pracy czy wreszcie danych na nośnikach komputerowych, to często definiowane uwarunkowania przy wprowadzaniu nowych aplikacji informatycznych. Mimo prowizorycznego charakteru wytwarzanego systemu nie można pozwolić sobie na obniżenie wymagań jakościowych. Zorientowanie na jakość produktu to jeden z podstawowych wymogów metodyki.

Cykl życia systemu wyznacza kolejne kroki realizacji prac zmierzających do wytworzenia systemu informatycznego. Najczęściej stosowaną metodyką jest kaskadowy model realizacji systemu. Kolejno realizowane czynności gwarantują wytworzenie systemu informatycznego zgodnie z założeniami zdefiniowanymi w początkowym etapie prac. Metodyka RAD z założenia, przewiduje cykliczne powtarzanie kolejnych faz realizacji systemu. Ewolucyjny charakter realizacji procesu wytwarzania zakłada równocześnie bieżące opiniowanie stosowanych rozwiązań przez użytkownika i wprowadzanie zmian i modyfikacji zgodnie z życzeniami klienta.

Ewolucyjny sposób realizacji stwarza problemy zarządzania procesem wytwarzania, z utrzymaniem odpowiednich wymagań jakościowych w kolejnych iteracjach cyklu produkcyjnego. Podstawowym założeniem metodyki jest ewolucyjne dopasowywanie się systemu informatycznego do zmieniających się wymagań środowiska eksploatacji systemu.. System informatyczny musi być na bieżąco dopasowywany do zmieniających się wymagań biznesowych. Dotyczy to nie tylko części informatycznej ale również działalności podstawowej wspomaganej systemem informatycznym.

Adaptowanie procesu wykonania do zmieniającej się na bieżąco aplikacji to wyzwanie zmuszające zespół informatyczny do stałego doskonalenia warsztatu pracy i stosowanych metod, narzędzi i organizacji pracy. Szybsze dostarczanie użytkownikowi rozwiązania informatycznego pozwala na ewolucyjne dokonywanie zmian w technologii pracy ale zmusza do stałego kontaktu i natychmiastowego reagowania na zgłaszane uwagi i sugestie. Taki sposób realizacji systemu informatycznego w sposób znaczący zmniejszenie ryzyka popełnienia błędów. Użytkownik systemu informatycznego staje się współtwórcą rozwiązania przez co identyfikuje się z nim, co jest istotnym warunkiem ostatecznego sukcesu wdrożeniowego.

Zastosowania informatyki wtapiane są w konkretne, zmieniające się środowisko. Utrzymanie systemu informatycznego w stałej zgodności ze zmieniającymi się warunkami otoczenia i technologii informatycznych jest przedsięwzięciem bardzo trudnym. Zmiany nie dokonują się



skokowo ale ewolucyjnie i wspomagający pracę system informatyczny też powinien tak być modyfikowany aby nadążać za tymi zmianami. Technologia informatyczna stwarza nowe możliwości, które zwiększają produktywność obiektu stosowania informatyki ale to wymaga rozszerzenia zakresu rozwiązania informatycznego. Pomoc w przyrostowym rozwoju systemu to kolejne wyzwanie dla zespołu informatycznego.

Sprostanie tym wyzwaniom jest przesłanką rozwoju metodyk wytwarzania systemów informatycznych. Metodyka RAD jest próbą zaspokojenia oczekiwań. Oferowane metody wytwarzania, narzędzia wspomagające i organizacja procesu wytwarzania stanowią spójne środowisko pozwalające na szybkie wytwarzanie aplikacji spełniających wysokie wymagania jakościowe stawiane przez użytkowników.

Dostarczone materiały konferencyjne wraz z prezentacjami w trakcie konferencji przybliżą niektóre z oferowanych obecnie rozwiązań i pozwolą na wyrobienie sobie opinii o aktualnym stanie metodyki RAD.



# Charakterystyka architektury Systemu Zarządzania Dokumentami i Przepływem Pracy.

Witold Staniszkis

RODAN SYSTEM Sp. z o.o.

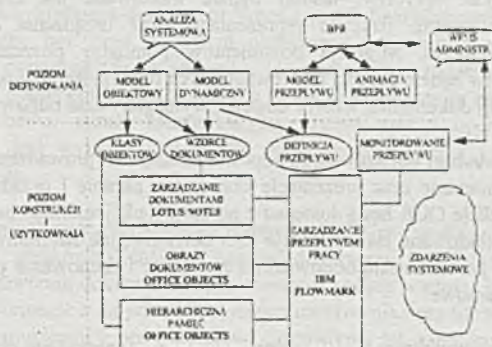
Jagielska 50c. 02-886 Warszawa

tel. 643 92 08 fax: 643 92 10

E-mail: rodmail@rodan.waw.pl http://www.rodan.waw.pl

Architektura Systemu Zarządzania Dokumentami i Przepływem Pracy /SZDiPP/ zaprezentowana jest w terminologii zasadniczych bloków funkcjonalnych systemu i powiązań między nimi. Prezentacja schematyczna głównych obszarów funkcjonalnych pokazana jest na rys. 1. Każdy ze składników funkcjonalnych systemu odzwierciany jest na odpowiedni element architektury klient/serwer proponowanej dla systemu.

## Architektura Systemu



Rys.1 Architektura funkcjonalna Systemu Zarządzania Dokumentami i Przepływem Pracy

Zgodnie z naszą metodologią faza analizy i projektowania koncepcyjnego projektu informatycznego zawiera definicje pojęć związanych z systemem komputerowym (np.: klasy obiektów, formaty dokumentów, formaty ekranów itp.) oraz specyfikacje pojęć związanych z przedsięwzięciem (np.: czynności użytkownika, procedury, zasady). Ten drugi zbiór czynności projektowych zawiera zasadnicze kroki obecne we współczesnych metodologiach przebudowy procesów funkcjonowania przedsiębiorstw. Notacja Workflow Graph rozszerzająca powszechnie używaną notację przepływu danych (DFD) będzie używana do specyfikowania wymagań procesów funkcjonowania przedsiębiorstw, które mają być wspierane bądź utworzone przez proponowany system informatyczny.

**Funkcje zarządzania przepływem pracy** dostarczają zasadniczej warstwy kontrolnej dla całego systemu; odpowiadają za zapoczątkowanie i zakończenie czynności, zarówno automatycznych, jak i sterowanych ręcznie, w związku z odpowiednimi zdarzeniami w systemie informatycznym. Oznacza to, że w większości przypadków funkcje innych obszarów funkcjonalnych będą uruchamiane z poziomu funkcji zarządzania przepływem pracy. Zatem procesy zarządzania funkcjonowaniem przedsiębiorstwa zawierające funkcje z różnych obszarów funkcjonalnych np.: imaging'u dokumentów, zarządzania dokumentami itd., będą wykonywane zgodnie z grafem przepływu pracy jako ciąg zautomatyzowanych czynności. Dlatego dynamika systemu zdefiniowana zostanie na dwóch oddzielnych poziomach: na poziomie definiowania grafu przepływu pracy reprezentującym globalne zachowanie systemu oraz na poziomie klas obiektów i formatów dokumentów reprezentującym lokalną charakterystykę zachowań poszczególnych elementów systemu.

**Funkcje zarządzania dokumentami** dostarczą wszystkie wymagane własności, takie jak klasyfikacja, składowanie i zapamiętywanie dokumentów, kontrola dostępu oraz prezentacja i transmisja dokumentów. Zakłada się niewidoczną integrację z funkcjami archiwizacji dokumentów w cel umożliwienia zintegrowanego podejścia do zarządzania składowaniem dokumentów. Hierarchia nośników dokumentów (pamięć operacyjna, dysk magnetyczny, dysk optyczny, taśma) będzie traktowana jak zintegrowany magazyn dokumentów, w którym fizyczna reprezentacja oraz urządzenia są transparentne dla użytkowników systemu. Migracja dokumentów pomiędzy poszczególnymi poziomami hierarchii nośników będzie zależna od wewnętrznych reguł systemu, takich jak np.: predykaty ważności czasowej dokumentu, wzorce dostępu, wymagany czas odpowiedzi itp.

**Funkcje obsługi obrazów** będą odpowiedzialne za wprowadzenie obrazu dokumentu, kompresję, zapamiętanie oraz prezentację obrazu (na ekranie i urządzeniach drukujących). Odpowiednie funkcje OCR będą dostępne z obszaru funkcjonalnego imaging'u dokumentów. Dokumenty są składowane jako instancje klas obiektów, jak zdefiniowano podczas procesu analizy systemu, przedstawiają odpowiednią zawartość i zachowanie podczas dostępu przez inne funkcje systemowe.

Powyższa architektura funkcjonalna SZDiPP proponowanego dla Użytkownika będzie zrealizowana przy użyciu następujących technologii informatycznych i związanych z nimi narzędzi:

- praca grupowa i zarządzanie dokumentami - LOTUS Notes wersja 4.0 jest proponowany jako podstawowa platforma software'owa
- obrazowanie dokumentów i archiwizacja - Office Objects wersja 1.0 jest proponowany jako podstawowa platforma software'owa
- zarządzanie przepływem pracy - IBM FlowMark wersja 2 jest proponowane jako podstawowa platforma software'owa.

Analiza systemu, przebudowa organizacji, projektowanie i proces implementacji będą prowadzone zgodnie z wytycznymi i zasadami "Metodologii projektowania systemów RODAN OO", bazującej na podejściu zorientowanym obiektowo i technice Object Management Technique (OMT). Narzędzie CASE, które będzie używane przez zespół projektowy to ObjectTeam firmy Cadre Technologies.



## Charakterystyka techniczna LOTUS Notes wersja 4.0

### Co to jest LOTUS NOTES?

Lotus Notes zdefiniował nowy rodzaj oprogramowania zwany groupware, który umożliwia przedsiębiorstwom wykorzystanie pełnego potencjału, zarówno ich pracowników, jak i sieci. Zespoły mogą współpracować w sposób szybszy, bardziej zorganizowany, produktywny i osiągnąć więcej, używając mniejszej ilości zasobów.

Lotus Notes jest środowiskiem klient/serwer, który umożliwia użytkownikom (lub klientom) na bezpieczne komunikowanie się poprzez sieci lokalne albo łącza telekomunikacyjne z dokumentami, które rezydują na dzielnym komputerze (lub serwerze). Notes łączy środowisko do rozwoju oprogramowania, bazę danych dokumentów, wyrafinowany system pocztowy, dając możliwość kreowania specjalistycznego oprogramowania dla poprawienia jakości procesów biznesowych w takich obszarach, jak tworzenie nowych produktów, obsługa klientów, sprzedaż, czy zarządzanie współpracą z klientami.

### Opis LOTUS NOTES

#### Bazy danych dokumentów

Lotus Notes jest systemem bazy danych dokumentów - podstawowym elementem bazy danych jest dokument. Struktura dokumentu Notes'a jest definiowana przez formularz, który zawiera pewną liczbę pól. Np.: dokument określający procedury i politykę względem "czegoś" może być opisywany przez pola takie jak: data, nazwa rodzaju polityki, założenia polityki, pełny tekst dokumentu, zaś dokument o obsłudze klienta może zawierać takie rzeczy jak data, nazwa klienta, numer identyfikacyjny klienta, nazwa operatora, pole tekstu swobodnego, który jest wykorzystywany dla opisu pytania klienta i pole statusu tego konkretnego zapytania.

Użytkownicy mają możliwość przeglądania zestawień Notes'ów według nazw pól, w formie kolumn. Np.: jeżeli użytkownik chciałby oglądać dokumenty posegregowane według dat, Notes pokazuje zestawienie dokumentów posortowaną według wartości wprowadzonych do pola "data". Inne informacje z innych pól np. nazwa użytkownika czy nazwa rodzaju polityki, umieszczone są w kolumnach z prawej strony. Zestawienia tak zaprezentowane są bardzo elastyczne, co oznacza, że jeżeli dokument źródłowy ma wiele dokumentów pochodnych, użytkownicy mogą sobie zażyczyć prezentacji tylko dokumentu źródłowego lub dokumentu źródłowego i wszystkich dokumentów bezpośrednio po nim następujących, bądź też prezentacji wszystkich dokumentów związanych z dokumentem początkowym.

Podstawowym elementem bazy danych Notes, jak już wspomniano, jest dokument, który może znajdować się w dowolnym miejscu w układzie strukturalnym całego zbioru dokumentów. Notes może więc przechowywać i zarządzać zbiorami dokumentów, które w nieoczywisty sposób można sklasyfikować w relacyjnym lub innym systemie bazy danych. Co więcej, ponieważ system używa modelu bazy danych dokumentów, Notes może dostarczyć użytkownikowi wielu użytecznych narzędzi, jak na przykład:

- **Bogate narzędzia tekstowe i multimedialne.** Magazyn obiektów Notes'a jest typem kontenera zoptymalizowanego w celu dystrybucji i zarządzania informacjami biznesowymi. Te informacje często zakładają różnorodność typów danych, takich jak dane tabelaryczne, tekst sformatowany, grafika, przyłączone i wklejone obiekty, podobnie, jak obiekty multimedialne, takie jak skanowane obrazy, faxy, głos/dźwięk i video.
- **Wyszukiwanie pełnotekstowe.** Lotus Notes zawiera mechanizm wyszukiwania pełnotekstowego, który pozwala użytkownikom na indeksowanie dokumentów na podstawie pytań użytkownika. Notes wyświetla te dokumenty, które pasują do kryteriów przeszukiwania, w sposób posortowany według relewancji albo w sposób wyspecyfikowany przez użytkownika.
- **Kontrola wersji.** Lotus Notes dostarcza możliwości kontrolowania wersji dokumentów w ten sposób, aby możliwe było śledzenie różnorodnych zmian, dokonywanych na pojedynczym dokumencie przez wielu użytkowników. Automatyczna kontrola wersji jest narzucona w takiej postaci, w której każda zmiana jest dokonywana jako dokument podstawowy, albo jako odpowiedź na dokument podstawowy. W ten sposób zmiany wprowadzone do dokumentu podstawowego przez jednego użytkownika nie są zamazywane, podczas, gdy następny użytkownik, chce do niego wprowadzić inne zmiany. Kontrola wersji dokumentów jest na tyle elastyczna, że może sprostać zapotrzebowaniu dowolnego zespołu roboczego. Kiedy automatyczna kontrola wersji nie jest narzucona, Notes przez cały czas oznacza kolidujące zmiany specjalnym wskaźnikiem. Użytkownicy mogą również umieścić dodatkowe komentarze na oryginalnym dokumencie bez konieczności ponownego zapamiętywania dokumentu oryginalnego.

### Replikacja

Zasadniczą cechą platformy groupware'owej jest możliwość dzielenia informacji między wszystkimi członkami grupy roboczej niezależnie od czasu czy miejsca, w którym się znajdują. Grupy często mają członków, którzy rozlokowani są w różnych oddziałach firmy, często po całym kraju lub po całym świecie. Każde z tych miejsc zawiera swój własny serwer, z którym lokalni pracownicy mogą się w sposób mało kosztowny połączyć.

Platforma groupware'owa używa technologii replikacji, aby umożliwić oddalonym oddziałom, np. w Kansas City, wykonanie kopii bazy danych np. z Atlanty i składowanie jej na lokalnym serwerze. Zamiast wymagać od użytkowników z Kansas City, żeby połączyli się z serwerem bazy danych w Atlancie, dzięki replikacji, grupy robocze mogą używać informacji zsynchronizowanej pomiędzy poszczególnymi, oddalonymi od siebie oddziałami. Aplikacje groupware'owe nie wymagają integracji, bądź też dwufazowej realizacji transakcji wykonywanej przez system monitorowania transakcji. Systemy te mają raczej specjalne wymagania dotyczące replikacji. Replikacja w Notes jest unikalna pod względem jej rozdzielczości i funkcjonalności. Zasadniczymi cechami replikacji w Notes są:

- **Dwukierunkowość.** Zmiany w dokumentach w bazie danych następują równocześnie we wszystkich miejscach, które używają tej samej bazy danych albo nie następują nigdzie. Notes przyjmuje wszystkie zmiany, nie tylko te, które są przeprowadzone w centralnym miejscu, aby je potem przekazać do serwerów oddalonych.



- **Wydajność.** Podczas synchronizowania baz danych, replikowane są tylko te dokumenty, które są nowe albo te, które zostały zmienione w dowolnym miejscu. Notes rozpoznaje odpowiednie dokumenty, używając minimalnej ilości zasobów, potrzebnej do przeprowadzenia operacji.
- **Replikacja klienta.** Zdarza się, że członkowie grup tacy jak użytkownicy ruchomi, którzy pracują w miejscu oddalonym lub w domu, potrzebują tego samego poziomu dostępu do bazy danych, który jest właściwy dla użytkowników połączonych bezpośrednio. Notes nie jest ograniczony przez połączenie serwer-serwer, ale również umożliwia replikację klienta do serwera.
- **Replikacja selektywna.** Notes pozwala użytkownikom na definiowanie na swoich stacjach roboczych profili dokumentów, które muszą być replikowane. Używając selektywnej replikacji, użytkownicy mogą replikować tylko te dokumenty, które zostały zmienione np. w ciągu ostatnich 30 dni albo te, które pochodzą od pewnego szczególnego członka grupy roboczej.
- **Replikacja w tle.** Dla użytkowników ruchomych przeprowadzanie procesu replikacji nie powinno oznaczać tego, że ich inna działalność na laptopie czy komputerze domowym musi być zawieszona. Replikacja Notes'owa może być przeprowadzona w tle, co pozwala użytkownikom na kontynuowanie ich pracy.

### Bezpieczeństwo.

Przedsiębiorstwa na świecie zaczęły budować aplikacje groupware'owe, które rozciągają się nie tylko wewnątrz przedsiębiorstwa, ale też do jego dostawców i klientów. Mimo tego, że informacja zazwyczaj bywa strzeżona przez poszczególnych użytkowników, (departamenty, firmy), silna konkurencja na rynku wymaga przemian organizacyjnych, przekraczania tradycyjnych barier wewnątrz i pomiędzy przedsiębiorstwami w celu, mimo wszystko, dzielenia się informacjami. Wiele informacji dzielonych w aplikacjach groupware'owych jest poufnych, chronionych przed przypadkowym lub celowym, nieuprzywilejowanym wykorzystaniem. Dlatego też platforma groupware'owa wymaga takiego poziomu bezpieczeństwa, który jest z jednej strony wystarczająco rygorystyczny, aby ochronić ważną informację przed sabotażem, a z drugiej strony - elastyczny na tyle, aby upoważnieni użytkownicy mogli określać różnego rodzaju poziomy dostęp do dokumentów.

Kluczem do bezpieczeństwa w systemach rozproszonych jest kodowanie. De facto standard przemysłowy dla dostępu do katalogu X.500 nazywa się X.509 i bazuje na technologii kodowania kluczem publicznym RSA, uznawanym za jedyny algorytm kodowania nie posiadający jawnego punktu słabego. W rzeczywistości technologia kodowania RSA jest tak silna, że jest chroniona przez NSA przed niekontrolowanym eksportem i tylko wersje zmodyfikowane mogą być eksportowane z USA. Przy użyciu kryptografii i innych metod zabezpieczeń, Lotus Notes oferuje 4 poziomy bezpieczeństwa:

- autentykację,
- kontrolę dostępu,
- prywatność na poziomie pola,
- podpisy cyfrowe.

## Charakterystyka techniczna Office Objects wersja 1.0

### Opis funkcjonalny

**Office Objects** jest obiektowo zorientowaną biblioteką funkcji zarządzania dokumentami klasy klient/serwer, która zawiera interfejsy do procesów imagingu, standardowych aplikacji biurowych oraz zawiera mechanizmy do indeksowania, składowania i odzyskiwania dokumentów.

Pakiet umożliwia zintegrowanie wielu technologii, które uczestniczą w organizowaniu codziennych procesów w jednym środowisku, niezależnie od platform sprzętowych. Obiektowo zorientowana natura systemu przybliży zaimplementowaną funkcjonalność problemu świata rzeczywistego.

System jest chroniony i administrowany przez definiowanie użytkowników i ich grup, zapewniając zuniformizowaną kontrolę dostępu do obiektów oraz podstawowe możliwości wewnętrznej komunikacji. Jego otwartość ułatwia integrację z innymi złożonymi aplikacjami przez zespół interfejsów proceduralnych albo przez bezpośrednie odniesienia do obiektów, w przypadku aplikacji, które na to pozwalają. Architektura systemu umożliwia dokładanie nowych technologii, kiedy stają się niezbędne lub dostępne.

Niezależnie od źródła pochodzenia dokumentu (przychodzący, wychodzący, wewnętrzny) oraz jego natury (fax, tekst), dokumenty transformowane są do postaci tekstu, który jest używany do przeprowadzenia indeksacji jego zawartości. Jest możliwe dodatkowe indeksowanie po atrybutach. Dokumenty są przechowywane w zasobnikach dokumentów, które mogą być bazami danych, bibliotekami optycznymi, w celu późniejszego odzyskiwania tych dokumentów. Aby umożliwić grupowe edytowanie dokumentów zaprojektowano procedury zgłoszenia i zaprzestania pracy na dokumencie, które umożliwiają dostęp do dokumentu. Dokumenty, podobnie jak inne obiekty systemowe, są chronione przez listę kontroli dostępu. Każdy obiekt ma określony poziom dostępu właściciela, grupy wraz z zaznaczeniem ograniczenia dostępu.

### Imaging

Imaging Office Objects umożliwia obróbkę obrazów tzn. ich skanowanie, przeglądanie, tworzenie adnotacji, drukowanie i zachowywanie. Imaging Office Objects umożliwia ponadto:

- pełne ustawianie parametrów skanowania
- skanowanie zarówno pojedynczych stron, jak i grup do jednego pliku
- skanowanie, drukowanie, przeglądanie i zachowywanie w trybie kolorowym, czarno-białym lub przy użyciu poziomów szarości
- dokonywanie adnotacji tekstowych oraz odręcznych
- dowolne ukrywanie lub uwypuklanie fragmentów obrazu
- zmianę rozmiarów przeglądanych obrazów, ich obracanie, wykorzystanie inwersji i efektu lustrzanego odbicia



- drukowanie obrazów z przekształceniem ich do rozmiarów strony lub w aktualnym rozmiarze
- drukowanie wielu obrazów na jednej stronie
- zachowywanie obrazów w wielu popularnych formatach pliku jak TIFF, GIF, PDA czy PCX, oraz zastosowanie popularnych algorytmów kompresji jak CCITT Group 3 i 4, packbits czy LZW

Imaging Office Objects może występować odrębnie lub być zintegrowany poprzez API z Lotus Notes.

Wersja 1.1 Imaging Office Object pracuje na większości komputerów osobistych, znajdujących się obecnie na rynku. Minimalne wymogi programowe oraz sprzętowe, które musi spełniać system komputerowy przedstawiają się w następujący sposób:

- Komputer PC 386SX
- 4MB pamięci RAM
- 2MB wolnego dysku twardego
- karta VGA
- mysz lub inne urządzenie wskazujące
- system operacyjny DOS 5.0
- system Windows 5.1

#### **Hierarchiczne przechowywanie dokumentów**

Hierarchiczny system obsługi przechowywania danych zapewnia optymalne wykorzystanie mediów. W miarę przyrostu ilości informacji zgromadzonej w bazie danych konieczne jest wprowadzenie optymalnego sposobu zarządzania dostępem do informacji. Ogólna zastosowana tu reguła polega na przenoszeniu przeznaczonych do archiwizacji lub rzadziej używanych informacji na tańsze (wolniejsze, lecz bardziej pojemne) nośniki np. jukebox'y (optyczne dyski) lub taśmy magnetyczne. Zgodne jest to z zasadą, że do często używanych danych dostęp musi być relatywnie szybki - do rzadziej używanych danych może być wolniejszy. Czynności archiwizacyjne są zautomatyzowane, co w dużej mierze odciąża pracę administratora systemu. Dzięki zastosowaniu techniki pre - fetching'u system odpowiednio przygotowuje dane przenosząc je w miarę potrzeby na nośniki efektywniejsze pod względem czasu dostępu. Czynność ta wykonuje się "w tle" bez wiedzy użytkownika, który w zasadzie nie odczuwa różnicy w prędkości pracy systemu.

## Charakterystyka IBM FlowMark wersja 2

### Opis funkcjonalny.

IBM FlowMark jest narzędziem zarządzania dokumentami i przebiegiem pracy, które pozwala na optymalizację procedur gospodarczych i administracyjnych. FlowMark umożliwia również projektowanie, doskonalenie, kontrolowanie i dokumentowanie tychże procesów i procedur. Tym samym możliwym staje się:

- definiowanie i dokumentowanie procedur zarządzania
- dokonywanie sprawdzeń, kontroli przebiegu czynności
- takie kierowanie procesami aby:
- wspomagać pracowników w wykonywaniu nałożonych na nich zadań
- w pełni zautomatyzować czynności, które nie wymagają ludzkiej asysty lub kierowania
- monitorować poszczególne operacje składające się na uprzednio zdefiniowany proces
- doskonalić i aktualizować (stosownie do nowej sytuacji) procedury

### Określanie i dokumentowanie procesów

FlowMark stanowi połączenie techniki komputerowej klient / serwer z dwoma podstawowymi ideami postępowania: listy zadań "to-do list" oraz metodyki realizacyjne (procedures manuals). Listy zadań "to-do lists" wprowadzane są do systemu FlowMark na bieżąco (online). Metodyki realizacyjne pracują również na zasadzie online, stając się modelami przebiegu pracy i zarządzania dokumentami, które to modele zabezpieczają uruchomienie procedur i procesów.

Budowanie modelu przebiegu pracy odbywa się poprzez jego wyrysowanie za pomocą zaawansowanego narzędzia projektowego, które jest częścią FlowMark Managera Procesu. W ten sposób ma miejsce opisanie poszczególnych czynności, zespołu / zespołów pracowniczych je wykonywujących, programów wspomagających pracowników, przebiegu kontroli oraz informacji pomiędzy czynnościami. Te informacje opisujące model są magazynowane w bazie danych "ObjectStore", która jest częścią FlowMark Managera Procesu.

Ponieważ określany model jest dokumentowany przez użytkownika, jako diagram sam w sobie oraz przez wspomagającą dokumentację online, FlowMark zawiera możliwości wydrukowania modeli przebiegu pracy i zarządzania dokumentami, w dowolnym, wymagalnym przez użytkownika momencie.

### Badanie procesów

Wraz z FlowMark-iem użytkownik otrzymuje narzędzie animacyjne, które pozwala "kroczyć" poprzez model przebiegu pracy, poddając symulacji realne warunki eksploatacyjne. Możliwe są do określenia tzw. "wąskie gardła" a tym samym uniknięcie martwych miejsc przed uruchomieniem zaprojektowanego systemu. Można także użyć animacji dla zademonstrowania pracownikom jak funkcjonują procesy. Czynności do wykonania pojawiają się na listach prac FlowMark-u, przy nazwiskach wyznaczonych do nich członków załogi. Kiedy pracownik dokonuje selekcji czynności, program wspierający pracę jest uruchamiany automatycznie, wraz z dostępem do potrzebnych informacji. Listy prac FlowMark-u dają pracownikom stale uaktualniany przegląd (obraz) ich nierozstrzygniętych, bieżących spraw i czynności. Po wdrożeniu zarządzania przebiegiem pracy / dokumentami, pracownicy będą



używać listy prac jako ich podstawowego punktu kontaktowego ze środowiskiem techniki komputerowej. Możliwy będzie dla nich łatwy i jednolity dostęp do aplikacji i danych na wielu różnorodnych platformach.

W przypadku zmian warunków działania / gospodarowania, użytkownik może zaprojektować nowe procesy i procedury lub skorygować już istniejące. Może także dodać lub usunąć czynności, zmienić kolejność, w której są one wykonywane oraz dokonać innego ich przypisania różnym ludziom.

### **Drogi uzyskiwania rezultatów**

FlowMark może generować dane kontrolne, gdy użytkownik eksploatuje zaprojektowane przez niego modele przebiegu pracy. Procesy kontrolne pomagają znaleźć i wyeliminować miejsca przysparzające kłopotów i trudności w procesach.

### **Redukcja kosztów dokumentacji**

Dokumentacja jest integralną częścią zarządzania przebiegiem pracy. Użytkownik nie musi oddzielnie projektować, wprowadzać w życie i dokumentować procesu. Stworzony model "online" jest udokumentowanym procesem. Użytkownik może także wydrukować swój model, jeżeli uzna to za potrzebne.

### **Pomoc w doprowadzeniu do zgodności z uregulowaniami rządowymi**

W Uni Europejskiej (EU), firmy ubiegające się o kontrakty z agencji publicznych muszą dokumentować swoje procesy / procedury w zgodności z International Standards Organisation (ISO) Standard 9000. W Wielkiej Brytanii oferenci muszą postępować zgodnie z British Standard 5750. W Stanach Zjednoczonych agencje federalne poddają inspekcji dokumentację procedur ofertowych co do ISO Standard 9000. FlowMark daje użytkownikowi elastyczne narzędzie, które jest mu niezbędne dla spełnienia tych wymagań.

### **Utrzymywanie procesu "w ruchu"**

Wówczas, gdy jedna czynność w procesie jest zakończona, stosowni pracownicy (na określonym poziomie odpowiedzialności) widzą natychmiast na ich listach pracy następną czynność do wykonania. Spotkania, notki, notatki służbowe i telefony nie są konieczne dla utrzymania rutynowej ciągłości pracy.

### **Uproszczenia w użyciu komputerów przez pracowników**

Listy pracy FlowMark-u nie tylko przypominają pracownikom, iż mają oni pracę do wykonania. One także automatycznie łączą użytkowników z programami lub narzędziami wspomagającymi, których akurat wymaga ich praca. Wyeliminowana jest praca związana ze znalezieniem i uruchomieniem programów.

### **Koordinacja odwzorowania i rozwiązań dot. pracy biurowej**

W obszarze odwzorowywania i rozwiązań w zakresie pracy biura, modele przebiegu pracy i zarządzania dokumentami dodają wartościową cechę poprzez określanie przepływu danych i kontroli pomiędzy programami, które przetwarzają dokumenty. Wynikiem jest większa przepustowość (zwiększony przepływ) i większa niezawodność (mniejsza ilość pomyłek) w operowaniu dużymi ilościami informacji zawartych na papierze.

### **Automatyzacja zadań**

Czynności, które zawierają jedynie uruchomienie pewnych programów mogą być wykonywane automatycznie przez FlowMark. Pracownicy mogą spożytkować zaoszczędzony czas na pracę, która rzeczywiście wymaga posiadanych przez nich umiejętności.

### **Redukcja błędów przy wprowadzaniu danych**

W typowych procesach dane i informacje jak: nazwy dokumentów, nazwiska klientów i numery transakcji używane są w wielu czynnościach w sposób powtarzający się. Powtórne wprowadzenie tych informacji jest głównym źródłem błędów w nieautomatyzowanych procesach. Przy zastosowaniu FlowMark-u dane muszą być wprowadzone tylko jeden raz i są potem przekazywane automatycznie do późniejszych operacji. Błędy są zredukowane do minimum oszczędzając czas i pieniądze instytucji.

### **Redukcja kosztów szkolenia**

Użytkownik może wbudować wskazówki i system pomocy "online" w modele przebiegu pracy. Razem z zadaniami na liście prac, pracownicy otrzymują także instrukcje co do sposobu wykonania. Tego typu pomoc może istotnie zredukować wymogi szkoleniowe.

### **Doskonalenie pracy grupowej oraz usług**

Wprowadzanie wskazówek online umożliwia zatrudnionym na bardziej efektywną interakcję między sobą oraz z klientami. Listy sprawdzeń online mogą przypominać członkom zespołów o informacjach (szczegółach), które powinny być uwzględnione w raportach. Gotowe scenariusze (opisy struktury działań lub procesu) mogą "prowadzić" pracowników w całym przebiegu transakcji z klientami. FlowMark jest zaprojektowany dla każdego z pracowników w instytucji / firmie, który użytkuje komputery. Nie są wymagane żadne specjalne umiejętności poza tymi, które zwykle związane są z pracami ludzi.

### **Administratorzy systemu**

Administratorzy systemu instalują FlowMark, tworzą bazę danych FlowMark, definiują pracowników oraz autoryzują użytkowników. Oni także zarządzają procesami kontrolnymi (danymi kontrolnymi) FlowMark-u i używają mechanizmów (import i eksport) dla modelowania przebiegu pracy. Administratorzy systemu muszą wiedzieć jak utworzyć i utrzymywać sieć lokalną (LAN) dla OS/2, AIX lub Windows oraz jak zarządzać bazą danych instytucji / firmy.



### **Specjaliści ds. modelowania**

Specjaliści ds. modelowania budują, testują i dokumentują modele zarządzania przebiegiem pracy. Dokonują oni rejestracji programów i struktur danych wymaganych przez modele. Specjaliści ci, winni posiadać umiejętności analizy działalności instytucji i analizy systemowej dla projektowania i wdrażania modeli. Muszą oni rozumieć wymagania pracowników, programy oraz struktury danych dla procesów, które modelują.

### **Programiści**

Programiści modyfikują aplikacje, które będą "przywiązane do modeli przebiegu pracy oraz rozwijają nowe aplikacje. FlowMark wspomaga te zadania programowe poprzez interfejsy programowe (API) dla języka C (OS/2, AIX i Windows), dla REXX (OS/2 i AIX) oraz dla Microsoft Visual Basic (Windows).

### **Administratorzy procesu**

Administratorzy procesu (oraz upoważnieni użytkownicy listy pracy) kontrolują przebieg procesów. Mogą oni rozpoczynać, przerywać i podejmować na nowo procesy oraz mogą oni zmieniać przydział pracy.

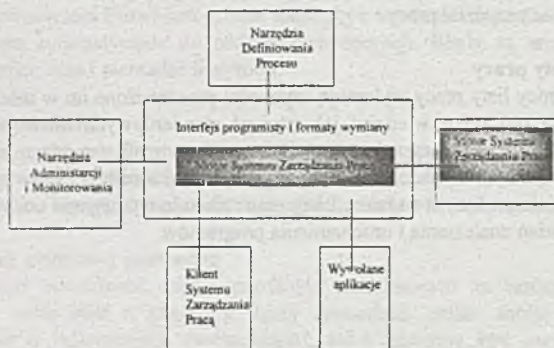
### **Użytkownicy listy pracy**

Użytkownicy listy pracy wykonują czynności przeznaczone im w modelach przebiegu pracy. Potrzebują oni tylko wiedzieć jak używać standardowych interfejsów graficznych "point-and-click" oraz jak korzystać z programów aplikacyjnych wspomagających czynności. Aby rozpocząć czynność umieszczoną na liście pracy, użytkownik w prosty sposób wykonuje podwójne kliknięcie na danym zadaniu. Listy prac redukują wymagania umiejętności poprzez automatyzację zadań znalezienia i uruchamiania programów.

## Charakterystyka techniczna IBM FLOWMARK.

IBM FlowMark wersja 2 jest jednym z czołowych produktów w obszarze zarządzania przepływem pracy. Jest on w pełni zgodny z modelem referencyjnym Workflow Management Coalition /WMC/, zaprezentowanym na rys. 2, który jest uważany za standard przemysłowy w tej dziedzinie.

Przedyskutujemy własności systemu FlowMark w kontekście modelu referencyjnego WMC i odniesiemy zaprezentowane możliwości systemu do wymagań technicznych Użytkownika. Spełnianie standardowych wymagań WMC oraz specyfikacji CORBA OMG pozwala systemowi FlowMark na współpracę z innymi otwartymi produktami, w naszym przypadku z LOTUS Notes i Office Objects.



Rys. 2. Model referencyjny przepływu pracy według WMC

IBM FlowMark jest narzędziem do zarządzania przepływem pracy, które pomaga zoptymalizować procesy funkcjonowania przedsiębiorstwa w organizacji. System ten jest używany w fazie analizy i projektowania, dostarczając zasadnicze narzędzie dla czynności BRP (Business Re-engineering Process) oraz w fazie implementacji i działania do dokumentowania i kontroli wykonania procesów.

FlowMark towarzyszy w codziennych czynnościach, w planowaniu i zarządzaniu procesami, wspierając następujące czynności:

- definiowanie i dokumentowanie procesów
- symulacja i testowanie procesów;
- wspieranie i kontrola czynności wykonywanych "ręcznie";
- automatyzacja kierowania czynnościami, które nie wymagają interwencji człowieka
- opieka nad specyfikacją procesu i regułami kontroli.



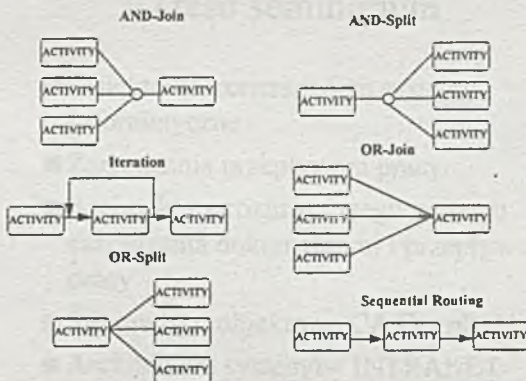
## Narzędzia Definiowania Procesów

FlowMark łączy przetwarzanie klient/serwer z dwiema standardowymi ideami: listą rzeczy do zrobienia i podręcznikiem proceduralnym. Lista rzeczy do zrobienia staje się listą czynności systemu FlowMark, a podręcznik proceduralny staje się modelem przepływu pracy, który zapewnia wykonanie wymaganych procesów. Model przepływu pracy jest prezentowany jako graf przy użyciu zaawansowanego narzędzia graficznego zawartego w systemie. Model graficzny reprezentuje czynności biznesowe, zespoły, które je wykonują, programy, które wspierają procedury, wykonywane "ręcznie" oraz przepływ sterowania i informacji pomiędzy czynnościami. Informacja modelująca przechowywana jest w obiektowo zorientowanej bazie danych zarządzanej przez system zarządzania bazą danych ObjectStore. ObjectStore jest wiodącym obiektowo zorientowanym systemem zarządzania bazą danych, w pełni zgodnym ze specyfikacją ODMG, uważaną za standard przemysłowy w dziedzinie obiektowo zorientowanych baz danych. Model przepływu pracy może zostać udokumentowany jako diagram procesu oraz w formie dokumentacji procesu on-line.

Graficzna reprezentacja modelu obciążenia pracą składa się z dwóch części:

- diagram procesu, który jest rysowany, żeby pokazać poszczególne czynności, ich sekwencje i następstwa,
- określanie notatników, w których są specyfikowane zadania przypisane poszczególnym członkom zespołu; określone są programy i inne szczegóły modelu przepływu pracy.

Diagram procesu jest rysowany za pomocą symboli, które reprezentują czynności, przepływ sterowania i przepływ danych pomiędzy nimi, czynności programu aplikacyjnego, czynności całego procesu oraz bloki czynności (iteracje czynności) są reprezentowane jako węzły skierowanego grafu procesów, a sterowanie, jak również połączenia między danymi, oznaczone symbolami strzałek, są reprezentowane jako łuki tego skierowanego grafu. Typy przepływu sterowania wewnątrz procesu poszczególnych czynności reprezentowanych przez skierowany graf są pokazane na rysunku 3. Graf procesu jest wystarczająco ogólny, aby zapewnić reprezentację dowolnie skomplikowanego procesu.



Rys. 3 Typy przepływu sterowania reprezentowane w diagramie procesu FlowMark

Określanie notatnika zezwala na specyfikowanie następujących właściwości modelu obciążenia pracą:

- kto jest przypisany do czynności
- jaki program i dane są przypisane do wykonania tych czynności
- jakie warunki muszą być spełnione, żeby te czynności mogły się rozpocząć i zakończyć

# Systemy zarządzania przepływem pracy jako narzędzie tworzenia rozproszonych aplikacji

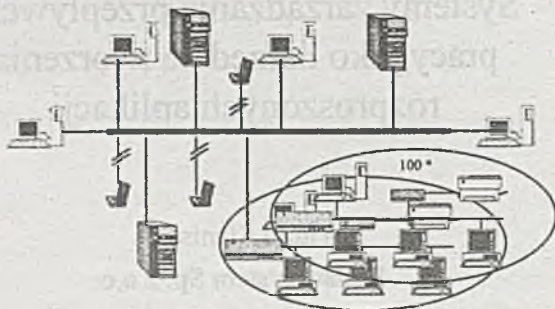
Witold Staniszkis  
Rodan System Sp. z o.o.

## Treść seminarium

- Obiektowo-zorientowane systemy informatyczne
- Zarządzania przepływem pracy
- Architektura rozproszonego systemu zarządzania dokumentami i przepływem pracy
- Środowisko obiektowe CA OpenROAD
- Architektura systemów INTRANET



## Rozproszone przetwarzanie - wspólne zasoby danych



## Wymagane cechy środowiska informatycznego

- Możliwość współpracy heterogenicznych, autonomicznych i rozproszonych systemów (HAR)
- Możliwość odwzorowania procesów działalności przy jednoczesnym wykorzystaniu wielu systemów HAR
- Zapewnienie prawidłowości i odporności systemów w przypadku równoległego dostępu i awarii
- Możliwość wspomagania ewolucji, wymiany, lub rozwoju aplikacji zanurzonych w systemie sterowania pracą



**Zarządzanie rozproszonymi obiektami (CORBA)**  
**Dopasowany model zarządzania transakcjami**



# WYZWANIE

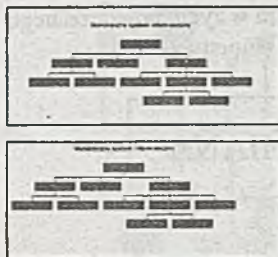
Jedynym pewnikiem w życiu nowoczesnego  
przedsiębiorstwa jest

ZMIANA

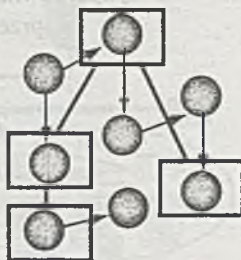


## Alternatywa organizacji oprogramowania SI

MONOLITYCZNA ARCHITEKTURA SI



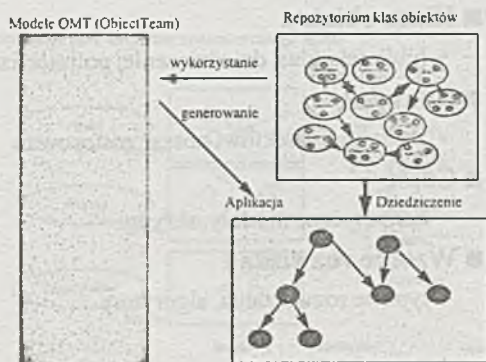
ROZPROSZONA ARCHITEKTURA SI



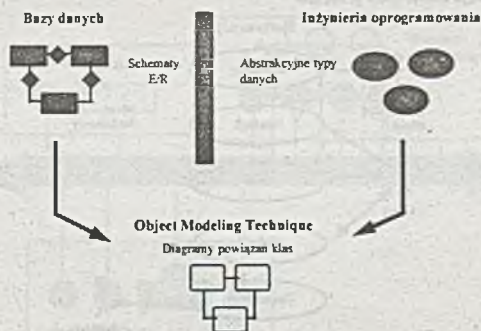
## Rozwój technik programowania obiektowego

- Języki programowania obiektowego
  - C++, Smalltalk, Eiffel
- Architektura obiektowych systemów informatycznych
  - Corba (OMG), OLE (Microsoft)
- Obiektowy model danych
  - Object Database Standard (ODMG-93)

## Tworzenia aplikacji obiektowych



## Koncepcja “Business Object”





## Modularność powielarności obiektów

### ■ Klasa obiektu

- biblioteki klas, dziedziczenie, polimorfizm

### ■ Kontekst

- zbiór klas obiektów, obszar zastosowań

### ■ Część

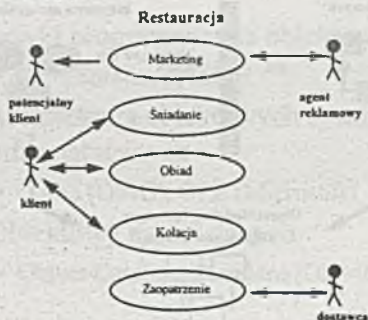
- hermetyczne moduły, skrypty

### ■ Wzorce rozwiązań

- typowe rozwiązania, algorytmy

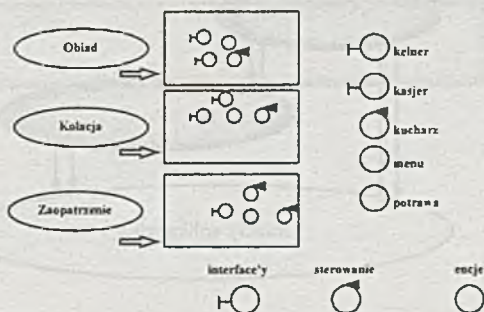
## Notacja obiektowa w reprezentacji procesów (Jacobson)

Definicja scenariuszy procesów

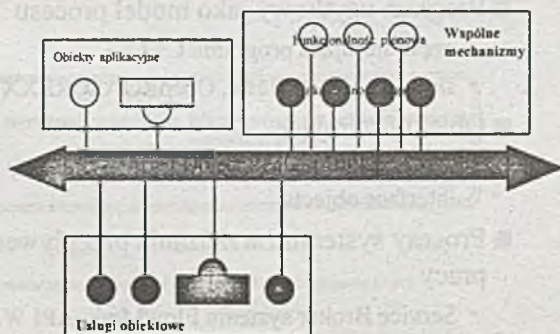


## Notacja obiektowa w reprezentacji procesów (Jacobson)

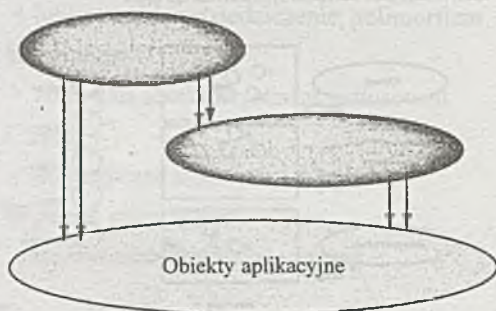
Klasyfikacja jako narzędzie systematycznego opisu



## Common Object Request Broker Architecture



## Struktura dziedziczenia CORBA



## Niezależność od algorytmu procesu

- Program użytkowy jako model procesu
  - część sterująca programu C++
  - skrypty (Visual Basic, OpenROAD, REXX)
- Obiekty sterujące
  - Jacobson - **control objects**, entity objects, interface objects
- Procesy systemu zarządzania przepływem pracy
  - Service Broker systemu FlowMark, API WMC



## Zarządzanie przepływem pracy

zarządzanie przepływem pracy = sterowanie wykonaniem zadań  
+ przydział zadań  
+ śledzenie i kontrola procesów pracy

ORAZ

symulacja procesów pracy i zarządzanie wydajnością pracy

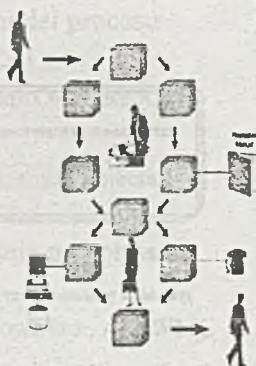
## Zastosowania Systemów Zarządzania Pracą



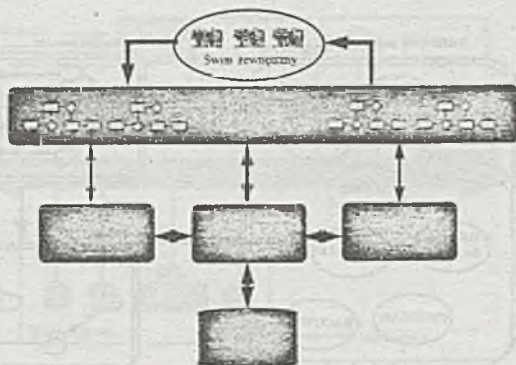
## Zarządzanie przepływem pracy

### ■ IBM FlowMark

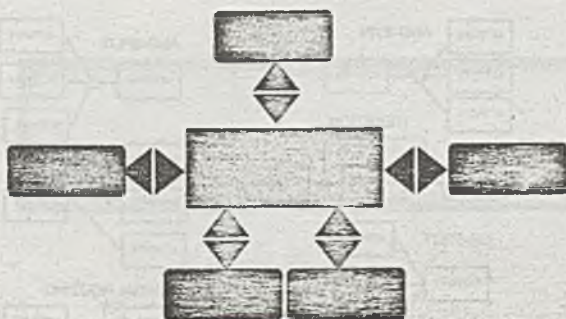
- definiowanie przepływu pracy (modele procesów)
- weryfikowanie modeli procesów
- kontrola wykonania procesów
- rozdział zadań wśród użytkowników



## Przykładowa architektura systemu

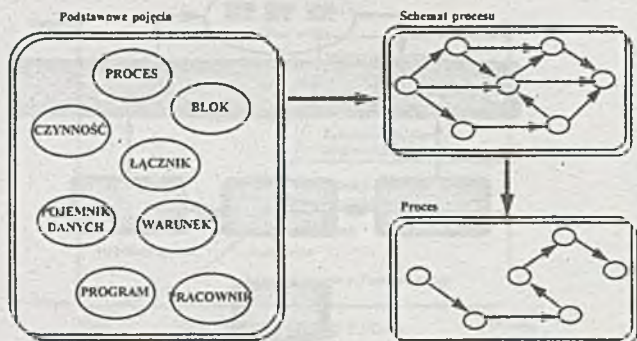


## Architektura Workflow Management Coalition

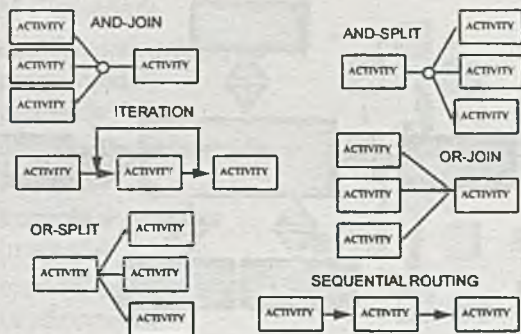




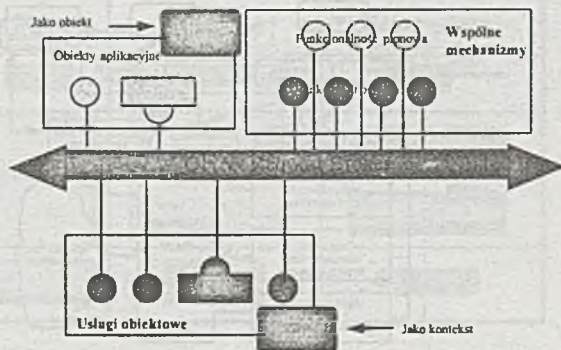
## Modelowanie procesów działalności



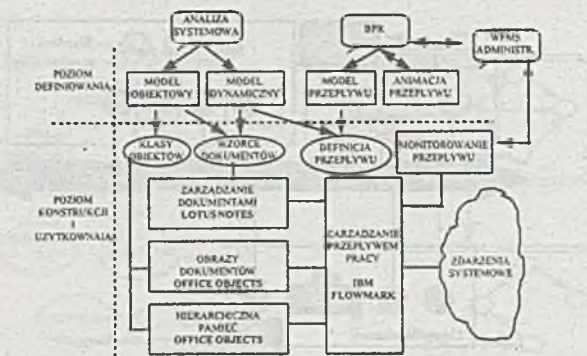
## Elementy definicji algorytmów przepływu



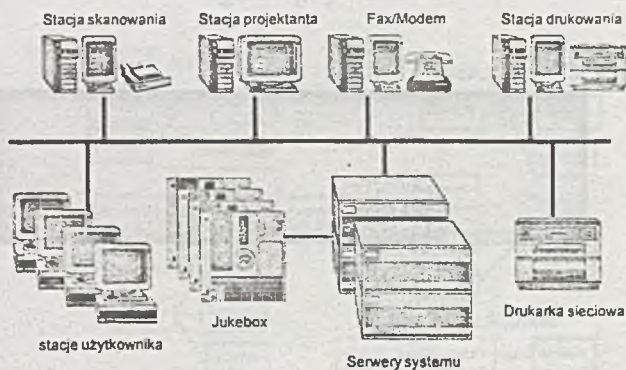
## Model CORBA/WfMC



## Architektura systemu



## Konfiguracja sprzętowa

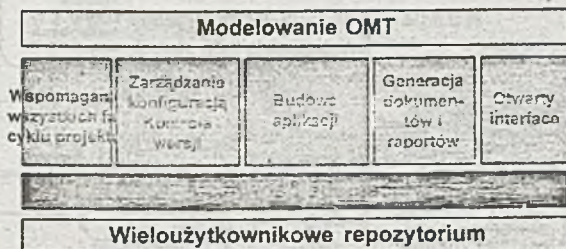




## Wspomaganie cyklu życiowego

Fazy	Modele	Narzędzia
Analiza	Model obiektowy (OMT) Model cyfrowy (OMT) Model funkcjonalny (OMT) Model symulacji między klasami (Cadrel)	Modelowanie
Projektowanie		Modelowanie
Implementacja		Modelowanie
Testowanie	OOPL Model SQL Model	Modelowanie

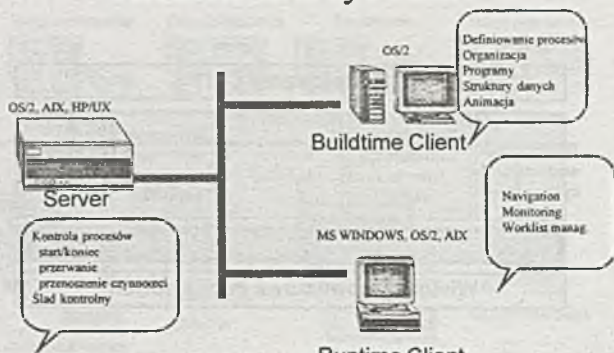
## Architektura ObjectTeam for OMT

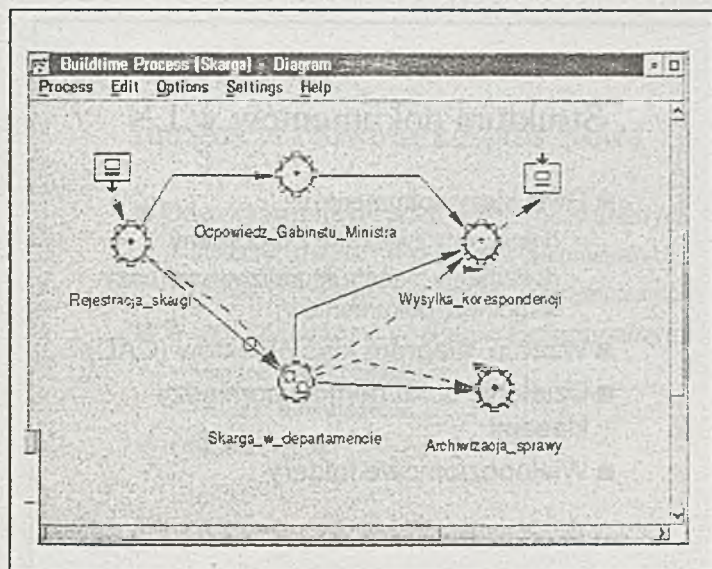


## Modelowanie procesów

- Projektowanie procesów (FlowMark animation facility)
- Projektowanie dokumentów
- Użycie Object Team OMT Tool
  - generowanie opisu organizacji, użytkowników, ról dla FlowMarka
  - generowanie opisu dokumentów dla tworzenia wzorców Lotus Notes

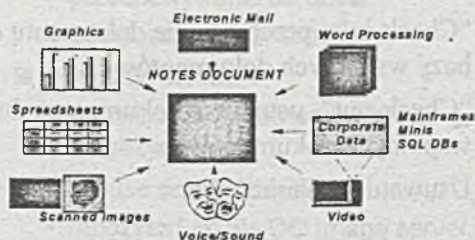
## IBM FlowMark v.2 - Architektura systemu





## Lotus Notes

### Typy danych w dokumentach





## Struktura dokumentów w LN

- Dwie klasy dokumentów:
  - dokumenty związane z procesami
  - dokumenty nieustrukturalizowane (bazy wspólnych dokumentów)
- Wzorce, diagram klas obiektów (CAD)
- Dzielenie dokumentów pomiędzy klasami
- Wielopoziomowe foldery

## Operacje na dokumentach

- Kopiowanie dokumentu - Kopia i Link
- Przesuwanie dokumentu
- "Check-in" - przeniesienie dokumentu do bazy wspólnych dokumentów
- "Check-out"- usunięcie dokumentu z bazy wspólnych dokumentów
- Usuwanie dokumentów

## Integracja FlowMark/Lotus Notes

- Odzworowanie na poziomie konceptualnym
  - Schemat procesu  $\Longleftrightarrow$  wzorce dokumentów
  - Instancja procesu  $\Longleftrightarrow$  folder z dokumentami
- Współdziałanie na poziomie funkcjonalnym
  - Service Broker Manager
  - "Bloki" funkcjonalne
  - Metodologia

## Serwery systemu: Obsługa obrazu

- OO Image server
  - przechowywanie obrazów
  - udostępnianie obrazów
- OO HSM server
  - migracja pól dokumentów LN do magazynów
- Database server (INGRES)
  - repozytorium dla OO Image server i OO HSM server

## Imaging Client/Lotus Notes

- Połączenie poprzez DDE
- Handler do obrazu
- OCR obrazu umożliwiające późniejsze przeszukiwanie
- Operacje na obrazie: View, Print, Edit, Scan (Scan Station), OCR

## OfficeObjects Imaging Set - Skanowanie

- Interface do ponad 20 typów sanerów:
  - flatbed/ADF
  - simple/duplex
  - do rozdzielczości 2400 dpi
  - kolorowe/czarnobiałe
- Używa standardowego protokołu ISIS.
- Obsługuje skanery TWAIN.



## OfficeObjects

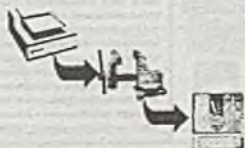
### Imaging Set - Wizualizacja/Wydruk

- Pełne możliwości wyświetlania:
  - skale szarości
  - zoom-in/zoom-out
  - Pan Window
  - Rotacja 90, 180 and 270 degrees
  - Wielostronicowe dokumenty
- Drukuje używając standardowego akceleratora wydruku Windows.

## OfficeObjects

### Imaging Set - Kompresja

- Obsługa standardowych formatów kompresji:
  - TIFF G4 and G3
  - JPEG, JBIG
  - PCX, DCX
- Kompresja “w biegu” - wykorzystanie techniki pipeline:



## OfficeObjects

### Imaging Set - Anotacje

- **Możliwości anotacji::**
  - Tekst wykorzystujący standardowe czcionki Windows
  - Anotacje pismem ręcznym
  - Anotacje w formie nakładek
- **Drukowanie dokumentów z lub bez nakładek.**

## OfficeObjects

### Imaging Set - Look & feel



## OfficeObjects

### Imaging Set - Interface do LN

■ DDE Interface do Lotus Notes umożliwia:

- Skanowanie, wizualizację i wydruk; dokumenty są zapamiętywane zgodnie z regułami określonymi w HSM.
- Interface do programu OCR (Recognita), automatycznie wprowadza rezultat OCR do dokumentu Lotus Notes co umożliwia indeksowanie pełnotekstowe.

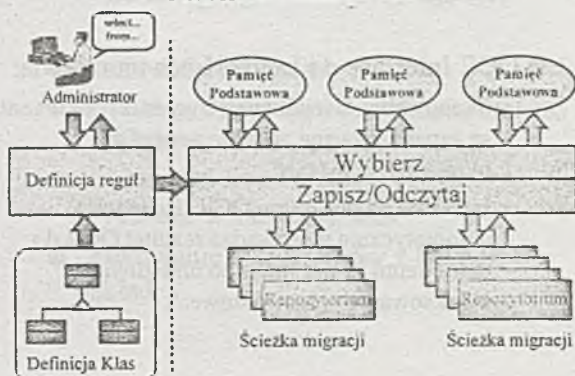
## OfficeObjects

### Imaging Set - Recognita™ OCR

- Rozpoznaje 22 języki (w tym Polski) ze wsółczynnikiem rozpoznania równym 95%; Obsługuje standardowe czcionki i style..
- Automatyczne dekompozycja paragrafów i rozpoznawanie obszarów nie-tekstowych.
- Wyprowadza tekst w formatach powszechnie używanych word processorów (ASCII, WinWord, WordPerfect).



## OfficeObjects Architektura HSM

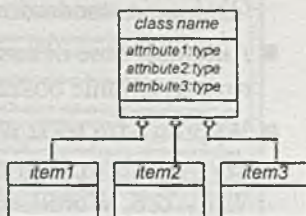


## OfficeObjects Reguły HSM

*Reguły i klasy dokumentów*

*Ogólny format reguły*

```
SELECT items
FROM classes
WHERE attributes = values
```

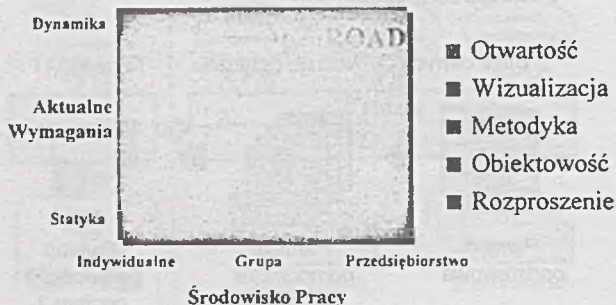


## OfficeObjects Ścieżka migracji HSM

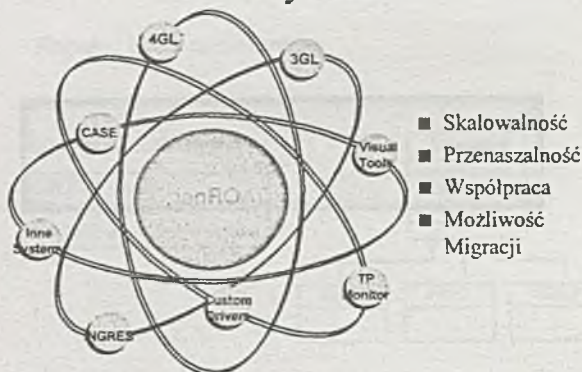
*Przykład ścieżki migracji*



## OPEN Rapid Object Application Development

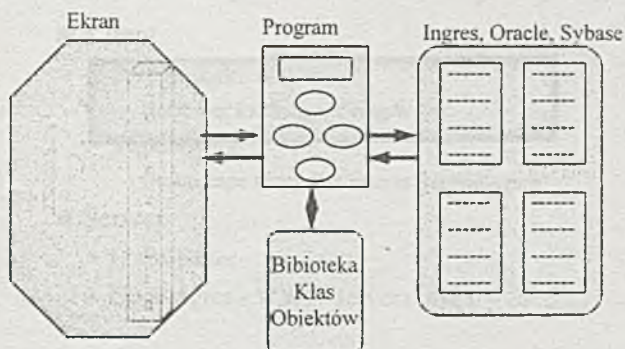


## Elastyczność i Swoboda Wyboru

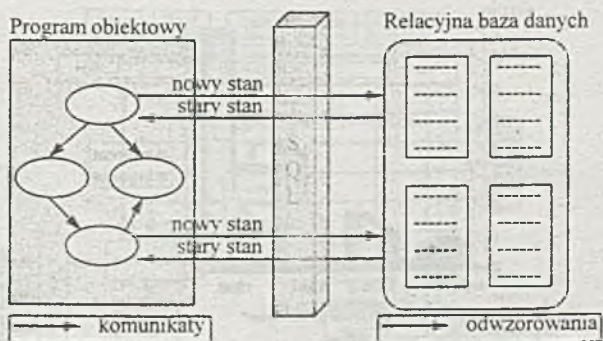




## Środowisko programowania obiektowego

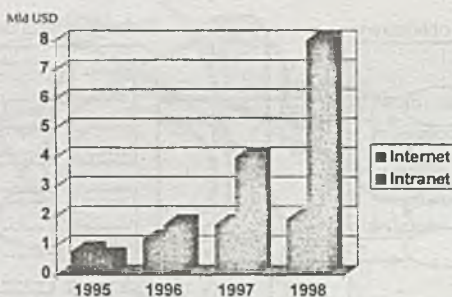


## Obiektowość a relacyjna baza danych





## Intranet - współczesny paradygmat przetwarzania



Rozwój sprzedaży "Web Serwerów"

Zródło: Zana Research (Economist, 13.01.96)

## Oprogramowanie dla Intranet

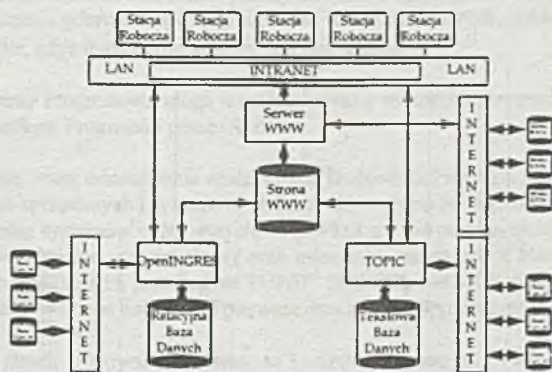
### ■ Klient

- Przeglądarki WWW
  - Netscape, Exchange, GroupWise
- Middleware
  - Groupscape firmy Brainstorm Technologies

### ■ Serwer

- Lotus Notes
- OpenIngres - WWW server option

## Proponowana architektura bazy informacji





**Jerzy Sitarz, Jarosław Kaczorowski**  
**CSBI S.A.**

# **Szybkie tworzenie aplikacji w wersji 8 Progressa**

## Informacje o autorach

Jerzy Sitarz pracuje jako programista od 20 lat, w CSBI S.A. (w Biurze Techniczno-Handlowym w Katowicach) od 2 lat, w języku Progress 4GL programuje od lat 5.

telefony: (0-32) 153 72 79, (0-32) 153 72 80

e-mail: jsi@csbi.katowice.pl

Jarosław Kaczorowski jest pracownikiem Zespołu Progress w CSBI S.A. w Warszawie od ponad 3 lat.

telefony: (0-22) 659 03 42, (0-22) 23 58 32

e-mail: jki@csbi.waw.pl

## 1. Historia Progressa.

### Co to jest Progress?

Progress jest **kompletnym środowiskiem** do tworzenia i uruchamiania oprogramowania. W skład tego środowiska wchodzi między innymi: własny system zarządzania relacyjnymi bazami danych, język programowania czwartej generacji Progress 4GL, bogaty zestaw narzędzi zarówno dla programistów, jak i finalnych użytkowników aplikacji stworzonych w Progressie, a także *drivers* do baz danych oraz narzędzi typu CASE różnych producentów.

Aplikacje progressowe mogą pracować także na bazach danych innych producentów.

Bazy danych Progressa wyróżniają się wśród innych relacyjnych baz danych dużą liczbą atrybutów pól oraz kompresją danych (!).

Język programowania czwartej generacji Progress 4GL jest dużo silniejszy od SQL (około 3 do 7 razy zwięźlejszy kod) i bardzo silnie wykorzystuje atrybuty pól bazy danych. W porównaniu z innymi narzędziami, Progress bardzo ułatwia oprogramowanie interfejsu użytkownika.

Zestaw narzędzi do tworzenia aplikacji zawiera silny program śledzący (*debugger*), możliwość przetłumaczenia gotowych aplikacji na wiele wersji narodowych (dokładniej – samego interfejsu użytkownika, gdyż logika programu nie jest tłumaczona).

Programy Progressowe mogą współpracować z wieloma serwerami innych baz danych przez własne interfejsy Progressa i przez ODBC.

Kolejną, wartą odnotowania właściwością środowiska Progressa jest możliwość pracy na wielu platformach sprzętowych i systemowych (lista referencyjna zawiera około 3 tysięcy pozycji maszyn i kilkadziesiąt systemów operacyjnych, w tym kilkanaście odmian samego Unixa) w architekturze klient-serwer, wielodostępnej (*host*) oraz mieszanej, łącząc się z bazami danych przy pomocy wszystkich popularnych protokołów: TCP/IP, SPX/IPX, NetBIOS, SNA, DECnet, CTOS Cluster. W Polsce stosowane są najczęściej pierwsze dwa z wyżej wymienionych protokołów.

W tej chwili oferowane i używane są 3 wersje Progressa (6, 7 i 8).

Bazy danych w wersji 6 zawierają wiele atrybutów. Język programowania tej wersji obsługuje interfejs znakowy. Zestaw narzędzi zawiera środki do dystrybucji oprogramowania.

Wersja 7 Progressa zawiera zarówno interfejsy znakowy jak i graficzny. Baza danych tej wersji ma znacznie zwiększoną liczbę atrybutów i metod. Zwiększono jej wydajność o około 50%. Programy w wersji 7 Progressa mogą być pisane zdarzeniowo (*Event Driven*) z wykorzystaniem technik obiektowych. Zestaw narzędzi w wersji 8 rozszerzony został o narzędzia dla tworzenia aplikacji (*User Interface Builder*) i raportów (*Report Builder*).

W wersji 8 Progressa także zwiększono wydajność bazy danych (szacunkowo o 30%). Zmian w samym języku programowania wprowadzono niewiele (modyfikowalne przeglądarki). Z nowych narzędzi w tej wersji warto odnotować program dla (półautomatycznego) tłumaczenia interfejsów użytkownika i zestaw narzędzi dla zarządzania tworzeniem i wdrażaniem aplikacji Roundtable.



## 2. Zasady tworzenia oprogramowania.

Każdy, kto sprzedaje dzisiaj systemy dla tworzenia programów twierdzi, że przy pomocy jego systemu można tworzyć bardzo szybko sprawne oprogramowanie. Zwykle przy tym dodaje się, że może to robić prawie każdy, kto zna tylko problem, który ma rozwiązać.

Można powiedzieć, że zdecydowanie zanika profesja dawnego programisty. Programisty, czyli człowieka, który znał zarówno sprzęt na którym tworzył oprogramowanie (i sprzęt, na którym to oprogramowanie miało być uruchamiane), zagadnienia merytoryczne związane z pisanym programem, jak i miał ogólną wiedzę o zasadach tworzenia oprogramowania.

Dzisiaj znajomość sprzętu przestaje być w ogóle potrzebna, ponieważ ujednoliciły się interfejsy dostępu do zgromadzonych danych (plików, baz danych), jak i interfejsy użytkownika (terminale, klawiatury).

Znajomość zagadnień merytorycznych jest ważna. Ale bądźmy szczerzy. Dzięki stałej wymianie informacji większość wielkich zagadnień jest już sformalizowana. Nie trzeba już odkrywać Ameryki, projektując systemy finansowe, magazynowe, sprzedaży, zarządzania czymkolwiek czy temu podobne. Przy tego typu zagadnieniach zaczyna pojawiać się takie nasycenie oprogramowaniem, że bliska jest chwila, że programowanie określonego zagadnienia będzie sprowadzało się do zadania parametrów przygotowanym wcześniej półproduktom.

Jednak oprogramowanie będzie tworzone zawsze. I dlatego warto pamiętać o zasadach tworzenia oprogramowania. Tu będzie położony akcent na jedno zagadnienie: **dekompozycję**. Dekompozycja – podzielenie skomplikowanego zagadnienia na mniejsze części – jest niezbędnym warunkiem dla rozwiązania dużych zagadnień.

Zastanówmy się przez chwilę, co to znaczy „szybko tworzyć oprogramowanie”. Jedną skrajnością mogą być duże projekty, które są realizowane od początku do końca, z dostawą sprzętu, narzędzi i często pełnym szkoleniem przyszłych użytkowników. Inną skrajnością jest modyfikacja istniejącego oprogramowania w związku ze zmianą przepisów.

Z punktu widzenia Progressa należy mieć możliwość:

- tworzenia dużych projektów, mając na względzie zarówno etap pierwszy rozwiązania, jak i bardzo prawdopodobny dalszy jego rozwój (praca przy tworzeniu bazy danych i podstawowej logiki programu);

- tworzenia specyficznych raportów wykorzystujących zgromadzone dane;

wprowadzaniu prostych zmian związanych z wprowadzeniem nowych atrybutów.

Przy dużych projektach zawsze podstawowym działaniem będzie dekompozycja projektu, czyli podział na prostsze zadania. Typowym elementem wiążącym wszystkie zadania jest baza danych, której dobra definicja jest podstawą dalszego rozwoju aplikacji (logika programu i interfejsy użytkownika powstają niemal automatycznie). Wskazaniem jest jednak na tym etapie korzystanie z narzędzi wspomagających (CASE). Nie ulega najmniejszej wątpliwości, że przy tego typu zagadnieniach muszą działać najlepsi programiści.

Przy tworzeniu specyficznych raportów większość poważnych systemów nie wymaga się współdziałania programisty. Wysokiej klasy systemy, w tym Progress, umożliwiają tworzenie profesjonalnych raportów przez osoby znające tylko swoją aplikację (prowadzenie użytkownika przez system przy tworzeniu raportu).

Przy wprowadzaniu zmian, przy modyfikacji atrybutów pól i tablic są dwa istotne zagadnienia: niebezpieczeństwo wprowadzenia niespójności bazy danych z jednej strony i konieczność szybkiego dostosowania aplikacji do zmieniających się warunków (wprowadzenie danych dla określonych atrybutów).



### 3. Tworzenie bazy danych.

Wszystkie wersje Progressa mają bardzo rozbudowany opis tablic i pól bazy danych. Szczególnie obszerny jest zestaw metod i atrybutów pól i tablic dla wersji 7 i 8 Progressa. Warto w tym miejscu wspomnieć, że serwer progressowy, oprócz tablic ma możliwość obsługi sekwencji, które są obsługiwane poza transakcjami.

Na poziomie tablic w Progressie najistotniejszym elementem są wyzwalacze (*triggery*). Każda tablica może mieć przypisanych 4 wyzwalacze związane z dodaniem nowego rekordu do tablicy, skasowaniem rekordu z tablicy, zmianą pól rekordu i z wyszukaniem rekordu. Każdy z tych wyzwalaczy jest programem (w języku Progress) z możliwością dostępu do pól bieżącego (tworzonego, wyszukanego) rekordu, zarówno przed zmianą jak i po zmianie pola. Typowym przykładem działania wyzwalacza może być obsługa zmiany numerów faktur w pozycjach faktury w przypadku zmiany numeru faktury. Oczywiście jest możliwość przerywania operacji, jeśli dane nie spełniają określonego warunku. Pozwala to tworzyć bazę danych pozwalającą na wprowadzanie tylko spójnych danych (bez pisania konkretnego programu wprowadzania danych).

Na poziomie pól danych dysponujemy jednym wyzwalaczem (reagujący na przypisanie wartości do pola). Kilkadziesiąt atrybutów pól pozwala w momencie tworzenia opisu pola określić wiele aspektów pola.

Pierwsza grupa atrybutów dotyczy identyfikacji pola. Jest to nazwa pola, numer kolejny pola (określa kolejność na formatkach) i etykiety (sąsiadująca i kolumnowa). Do tej grupy atrybutów należy możliwość zadawania atrybutów wszystkich tekstów (etykiet, tekstów odpowiedzi itp.), które muszą być zachowane podczas tłumaczenia na inne języki.

Druga grupa atrybutów dotyczy typu pola. Jest to typ pola (znakowy, całkowity, zmiennoprzecinkowy, logiczny), liczba cyfr znaczących, format i liczba wystąpień. Również do tej grupy można przypisać sposób przedstawiania danego pola. Dla przykładu pole o wartości całkowitej może być przedstawione jako pole do wprowadzania wartości lub jako suwak, jako przełączniki „radiowe” lub jako pole rozwijane. Do tej grupy atrybutów należy również możliwość zadania wartości domyślnej dla pola oraz możliwość zadawania atrybutów wszystkich tekstów (etykiet, tekstów odpowiedzi itp.), które muszą być zachowane podczas tłumaczenia na inne języki.

Kolejna grupa atrybutów dotyczy sprawdzania poprawności wprowadzania danego pola. Jest tu zarówno standardowy tekst odpowiedzi, który pojawi się przy przejściu do danego pola, jak i warunek (w języku Progress), sprawdzający poprawność pola, tekst wyświetlany w przypadku gdy warunek nie jest spełniony. Istnieje możliwość zadania wymogu, że jakaś wartość do danego pola musi być koniecznie wprowadzona przy tworzeniu nowego rekordu.

Wymienione wyżej atrybuty pozwalają bardzo dobrze określić wiele cech aplikacji jeszcze na etapie poprzedzającym samo pisanie programu.

Tworzenie i modyfikacja tablic i pól bazy danych dokonywana jest w środowisku graficznym.



## 4. Tworzenie dużych programów.

Większość współczesnych systemów tworzenia dużych programów daje narzędzia do graficznego tworzenia interfejsu z użytkownikiem. Progress również oferuje bardzo silne narzędzie dla tworzenia takiego interfejsu – *User Interface Builder* (UIB).

UIB pozwala na tworzenie zarówno procedur strukturalnych, jak i programów dla obsługi okien (interfejsu) użytkownika.

Procedury strukturalne pozwalają na podział kodu programu na mniejsze jednostki: definicje, procedurę główną i procedury wewnętrzne. Każda procedura wewnętrzna (mimo że jest tylko jedną z części składowych większego pliku) może być bardzo szybko wprowadzona do edytora przez wybór z menu.

Okna użytkownika oprócz tych elementów, które są zawarte w procedurach strukturalnych, dają możliwość traktowania jako osobnych jednostek podprogramów obsługi zdarzeń dla wszystkich obiektów występujących w oknie. Tak przeprowadzona **dekompozycja** pozwala na podział dużego programu na wiele dobrze określonych fragmentów, które można traktować jako osobne jednostki.

Ważne są 4 poziomy obiektów, które występują w oknach.

Pierwszy poziom dotyczy obiektów graficznych. Pola i zmienne Progressa mogą być przedstawiane na wiele sposobów. Każde pole i zmienna mają swój domyślny (zdefiniowany w słowniku danych) format przedstawienia, który może być oczywiście w przypadku konkretnego użycia tego pola lub zmiennej zmieniony. Wstawienie i modyfikacja obiektów graficznych reprezentujących pola i zmienne jest intuicyjne i wykonywane jest w środowisku graficznym. Oczywiście obsługa specyficznych akcji wymaga znajomości języka Progress.

Drugi poziom obiektów to zestawy. Zestaw zawiera zbiór obiektów graficznych, wraz z procedurami dla obsługi zdarzeń dla tych obiektów. Użytkownik może dowolnie modyfikować, w ramach konkretnego okna, fragmenty użytego zestawu.

Trzeci poziom obiektów to szablony (*templates*) – „półprodukty”. Szablon zawiera kompletne okno. Okno to może mieć część obiektów, które są potrzebne w aplikacji. Dzięki możliwościom wbudowanego preprocesora szablon może być przygotowany do obsługi np. dowolnego pliku. Również w szablonie programista może dowolnie zmieniać dowolne fragmenty jego kodu.

Czwarty poziom obiektów to obiekty typu smart (*SmartObjects™*, czyli inteligentne obiekty). Obiekty smart tworzą zamkniętą całość i komunikują się przez specjalny system połączeń (*Links*). Obiekty typu smart nie mogą być modyfikowane przez programistę, który ich używa i na odwrót: zmiana definicji obiektu typu smart powoduje zmianę we wszystkich aplikacjach.

Bardzo istotna jest komunikacja między obiektami typu smart. Każde dwa połączone obiekty udostępniają zestawy procedur. Dokładniej, każdy obiekt typu smart zakłada, że połączony z nim

obiekt ma ściśle określony zestaw procedur (metod), które uruchamiają się w przypadku wystąpienia określonych zdarzeń.

Przykładem połączenia obiektów typu smart może być połączenie kontenera z innym obiektem. W tym przykładzie zakłada się że po każdej stronie jest procedura obsługi skasowania (*destroy*) obiektów.

Należy wyraźnie podkreślić, że zestaw procedur dla obu elementów pary połączenia (*source* i *target*) jest z reguły różny.

## 5. Tworzenie skomplikowanych raportów.

Raporty tworzone są przez doświadczonych projektantów jako fragmenty aplikacji, ale wiele raportów powstaje dla zobrazowania sytuacji na podstawie (zwykle dużej porcji) danych znajdujących się w bazie. Absolutnym standardem są zapytania do bazy na poziomie zdań SQL, ale jest to sposób wymagający znajomości zarówno logicznej struktury bazy, jak i języka SQL.

Progress dostarcza wyrafinowanego narzędzia dla tworzenia skomplikowanych raportów. Jest to Report Builder. Narzędzie to pozwala dzięki podziałowi raportu na części składowe (dekompozycja) na jego szybkie przygotowanie, zmianę i wykonanie.

Praca z nowym raportem rozpoczyna się od określenia tablic i pól, z których dane wejdą w skład raportu. Można zadać zarówno skomplikowane połączenia między tablicami, jak i filtry dla wyboru tylko pewnych rekordów.

Następnie rekordy są dzielone na grupy. Może wystąpić do 8 grup. Grupy służą dla dwóch celów. Z jednej strony dzięki grupom można przeprowadzać sortowanie rekordów. Drugim celem wprowadzenia grup jest otrzymywanie danych zbiorczych (zagregowanych) dla grupy.

Kolejnym elementem jest odpracowywanie nagłówków i stopek. Nagłówki i stopki można określić dla całego dokumentu, dla strony i dla każdej z 8 grup.

Nagłówki, stopki wydruk samego rekordu mogą składać się z wielu linii.

W każdej z tych linii można wstawiać teksty stałe, pola z bazy danych, pola wyliczane (wirtualne) i funkcje specjalne (np. numer strony, rekordu itp.). Krój pisma (wielkość, czcionka, kolor, wytuszczenie itp.) każdego fragmentu może być wybrany zgodnie z życzeniem osoby projektującej raport.

Tak przygotowany raport można wstępnie zobaczyć na ekranie. Czasami nic więcej nie trzeba robić.

Raport taki można oczywiście wyprowadzić na drukarkę (dowolną w ramach systemu Windows), a przy pomocy dodatkowego pakietu Report Server także na drukarkę podłączoną do serwera pracującego pod kontrolą MS-Windows NT lub którejś z odmian systemu Unix.

Raport można zachować i wywołać ponownie. Raport można wywołać zarówno z poziomu Report Buildera, jak i z programu aplikacyjnego. Ta ostatnia możliwość może być pomocna dla programistów projektujących duże aplikacje.



## **6. Praca użytkowników z programem RESULTS™.**

Użytkownicy końcowi mogą dostać wiele narzędzi do pracy. Jedną skrajność to czyste środowisko uruchomieniowe (*run-time*), gdzie użytkownik może pracować tylko z gotowymi programami przygotowanymi i skompilowanymi w środowisku rozwojowym. Drugą skrajność to pełne środowisko rozwojowe.

Dla niezbyt wymagających użytkowników Progress dostarcza wygodne narzędzie, które w środowisku graficznym i znakowym pozwala na:

- przygotowanie prostych raportów i przeglądów z wielu tablic i różnych baz,
- przygotowanie programów do edycji pól dowolnej tablicy,
- przygotowanie programów dla tworzenia etykiet z dowolnej tablicy,
- przygotowanie eksportu danych w jednym z kilkudziesięciu formatów,
- utworzenie programu realizującego jedną z powyższych funkcji, który można włączyć do aplikacji bądź uruchomić z poziomu RESULTS.

Praca z programem RESULTS jest bardzo intuicyjna. Aby uzyskać wyniki, nie trzeba pisać ani jednego wiersza kodu. Wydaje się, że takie narzędzie może być pomocne wielu administratorom systemów.

## **7. Praca z Roundtable.**

Przedstawione dotychczas narzędzia przeznaczone były dla tworzenia aplikacji przez jedną osobę lub niewielką grupę osób. Choć sam proces tworzenia można dekomponować i łatwo utworzyć działającą aplikację, to zostaje druga strona medalu – utrzymanie aplikacji.

Jeśli przedsiębiorstwo dysponuje dużą lub bardzo dużą aplikacją, zainstalowaną u wielu użytkowników, w wielu miejscach i w różnych wersjach, to problem dokonania choćby najmniejszej zmiany w kodzie aplikacji jest często problemem nie do przejścia.

Dla takich systemów przeznaczony jest Roundtable – nowe narzędzie dostępne w wersji 8 Progressa. (Dla wcześniejszych wersji Progressa można je także nabyć bezpośrednio od producenta, firmy StarBase.)

RoundTable prowadzi ścisłą administrację wszystkich wersji wszystkich obiektów aplikacji. Obiektami są zarówno pola bazy danych jak i same bazy danych, dokumentacja i skrypty uruchomieniowe, wersje źródłowe i wykonywalne programów.

Bardzo ważnym pojęciem w RoundTable jest pojęcie obszaru roboczego i zadania.

Obszar roboczy udostępnia programiście lub grupie programistów konkretne wersje obiektów jednej lub kilku aplikacji.

Zadanie jest wykonywane w celu dokonania zmian (poprawki, modyfikacji) zleconej przez konkretnego kierownika konkretnemu programiście.

Każda wersja obiektu może być zapamiętywana (między innymi) w obszarze związanym z zadaniem, z grupą zadań, centralnie w obszarze roboczym.

Pozwala to, aby programiści zupełnie niezależnie testowali różne fragmenty aplikacji wykorzystując obiekty utworzone przez innych.

Bardzo istotna jest droga kolejnych wersji obiektów z obszarów zadań do obszarów roboczych. Żaden obiekt nie może być przeniesiony na wyższy poziom, jeśli nie spełnia odpowiednich warunków. Pozwala to gromadzić w obszarze roboczym aktualne wersje wszystkich obiektów i zapewnić bardzo wysoką jakość końcowego oprogramowania, instalowanego u użytkowników.

Można utworzyć nowy obszar roboczy, który dziedziczy wszystkie obiekty kilku aplikacji z wielu innych obszarów roboczych i posiada własne aplikacje.

Wszystko to jest wykonywane w graficznym lub znakowym środowisku i cechuje się wysokim stopniem automatyzacji.

Powstające w środowisku RoundTable aplikacje są tworzone z jednej strony bardzo szybko, a z drugiej strony są podstawy sądzić, że wykonane zmiany są bardzo stabilne.

## **8. Wnioski.**

Jak przedstawiono powyżej, narzędzia Progressa stanowią idealną platformę dla tworzenia i modyfikacji aplikacji zarówno dla dużych domów software'owych, dla małych grup programistów, jak i dla użytkowników końcowych.

Omówione narzędzia: Data Dictionary, User Interface Builder, Report Builder, RESULTS i RoundTable stanowią tylko podstawowy zestaw.

Silny debugger (praca krokowa na poziomie języka Progress) jest bardzo pomocny przy testowaniu procedur.

Możliwość półautomatycznej translacji aplikacji na różne języki narodowe w „Europie bez granic” będzie coraz bardziej istotna. Duże projekty mogą być rozwijane w wielu wersjach językowych.

Tak więc Progress jest tym systemem, który należy wybrać dla szybkiego tworzenia aplikacji.



*Wiosenna Szkoła PTI  
Świnoujście '96*

*Narzędzia firmy SAPIENS International jako  
środek realizacji metodologii RAD.*

*SYSTEM 2000 Sp. z o.o.  
Maciej Kwiatkowski*



**PRZEDSIĘBIORSTWO  
ROZWOJU I INNOWACJI Sp. z o.o.**  
00-048 Warszawa, ul. Mazowiecka 12  
tel. 27 42 62 fax 27 15 99 tlx 817096 kur pl  
Konto: PKO BP. XV Oddz. w Warszawie nr 1658-11178-136



## **Informacja o firmie SYSTEM 2000.**

Przedsiębiorstwo Rozwoju i Innowacji SYSTEM 2000 sp. z o.o. z siedzibą w Warszawie zostało założone w 1986 roku.

Specjalizujemy się w tworzeniu, sprzedaży oraz wdrażaniu systemów komputerowych opartych o nowoczesne generatory aplikacji firmy Sapiens International Corporation N.V. oraz metodologię RAD (Rapid Application Development), jednocześnie oferujemy doradztwo techniczne i organizacyjne oraz szkolenia i współpracę w dziedzinie wykorzystania tych narzędzi. W czasie naszej działalności współpracowaliśmy między innymi z Narodowym Bankiem Polskim, Bankiem Śląskim, Kancelarią Sejmu RP, PKO BP, PZL "Okęcie", TUiR "WARTA".

Z firmą Sapiens związani jesteśmy od 1991 roku kiedy to, w drodze konkursu, staliśmy się przedstawicielem i wyłącznym dystrybutorem jej produktów w Polsce. Do roku 1991 SYSTEM 2000 wykonywał oprogramowanie dla małych i średnich firm. Od czasu zawarcia umowy z SAPIENS-em cały wysiłek włożyliśmy w zbudowanie zespołu specjalistów, którzy swoje doświadczenie w dziedzinie tworzenia złożonych systemów informatycznych w oparciu o generatory aplikacji oraz doświadczenie w pracach nad systemami bankowymi zastosowali m.in. w pracach nad oprogramowaniem dla Centrali Narodowego Banku Polskiego oraz Banku Śląskiego S.A. Z naszymi klientami, w razie potrzeby, współpracować mogą obok naszych specjalistów również specjaliści zagraniczni, zatrudnieni w Sapiens International.

Dla Centrali PKO BP stworzyliśmy system obsługujący dystrybucję Powszechnych Świadczeń Udziałowych NFI.

W Banku Śląskim powstał generator sprawozdań finansowych. Aplikacja ta umożliwia tworzenie wielu typów sprawozdań na podstawie danych otrzymywanych z różnych systemów eksploatowanych w banku.

Kancelaria Sejmu RP wykorzystuje generator aplikacji SAPIENS Ideo do tworzenia aplikacji Client/Server na platformie HP-UX z dostępem do RDBMS Ingres.

W Centrali NBP w Warszawie, wspólnie ze specjalistami Departamentu Informatyki, korzystając z generatora aplikacji SAPIENS, opracowaliśmy specjalizowany system obsługi rachunków banków - SORB.

W Banku Śląskim powstał generator sprawozdań finansowych. Aplikacja ta umożliwia tworzenie wielu typów sprawozdań na podstawie danych otrzymywanych z różnych systemów eksploatowanych w banku.

Nasza specjalność to systemy "na miarę". Narzędzia, które stosujemy i promujemy oraz nasze doświadczenie w stosowaniu metodologii RAD (szybkiego tworzenia aplikacji) pozwalają nam tworzyć duże systemy o wysokiej jakości i dobrze dopasowane do potrzeb użytkownika. Jednocześnie stosowanie generatorów firmy Sapiens radykalnie skraca czas tworzenia i modyfikacji aplikacji, co pozwala nie tylko w krótkim czasie przekazać system do eksploatacji ale i szybko dostosowywać już działający do zmieniających się wymagań.



# SAPIENS International Education Center



## *The Wise Solution*



## *Critical Business Issues*

- Increased time-based competition
- Rapidly changing marketplace
- Re-engineering pressures
- Runaway costs



RAD - 1



## *Challenges Facing MIS*

- Increased application backlogs
- Runaway costs (high maintenance)
- Meeting user expectations (your internal customers)
- Blending new technologies while leveraging legacy investment



## *Challenges Facing MIS*

### **Top Three Systems Development Issues**

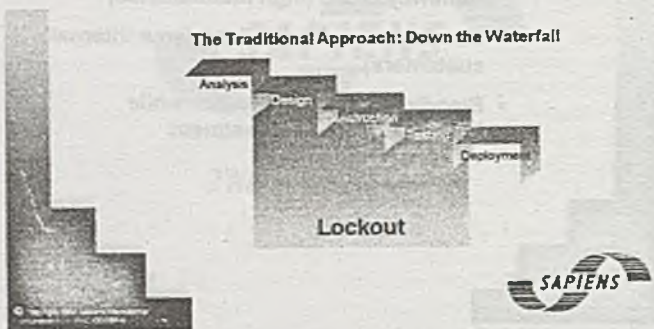
- Re-engineering business processes through information technology
- Aligning information systems and corporate goals
- Improving the systems development process

Source: 1993 CSC Index



## *Challenges Facing MIS*

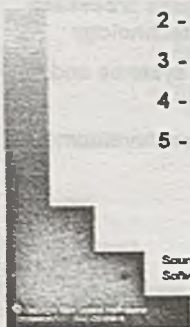
Improving the systems development process



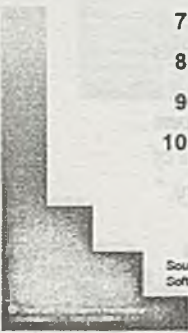
## *Most Serious MIS Application Risks*

- 1 - Inaccurate metrics
- 2 - Inadequate measurement
- 3 - Excessive schedule pressure
- 4 - Management malpractice
- 5 - Inaccurate cost estimating

Source: Assessment and Control of  
Software Risks, T. Capers Jones, 1994



## *Most Serious MIS Application Risks*

- 
- 6 - Silver bullet syndrome
  - 7 - Creeping customer requirements
  - 8 - Low quality
  - 9 - Low productivity
  - 10 - Cancelled projects

Source: Assessment and Control of  
Software Risks, T. Capers Jones, 1994



## **RAD**

### *Meeting the Challenge*





## *RAD: Meeting the Challenge*

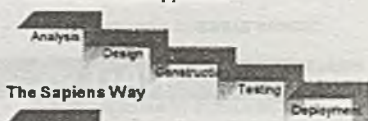
*"RAD is an application development methodology which uses computerized application development tools and human interaction techniques in a tightly woven fashion to achieve the goals of high speed and high quality."*

— James Martin,  
Rapid Application Development

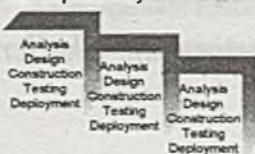


## *RAD: Meeting the Challenge*

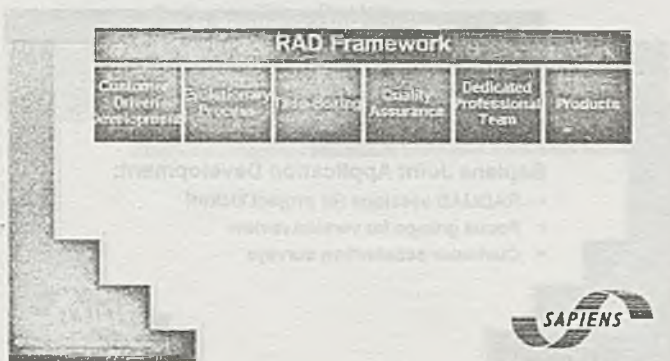
### **The Traditional Approach**



### **The Sapiens Way**

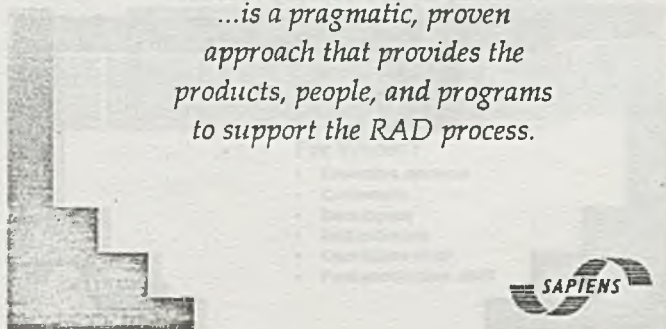


## Framework of RAD Methodology



## The Sapiens Way

*...is a pragmatic, proven approach that provides the products, people, and programs to support the RAD process.*



## *The Sapiens Way*

...Enables Customer-Driven Development



Sapiens Joint Application Development:

- RAD/JAD sessions for project kickoff
- Focus groups for version review
- Customer satisfaction surveys



## *The Sapiens Way*

...Creates an Evolutionary Process





## *The Sapiens Way*

### ...Enforces Time-Boxing



## *The Sapiens Way*

### ...Establishes Quality Assurance



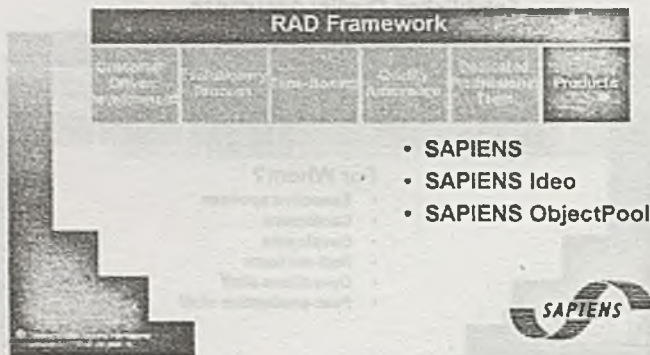
## *The Sapiens Way*


...Builds a Dedicated Professional Team



## *The Sapiens Way*

...Is Supported by an Integrated Suite of RAD Products



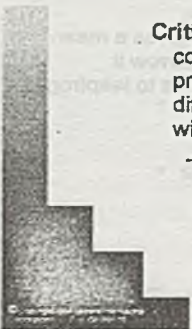
A stylized graphic of a staircase or steps, composed of several horizontal and vertical lines, located on the left side of the page.

## *The Sapiens Way at Work*



## *The Sapiens Way at Work*

**Siemens Energy and Automation (Atlanta, GA)**

A stylized graphic of a staircase or steps, composed of several horizontal and vertical lines, located on the left side of the page.

**Critical Business Issue:** "We need to meet continual customer requests for changes in product configuration and delivery. We want to differentiate ourselves by offering better services with shorter delivery cycles."

— John Murphy, Director of Corporate MIS





## *The Sapiens Way at Work*

**Siemens Energy and Automation (Atlanta, GA)**

**MIS Challenge:** Rebuild the order-processing system to improve functionality and performance

**Results:**

- Doubled development productivity
- 30% additional functionality
- One new function saved sales force 30,000 hours annually
- Response time <1 second for 85% of 75,000 transactions/day



## *The Sapiens Way at Work*

**Siemens Energy and Automation (Atlanta, GA)**

**Conclusion:** "The Sapiens Way gives a means to remain competitive today. Tomorrow it represents a platform we can use to leapfrog our competition."

— John Murphy





*SAPIENS is the rapid application  
development and maintenance platform  
for IBM mainframes*

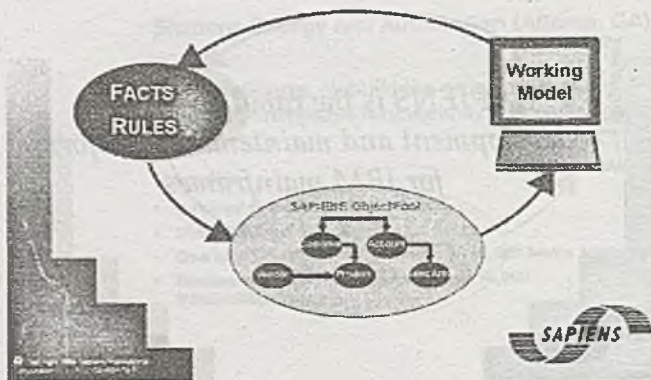


## *The SAPIENS Technology*

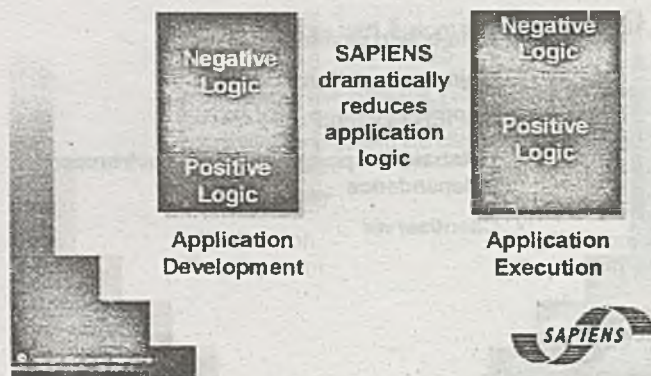
- Rule-based development
- Object-orientation
- SAPIENS ObjectPool
- Database & operating system environment independence
- Client/server



## *RAD with SAPIENS*

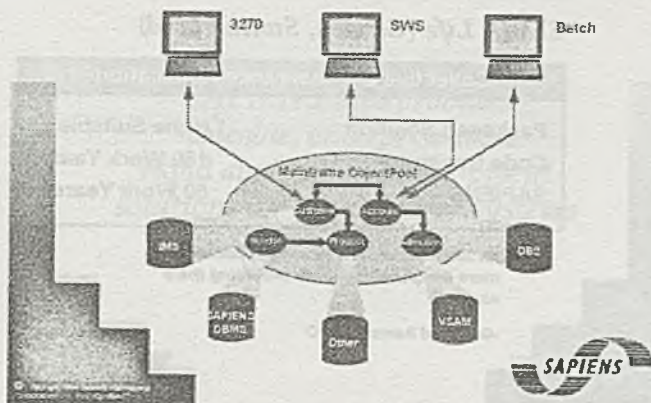


## *The Power of Positive Thinking*





## Environment-Independence



## SAPIENS Customers



## *Moving to the Next Generation*

### *ELVIA Life (Geneva, Switzerland)*

#### **Collective Life Insurance Application**

Packaged Solution	None Suitable
Code Generator	150 Work Years
SAPIENS	50 Work Years

*"With SAPIENS we can develop applications more inexpensively while delivering them earlier."*

-Gottfried Santschi, CIO



## *Conversion and Migration*

### *Canadian Imperial Bank of Commerce (UK)*

#### **Mortgage-Processing Application**

Packaged Solution	None Suitable
Code Generator	50 Work Years
SAPIENS	22 Work Years

*"SAPIENS has proved to be totally portable. The conversion to MVS took just three or four weeks, whereas this type of work can be a six-month project."*

-John Owens, Senior Project Manager

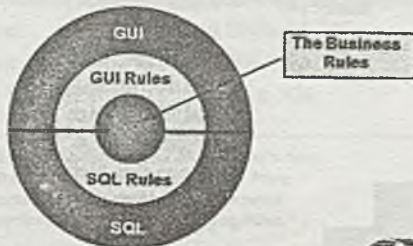


*SAPIENS Ideo provides  
scaleable, enterprise-wide  
rapid application development  
solutions for client/server systems*



## *Coding Requirements for a Client/Server Application*

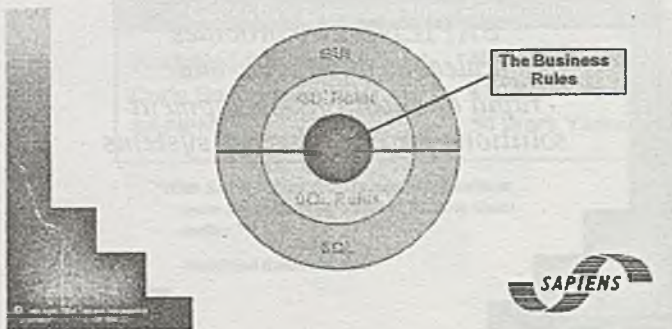
- Coding required in five areas
- The business benefits from only one area





## Ideo's Rule-Based Approach

- Write only the rules that define the unique business application



## SAPIENS Ideo - Cross Platform

### GUI

- Motif
- Windows
- Macintosh

### Platform

- VAX OpenVMS
- Alpha AXP OpenVMS
- Alpha AXP OSF/1
- Digital RISC Ultrix
- Sun SPARC SunOS
- Sun SPARC Solaris
- IBM RS/6000 AIX
- HP 9000/700 UX
- Silicon Graphics IRIX
- PC Windows 3.1
- PC Windows NT
- Macintosh (deploy only)

### Database

- Ideo RDBMS
- Rdb
- Oracle
- Sybase
- Ingres
- Informix



## SAPIENS Ideo Customers



### *An Order of Magnitude Improvement*

#### *Midland Brake (St. Louis, Mo.)*

##### ISO 9000 Certification Application

Server	Digital VAX
Clients	300 PCs
Ideo Productivity	10 x 3GL

*"Ideo is laying the ground for Midland's new strategic systems. We could not have come so far, so fast without its rapid application development capabilities."*

-Donovan Dean, MIS Director



## *A Declaration of Database Independence*

**DCSE Consulting (Irvine, California)**

### **Water Management Applications**

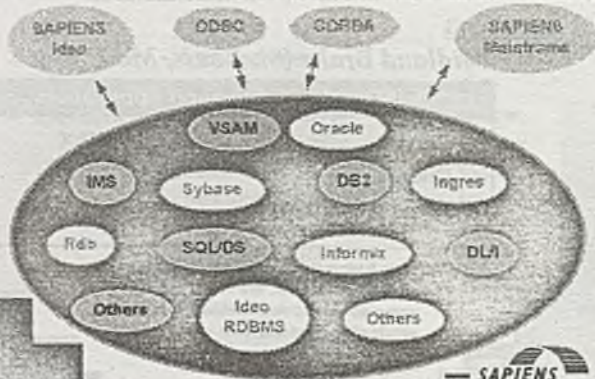
**Must Support Popular RDBMSs  
Platforms: PC, UNIX, VAX/VMS  
3GL Integration**

*"With Ideo tools, generally we can build our application on top of whatever platforms and databases the client already has."*

-All Diba, President



## *The SAPIENS ObjectPool*

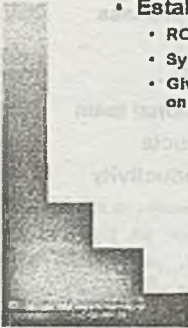




## *Integrating Systems with Sapiens*

### **Intercontainer-Interfrigo (ICF) s.c.**

- Established SAPIENS mainframe customer:
  - ROI = 2.4 years
  - System handles 23,000 invoices worth \$400 million per year
  - Gives ICF the unique capability to provide exact quotations on intermodal shipments



## *Integrating Systems with Sapiens*

### **Intercontainer-Interfrigo (ICF) s.c.**

- Moving to UNIX with SAPIENS Ideo:
  - Decentralized system brings more power and flexibility to regional operations throughout Europe
  - ObjectPool + Ideo protects legacy investment, while supporting client/server development
  - Comment: "You might say we're on the Sapiens 'Highway to Client/Server'... We are very happy with SAPIENS, and with the move to UNIX, we will be even happier."
  - René Christa, Corporate MIS Director



## *RADical Software Development*

...is a customer driven application development lifecycle that:

- employs an evolutionary process
- in a time-boxed fashion
- to deliver quality solutions
- from a dedicated professional team
- enabled by Sapiens products
- resulting in profound productivity benefits





COMPUWARE.

POLNET TECHNOLOGIES  
INTERNATIONAL

UNIFACE Six

## Informacje ogólne

*UNIFACE Six to odpowiedź firmy COMPUWARE na jeden z największych problemów, z jakimi borykają się współczesne przedsiębiorstwa - gwałtowny rozwój technologii. Wielość standardów, częstość zmian, szybkie starzenie się „moralne” rozwiązań, powoduje trudności w zarządzaniu, zwiększając ryzyko nietrafnego wyboru. UNIFACE Six należy do narzędzi uniwersalnych, racjonalnie wykorzystujących istniejące systemy, a przez to obniżających rzeczywiste koszty poniesione na technologię.*

Podstawowe zalety i unikalność UNIFACE Six polegają na:

- *zorientowaniu na potrzeby przedsiębiorstwa - (jest to narzędzie do budowy bizneso-centricznych aplikacji Klient/Serwer drugiej generacji). Model biznesowy (funkcjonalny i informacyjny) przedsiębiorstwa stanowi tu jądro koncepcyjnego systemu informatycznego;*
- *niezależności od infrastruktury technologicznej: systemów operacyjnych, sieciowych, bazodanowych i sprzętu;*
- *skalowalności umożliwiającej zaspokojenie potrzeby rozwoju każdej, nawet dużej firmy;*
- *łatwej adaptacji aplikacji do zmieniających się wymagań biznesowych i dynamicznie ewoluującej infrastruktury technologicznej;*
- *możliwości pracy w środowisku graficznym i tekstowym;*
- *pewności sukcesu - UNIFACE Six to system sprawdzony przez przeszło 8000 firm na całym świecie, skupionych w aktywnie działającym klubie użytkowników. Jednym z ostatnich użytkowników systemu UNIFACE jest fabryka samochodów Skoda w Czechach. Po zakupieniu zakładów przez firmę Volkswagen, w ramach unowocześnienia funkcjonowania zakładu, zbudowano w środowisku UNIFACE Six dedykowany system zarządzania. System ten świetnie zdał egzamin w praktyce;*



- *wysokiej jakości, której potwierdzeniem były wyniki konkursu organizowanego przez jedno z bardziej znanych pism poświęconych technologii informatycznej - CM Corporate. Organizowany corocznie konkurs sprawności różnych narzędzi informatycznych ma bardzo prestiżowy charakter. Uczestniczą w nim producenci oprogramowania o światowej renomie. W 1995 roku konkurs dotyczył narzędzi do szybkiego projektowania i rozwoju złożonych aplikacji Klient/Serwer. W konkursie wzięło udział 14 wiodących światowych firm - producentów narzędzi Klient/Serwer. Reprezentowane tu były produkty takich firm, jak: IBM, Microsoft, Borland, PowerSoft, Gupta, Magic itp. Zwyciężył UNIFACE Six, uznany za narzędzie najbardziej elastyczne i umożliwiające bardzo sprawne zbudowanie złożonej aplikacji;*
- *gwarancji rozwoju i trwałości - UNIFACE Six to produkt amerykańskiej firmy Compuware Corporation. Firmę założono w 1973 roku. Główna siedziba mieści się w Farmington Hills w stanie Michigan. Zatrudnia ona około 4000 osób w 38 krajach, osiągając obroty przekraczające 500 milionów dolarów rocznie. Compuware należy do światowej czołówki firm informatycznych. Jest jednym z najpoważniejszych producentów oprogramowania oraz firmą świadczącą wyspecjalizowane usługi informatyczne. Usługi te obejmują szerokie spektrum sprzętowo-programowe od mainframe do Klient/Serwer i dotyczą: planów strategicznych, konsultingu, projektowania, wdrażania, testowania i zarządzania wielkimi aplikacjami o zasięgu ogólnoswiatowym. W dziedzinie produkcji oprogramowania celem strategicznym Compuware jest obejmowanie całego cyklu życia produktu. Stąd oferuje trzy podstawowe grupy produktów: produkty do projektowania i budowy aplikacji (UNIFACE), produkty do testowania i wdrażania (PLAYBACK, File-AID, DBA-XPert, Abend-AID, XPEDITER, DATA-XPert, Hiperstation) oraz produkty do zarządzania aplikacjami (EcoTOOLS), RemoteControl/2). Wszystkie produkty Compuware należą do czołówki światowego oprogramowania i cieszą się bardzo dobrą opinią użytkowników;*
- *gwarancji profesjonalnej opieki serwisowej, szkolenia, hot-line - generalnym dystrybutorem produktów firmy Compuware w Polsce jest POLNET Technologies - firma działająca na rynku polskim od 1990 roku. POLNET jest firmą wyspecjalizowaną w świadczeniu zaawansowanych usług informatycznych i konsultingowych. Strategia działania POLNET zakłada: uniezależnienie klienta od dostawców sprzętu i oprogramowania, indywidualne podejście do każdego przedsięwzięcia, współudział klienta w prowadzonych pracach oraz przekazywanie klientom najnowocześniejszych technologii. Ta strategia predysponuje POLNET do roli dystrybutora zaawansowanych produktów informatycznych.*



**POLNET TECHNOLOGIES**  
INTERNATIONAL

COMPUWARE.



## **Koncepcja środowiska projektowego Klient/Serwer drugiej generacji**

**UNIFACE Six - jest to zintegrowane środowisko projektowe drugiej generacji służące do szybkiego rozwoju i konstruowania skalowalnych i technologicznie niezależnych systemów informatycznych o architekturze Klient/Serwer, wykorzystujące w procesie projektowym i budowaniu systemów metodologię „sterowania modelem”.**

System informatyczny buduje się w oparciu o model funkcjonalny i informacyjny przedsiębiorstwa. Model przedsiębiorstwa stanowi więc jądro koncepcyjne systemu informatycznego. Model ten - to formalna reprezentacja procesów przetwarzania informacji w przedsiębiorstwie oraz reguł jego działania. UNIFACE Six dostarcza projektantom szereg narzędzi do definiowania, modyfikowania, przechowywania i zarządzania modelem przedsiębiorstwa. Model jest przechowywany w zintegrowanym środowisku zwanym „Repozytorium”. W oparciu o Repozytorium, projektanci budują systemy informatyczne.

UNIFACE Six to także metodyka projektowania. Zespół narzędzi UNIFACE Six i zastosowana metodyka pozwalają na dużą produktywność procesu projektowania oraz skalowalność i zdolność adaptacji produktów (aplikacji) do zmieniających się wymagań biznesowych i dynamicznie ewoluującej infrastruktury technologicznej.

UNIFACE Six pozwala na wdrożenie i instalację, zbudowanych przy jego pomocy, aplikacji, w sposób niezależny od infrastruktury technologicznej.



Narzędzia do konstruowania systemów informatycznych o architekturze Klient/Serwer ewoluują bardzo szybko. Wraz z rosnącym zrozumieniem roli informatyzacji systemy informatyczne zwiększają swój zasięg w skali przedsiębiorstwa, a ich złożoność i zakres różnorodności obiektów informacyjnych i usług stale rosną. Stawia to nowe wyzwania wobec produktów wspomagających proces projektowania i konstruowania takich systemów.

Nowa technologia narzędzi projektowych systemów Klient/Serwer, zwana technologią drugiej generacji, musi dostarczać narzędzi znacznie bardziej elastycznych w porównaniu do narzędzi dotychczas stosowanych, a także pozwalających na tworzenie systemów bardziej efektywnych z punktu widzenia kosztów ich wytwarzania i kosztów ich eksploatacji. Jako najbardziej pożądane cechy systemów Klient/Serwer drugiej generacji można wymienić następujące:

- zdolność do swobodnego przesuwania składników programowych aplikacji między Serwerem i stacją Klienta;
- skalowalność w kierunku złożonych i bogatych aplikacji o zasięgu całej firmy;
- zdolność do partycjonowania, czyli elastycznego podziału aplikacji między wiele serwerów wykonujących programy aplikacji, realizujących obsługę i dostęp do systemów bazodanowych o złożonej topologii sieci i umieszczonych na wielu platformach sprzętowych i operacyjnych;
- spełnianie wysokich wymagań bezpieczeństwa danych - szczególnie w systemach sieciowych i systemach korzystających z usług telekomunikacyjnych;
- spełnianie wysokich wymagań testowalności i zarządzania;
- wykorzystanie możliwości jakie daje technologia orientacji obiektowej.

Według oceny wielu niezależnych firm analitycznych takich, jak: Gartners Group, Butlers Group, Hurwitz Consulting Group czy Patricia Seybold Group - UNIFACE Six należy do ścisłej czołówki narzędzi do projektowania i rozwoju systemów Klient/Serwer drugiej generacji.

Poniżej przedstawione są podstawowe cechy UNIFACE Six wyróżniające go wśród innych narzędzi Klient/Serwer - większość narzędzi tego typu nie posiada równocześnie wszystkich cech tu wymienionych:

1. **Uproszczenie procesu projektowego poprzez rozwój projektu aplikacji sterowany modelem:**
  - Rozwój aplikacji sterowany modelem jest podstawową metodyczną cechą UNIFACE Six.
  - Model umożliwia definiowanie systemu informatycznego na wysokim poziomie abstrakcji, uwalniając projektanta od konieczności posługiwania się szczegółami fizycznej implementacji w docelowym środowisku technologicznym.



- *Model Aplikacji zawiera model danych, reguły biznesu, zasady przetwarzania, wzorce i obiekty globalne.*
- *Model Danych zawiera wszystkie elementy stanowiące odzwierciedlenie struktury informacyjnej przedsiębiorstwa: encje (obiekty informacyjne), pola (składniki obiektów informacyjnych), związki logiczne między danymi oraz więzy integralności danych.*
- *Model Aplikacji definiuje uniwersalne, niezależne od fizycznej reprezentacji, zasady dostępu do danych, logiczne zarządzanie transakcjami oraz uniwersalne zasady obsługi sytuacji wyjątkowych i błędów.*
- *Model Aplikacji zawiera także, zintegrowane z modelem danych, elementy przetwarzania: reguły biznesu, zasady kontroli poprawności danych i funkcje obliczeniowe związane z encjami, polami danych itp.*
- *Wzorce (templates) definiują złożone, często używane obiekty (dane i elementy przetwarzania). Wzorce mogą zawierać dodatkowo różne atrybuty związane z prezentacją danych (np. kolor, krój czcionki, kształty przycisków, rozwijanych list itp.), które są „budulcem” dla konstrukcji takich składowych aplikacji, jak: formaty ekranowe i elementy raportów.*
- *Wzorce oferują, w bardzo praktycznym sensie, zdolność do wielokrotnego wykorzystywania elementów modelu, dziedziczenia i korzyści z abstrakcyjnego charakteru modelu, czyli tych własności, które stanowią cechy technologii orientacji obiektowej (Object Oriented Technology).*

**2. Repozytorium Obiektów Aplikacji, jako centrum przechowywania i zarządzania zasobami projektowymi:**

- *Sterowany modelem proces projektowy jest wspomagany przez użycie Repozytorium Obiektów Aplikacji.*
- *Repozytorium Obiektów Aplikacji jest to miejsce, w którym przechowywane są wszystkie informacje oraz obiekty wykorzystywane w procesie rozwoju aplikacji. W szczególności w Repozytorium zawarty jest pełny Model Aplikacji oraz inne informacje o obiektach będących komponentami aplikacji takimi, jak: formy ekranowe, raporty, menu, procedury itp.*
- *Repozytorium Obiektów Aplikacji jest bardziej złożone niż zwykła baza danych czy Słownik Danych, będące typowymi składnikami innych narzędzi do konstruowania systemów informatycznych. W Repozytorium uchwycona jest cała semantyczna strona aplikacji. Abstrakcyjne obiekty Modelu Aplikacji zawarte w Repozytorium są automatycznie przyłączane do interfejsu użytkownika i odpowiednich źródeł danych, uwalniając projektanta od specyficznego dla danej platformy technologicznej kodowania. Własności obiektów zdefiniowanych w Repozytorium są automatycznie dziedziczone przez następne aplikacje, formaty ekranowe itp.*

- Repozytorium posiada otwartą i udokumentowaną architekturę. Dzięki temu Repozytorium może być implementowane w różnych systemach bazodanowych, czerpiąc korzyści z wszystkich ich cech takich, jak: kontrola integralności, szybkie wyszukiwanie lub automatyczne kopie bezpieczeństwa. Dzięki swojej otwartej architekturze Repozytorium może być również utworzone w plikach znakowych - obniża to koszty konfiguracji dla stacji pojedynczego projektanta.
  - Stosowany w UNIFACE Six, sterowany modelem sposób prowadzenia projektu, pozwala uzupełniać wyniki analizy i projektowania wykonane przy użyciu narzędzi CASE. Model może być importowany i synchronizowany z modelami tworzonymi przy pomocy narzędzi CASE.
3. **Obiektowo zorientowane graficzne środowisko do szybkiego rozwoju aplikacji:**
- Graficzne środowisko do interfejsu z użytkownikiem jest stosowane w większości narzędzi projektowych systemów Klient/Serwer. W UNIFACE Six wszystkie obiekty zdefiniowane w Repozytorium mogą być użyte w graficznym projektowaniu aplikacji. Ponieważ Model zawiera elementy przetwarzania, mogą one być włączone do aplikacji, redukując potrzebę pisania kodu.
  - Uniwersalny interfejs prezentacyjny umożliwia projektantom konstruowanie aplikacji całkowicie niezależnie od docelowego fizycznego graficznego środowiska prezentacyjnego (np. Windows 3.x, Windows 95, OS/2, UNIX Motif, Macintosh). Ponieważ ogromna część aplikacji używa interfejsu znakowego lub terminali znakowych, uniwersalny interfejs prezentacyjny UNIFACE Six automatycznie dokonuje odwzorowania graficznych formatów ekranowych i wszystkich graficznych elementów kontrolnych do reprezentacji ekranów znakowych. Cecha ta powoduje, że aplikacje są całkowicie przenoszalne i w większości przypadków nie wymagają przeróbek przy przechodzeniu ze środowiska graficznego do znakowego.
4. **Technologicznie niezależne wdrażanie aplikacji:**
- Aplikacje zaprojektowane i testowane w środowisku projektowym są wdrażane w docelowym środowisku operacyjnym. Zarządzanie dystrybucją i wdrożeniem pozwala na dokonanie instalacji tej samej aplikacji (bez modyfikacji) na różnych platformach i dla różnych architektur takich, jak: host i terminale, architektura dwu i trzypoziomowa Klient/Serwer oraz inne, bardziej skomplikowane, konfiguracje przetwarzania rozproszonego.
  - Fizyczny dostęp do baz danych jest realizowany za pośrednictwem drajwerów specyficznych systemów bazodanowych, uwzględniając także szczególne cechy różnych wersji. Drajwery baz danych stanowią interfejs między uniwersalnymi operacjami dostępu i manipulacji danymi używanymi po stronie aplikacji UNIFACE Six i właściwym repertuarem operacji danego systemu bazodanowego (np. dla systemów zarządzania relacyjnymi



bazami danych drajwer generuje kod SQL, uwzględniający szczególne elementy dialektu danego systemu).

- Odpowiednia kombinacja drajwerów umożliwia równoczesny dostęp do różnych baz danych w sposób całkowicie przezroczysty dla aplikacji. Mimo tej automatyzacji projektant ma możliwość zarządzania stopniem automatyzacji obsługi fizycznego dostępu do danych.
- Optymalizacja operacji na danych odbywa się w drajwerach. Jeśli wersja bazy danych dostarcza mechanizmów wewnętrznych do kontroli integralności, umożliwia definiowanie trygerów i procedur zapamiętywanych w bazie, to drajwer generuje odpowiednie instrukcje dla bazy danych. W przypadku braku tych mechanizmów odpowiednie operacje są realizowane przez UNIFACE Six w aplikacji. Ta cecha UNIFACE Six powoduje, że duże firmy produkujące systemy bazodanowe takie, jak: Oracle czy Informix, kiedy kolejne wersje produktów są niekompatybilne ze sobą, proponują swoim klientom rozwiązanie problemu niekompatybilności przez użycie UNIFACE.
- Zarządzanie instalacją umożliwia dokonanie podziału i dystrybucji komponentów aplikacji (obiektów) pomiędzy wieloma serverami w sieci (partycjonowanie aplikacji). Ponadto Zarządzanie Przetwarzaniem Rozproszonym pozwala konfigurować aplikacje w środowisku wykorzystującym monitory rozproszonych transakcji (np. Tuxedo, Encina CICS) oraz otwarte protokoły przetwarzania rozproszonego takie, jak: DCE (Distributed Computing Environment).
- Jednym z wielkich problemów narzędzi Klient/Server jest słaba ochrona danych. W aplikacjach UNIFACE Six każda operacja pobrania lub zapamiętania danych może być połączona z szyfrowaniem i deszyfrowaniem danych.

##### 5. Projektowanie zespołowe:

- Repozytorium jest wielodostępne. Wielu projektantów aplikacji może bezkonfliktowo pracować równocześnie.
- Repozytorium posiada otwartą i w pełni udokumentowaną strukturę - może być zaimplementowane w dowolnym środowisku bazodanowym (nawet na wielu różnych systemach bazodanowych równocześnie). Cecha ta umożliwia wykorzystanie usług dostarczanych przez system bazodanowy takich, jak: kontrola integralności, automatyczne kopie bezpieczeństwa oraz replikacje.
- Obsługa Zespołu Projektowego dostarcza systemu kontroli wersji. Dzięki temu łatwo jest kontrolować i prowadzić projekty wielowariantowe. Możliwa jest kontrola wersji na poziomie pojedynczych obiektów modelu.



- Kontrola dostępu odbywa się na dwóch poziomach: na poziomie kontroli dostępu do poszczególnych funkcji UNIFACE Six oraz na poziomie elementów składowych modelu. Ta ostatnia cecha wspomaga zachowanie standardów rozwiązań dla wielu aplikacji powstających w ramach przedsiębiorstwa.
6. *Potencjał producenta UNIFACE Six gwarantuje poziom usług serwisowych i stały rozwój produktu:*
- Producent UNIFACE Six - firma Compuware, w rankingach niezależnych producentów oprogramowania w Stanach Zjednoczonych plasuje się w pierwszej dziesiątce najbardziej znaczących na rynku. Jej przychód w roku 1995 przekroczył znacznie 500 mln USD.
  - POLNET Technologies, jako generalny dystrybutor firmy Compuware w Polsce, zapewnia wszechstronny serwis techniczny, szkolenia, pomoc w instalacji itp. Klientom, z którymi zawarta jest umowa pielęgnacyjna, dostarcza regularnie nowe wersje produktu i innych elementów (np. drajwery dla baz danych są aktualizowane wraz z pojawianiem się nowych wersji systemów bazodanowych).
  - Pod patronatem Compuware odbywają się coroczne konferencje użytkowników UNIFACE Six oraz producentów aplikacji wykonanych przy jego pomocy oraz oprogramowania wspomagającego.
  - Na świecie ukazuje się kilka niezależnych czasopism i innych wydawnictw poświęconych problematyce UNIFACE Six.

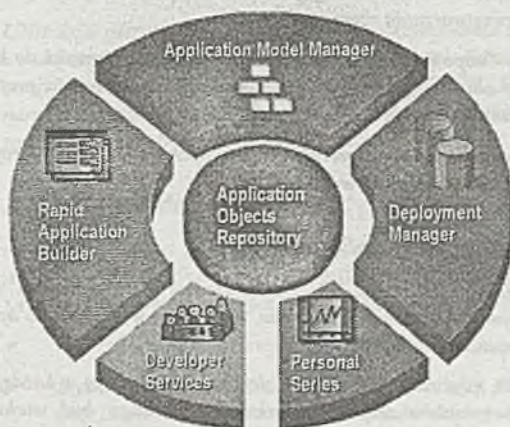


COMPUWARE.

**POLNET TECHNOLOGIES**  
INTERNATIONAL

**UNIFACE** **SIX**

## ***Sterowane modelem środowisko do tworzenia aplikacji Klient/Serwer***



- Wydajne Projektowanie, dzięki czerpaniu informacji z jednego modelu
- Unikalne możliwości zarządzania transakcjami
- Wydajna pielęgnacja, zarządzanie zmianami poprzez model w repozytorium
- Szybka i tania dystrybucja aplikacji na różne platformy, wiele DBMS i rodzajów interfejsów użytkownika
- Spójność i pełna kontrola projektu w pracy zespołowej
- Możliwość rozwijania aplikacji, przez wzrost oraz partycjonowanie w różnych konfiguracjach sieciowych



UNIFACE Six jest środowiskiem do tworzenia systemów Klient/Serwer drugiej generacji. Podstawowymi cechami tego środowiska są:

- Sterowana modelem budowa aplikacji.
- Obiektowe środowisko graficzne.
- Tworzenie aplikacji niezależnych od technologii, otwartych i skalowalnych.
- Partycjonowanie aplikacji i przetwarzanie rozproszone.
- Praca zespołowa zapewniająca wysoką wydajność w dużych projektach.

Architektura UNIFACE'a, oparta na obiektach, umożliwia szybkie tworzenie aplikacji Klient/Serwer i znacznie upraszcza budowę dużych aplikacji. Jądem UNIFACE'a jest repozytorium obiektów aplikacji. Repozytorium jest wspólne dla wszystkich narzędzi wchodzących w skład UNIFACE Six:

- **Application Model Manager** - umożliwia graficzne projektowanie aplikacji i łatwą pielęgnację definicji danych i funkcji aplikacji oraz obiektów globalnych.
- **Rapid Application Builder** - jest narzędziem do tworzenia aplikacji. Umożliwia szybkie tworzenie form ekranowych i raportów w środowisku graficznym.
- **Deployment Manager** - umożliwia konfigurację aplikacji dla różnorodnych platform sprzętowych, baz danych, interfejsów użytkownika oraz partycjonowanie aplikacji i obsługę przetwarzania rozproszonego.
- **Developer Services** - wspomaga pracę zespołową. Dostarcza narzędzi do kontroli wersji na poziomie obiektów, nadawania uprawnień i definiowania preferencji poszczególnych członków zespołu.
- **Personal Series** - umożliwia łatwe wyszukiwanie danych oraz tworzenie raportów i wykresów przez użytkownika końcowego.

### **Sterowany modelem rozwój aplikacji.**

Budowa aplikacji w UNIFACE opiera się na centralnie zdefiniowanym modelu aplikacji, przechowywanym w repozytorium systemu. Budowane w ten sposób aplikacje są elastyczne, otwarte, przenośne i skalowalne.

Aplikacje powstają szybko, gdyż raz zdefiniowane elementy aplikacji są w każdej chwili dostępne dla wszystkich członków zespołu projektowego i mogą być wielokrotnie wykorzystane w różnych sytuacjach. Każda zmiana obiektu w repozytorium jest od razu propagowana do wszystkich tych miejsc, w których obiekt został użyty.

Aplikacje można łatwo dostosować do zmieniających się potrzeb użytkownika. Wymaga to tylko zmiany w modelu, a nie w wielu miejscach w kodzie aplikacji.

Zmiany w modelu są automatycznie odzwierciedlane we wszystkich właściwych formach ekranowych i raportach.



## Repozytorium obiektów aplikacji.

Wszystkie obiekty tworzonej przy pomocy UNIFACE Six aplikacji są przechowywane w repozytorium:

- *model danych*, zawierający: encje, pola, związki, warunki integralności, własności prezentacyjne obiektów,
- *procesy (methods)* - reguły biznesu, logika aplikacji,
- *wzorce (templates)* dla obiektów takich, jak: encje, pola, elementy prezentacji, kryteria poprawności,
- *obiekty globalne* takie, jak: menu, paski ikon, procedury, komunikaty.

Repozytorium UNIFACE Six jest:

- *otwarte i wielodostępne* (może być umieszczone w wybranej przez projektanta bazie danych),
- *udokumentowane* (można generować raporty zawierające informacje o przechowywanych w repozytorium obiektach aplikacji),
- *zintegrowane z narzędziami CASE*,
- *ściśle związane ze wszystkimi fazami cyklu życia systemu*.

UNIFACE oferuje jako opcję możliwość utrzymywania zgodności modelu edytowanego wewnątrz (przy pomocy Application Model Editor) z modelami tworzonymi i modyfikowanymi przez zewnętrzne narzędzia CASE. Model Synchronizer jest grupą narzędzi stanowiących interfejs do różnych pakietów CASE i równocześnie zapewniających dwukierunkową aktualizację modelu.

## Application Model Manager.

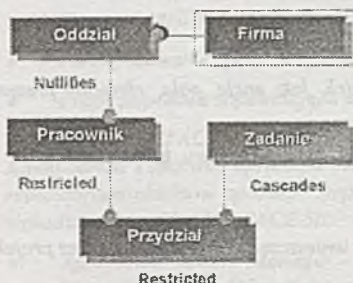
Application Model Manager (AMM) umożliwia tworzenie modeli i zarządzanie nimi. AMM zawiera zarówno interfejs graficzny, jak i znakowy. Application Model Editor (AME) jest interfejsem graficznym do AMM. AME umożliwia:

- *tworzenie diagramu encji i związków (ERD)*,
- *szybkie i łatwe tworzenie obiektów graficznych (takich, jak: przyciski, rozwijane listy i inne rodzaje ekranowych elementów sterujących)*,
- *definiowanie zachowania obiektów (np. trygery)*,
- *definiowanie własności obiektów (takich, jak: czcionki, kolory, efekt 3D)*.

Obiekty zdefiniowane w modelu aplikacji mogą być wielokrotnie użyte podczas tworzenia form ekranowych i raportów, co zwiększa wydajność i zapewnia spójny wygląd i obsługę aplikacji.

### Modelowanie Danych

- Encje
- Pola
- Związki



### Repozytorium Obiektów

- Wieży Integralności
- Przetwarzanie Zdarzeń
  - "Wyjdź z Pola" "Szczegół"
  - "Wykonaj" "Zapamiętaj"
- Centralne Procedury Obsługi Błędów
- Reguły
  - "If (koszt > 5000) run" "approve"
- Reguły Poprawności
- Reguły Formatowania
- Obliczenia. Inne typy przetwarzania

UNIFACE Six - Przykładowy model aplikacji

### Rapid Application Builder.

UNIFACE daje możliwość budowy aplikacji z wykorzystaniem cech podejścia obiektowego: abstrakcji, dziedziczenia, polimorfizmu i hermetyzacji obiektów. Aplikacje (formy ekranowe, raporty i inne obiekty) tworzone są zazwyczaj w środowisku graficznym. Możliwość zastosowania graficznej operacji drag-and-drop, która dokonuje powiązania danych biorących udział w operacji, znacznie ułatwia i przyspiesza tworzenie aplikacji.

Zachowanie obiektów jest definiowane w języku czwartej generacji PROC. Z każdym obiektem jest związany wiele procedur, zwanych trygerami. Są one automatycznie generowane w oparciu o typ obiektu i jego definicję w modelu danych. W przypadkach bardziej skomplikowanych projektant ma możliwość modyfikowania istniejących lub pisania własnych trygerów. Ponadto z procedur w języku PROC można wywoływać programy pisane w językach 3GL.

Rapid Application Builder (RAB) zawiera:

- Edytor Graficzny (GFP - Graphical Form Painter).
- Uniwersalny Interfejs Warstwy Prezentacji (UPI - Universal Presentation Interface).
- Standardowe ekranowe elementy sterujące.
- Debugger (testowanie aplikacji instrukcja po instrukcji, badanie wartości zmiennych, ustalanie punktów przerwań, sprawdzanie we/wy aplikacji).

Przy pomocy Edytora Graficznego (GFP), wyposażonego w paletę obiektów graficznych i interfejs „point-and-click”, można szybko i łatwo budować formy ekranowe i raporty. FastForm, będący elementem składowym GFP, umożliwia wielokrotne użycie w formach definicji modelu wyspecyfikowanych w Application Model Manager.



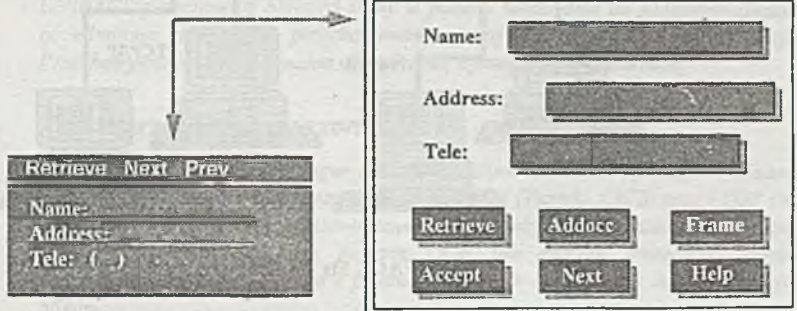
*FastForm*, będący elementem składowym *GFP*, umożliwia wielokrotne użycie w formach definicji modelu wyspecyfikowanych w *Application Model Manager*.

*GFP* daje możliwość szybkiego dostępu do wszystkich własności obiektów formy (w tym do trygerów), poprzez jedno tylko „kliknięcie myszą”. Struktura formy ekranowej wyświetlana jest w oknie umożliwiającym łatwe wyszukiwanie i wybieranie obiektów i ich własności.

Bardzo ważną własnością *GFP* jest konstrukcja zagnieżdżonych ramek (*frames-within-frames*), przydatna w budowaniu form ekranowych zawierających powiązane ze sobą encje. Dzięki tej konstrukcji, zbudowanie formy do obsługi powiązanych ze sobą encji nie wymaga pisania kodu obsługi wejścia/wyjścia z bazy danych, zarządzania transakcjami, obsługi błędów, nawet jeśli forma zawiera wiele powiązanych ze sobą encji.

Uniwersalny Interfejs Warstwy Prezentacji (*UPI*) zapewnia przenośność aplikacji pomiędzy różnymi środowiskami graficznymi i znakowymi, bez jakichkolwiek zmian w kodzie aplikacji. *UPI* zawiera *Open Widget Interface (OWI)*, który jest mechanizmem umożliwiającym włączanie standardowych (standard wewnętrzny *UNIFACE'a*, *VBX*, *SOM*, ...) i zdefiniowanych przez projektanta elementów sterujących ekranu do palety obiektów *UNIFACE'a*.

Ten sam kod dla  
środowiska znakowego  
i graficznego



UNIFACE Six - Uniwersalny interfejs użytkownika

**Deployment Manager** - technologicznie niezależne wdrożenie aplikacji.

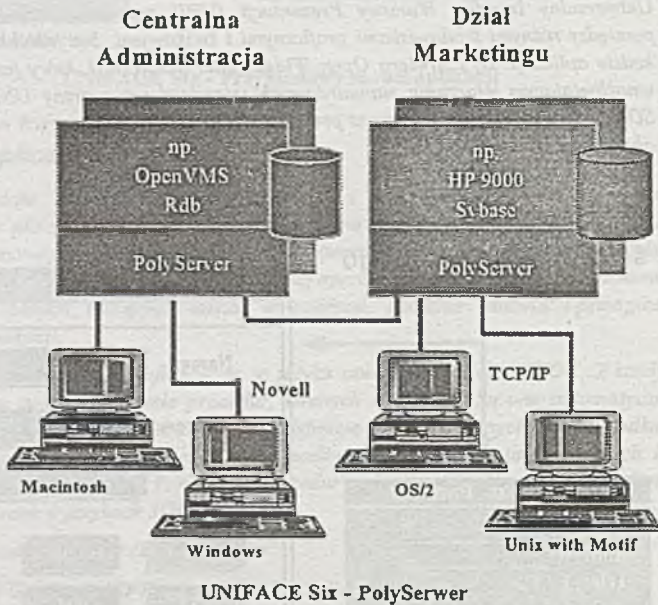
Zbudowanie dobrej aplikacji to jeszcze nie wszystko. Stale zmieniająca się architektura technologii oraz oczekiwania co do wydajności aplikacji skalowalnych stawiają nowe wymagania przed projektantami systemu informatycznego w przedsiębiorstwie.



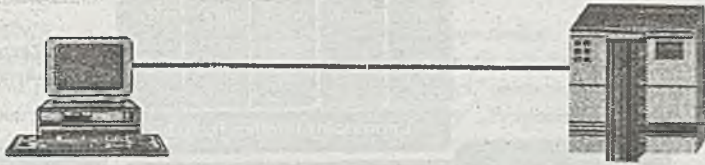
*Deployment Manager umożliwia wdrożenie aplikacji na wielu różnorodnych platformach sprzętowo-programowych, zarówno dla konfiguracji host-terminale, jak i Klient/Serwer, a w szczególności umożliwia wielowarstwowe partycjonowanie aplikacji. Do obsługi aplikacji w sieci UNIFACE wykorzystuje specjalny moduł PolyServer.*

### **Partycjonowanie aplikacji.**

*Celem partycjonowania aplikacji jest zapewnienie takiej dystrybucji przetwarzania i obiektów pomiędzy komputerami w sieci, aby aplikacje wielkiej skali, obsługujące dużą liczbę transakcji i użytkowników, mogły sprawnie, szybko i wydajnie działać. UNIFACE dostarcza narzędzi do elastycznego podziału aplikacji, dając możliwość konfiguracji z „grubym” lub „chudym” klientem, w zależności od potrzeb.*



*Komponenty aplikacji takie, jak: formy ekranowe, obrazy, procedury, komunikaty, można przy pomocy PolyServera umieścić w różnych miejscach w sieci. Daje to możliwość dynamicznego przydzielania do różnych CPU zadań takich, jak przetwarzanie wsadowe czy raportowanie.*



**Deployment Manager :**  
**Klient**

- Prezentacja
- Logika zdarzeń
- Obsługa błędów
- Przetwarzanie lokalne
- Sterowanie transakcjami

**Deployment Manager :**  
**PolyServer**

- Dynamiczne Operacje We/Wy
- Konwersje Protokołów
- Generacje Zapytań
- Obsługa Błędów
- Sterowanie Transakcjami
- Stored Procedures
- Remote Procedures
- Biblioteki Obiektów Dynamicznych
- Formy na Serwerze
- Przetwarzanie Wsadowe
- Raportowanie Wsadowe
- Kontrola Integralności

### **UNIFACE Six - Partycjonowanie (Deployment Manager)**

Aby zminimalizować ruch w sieci, UNIFACE dostarcza dynamiczną bibliotekę obiektów - Dynamic Object Library (DOL) wraz ze „skrytką na obiekty” (object cache). Obiekty są pobierane z biblioteki tylko raz i umieszczane w skrytce.

UNIFACE przechowuje obrazy i ikony w postaci niezależnej od platformy sprzętowej, co eliminuje konieczność przechowywania wersji obrazów zależnych od platformy. Przechowywanie i wyszukiwanie obrazów nie wymaga programowania.

### **Przetwarzanie rozproszone.**

*Distributed Computing Manager (DCM)* umożliwia działanie aplikacji w środowisku rozproszonym i współpracę z monitorami transakcji (Tuxedo, CICS) oraz z OSF DCE w sposób przezroczysty dla użytkownika i bez potrzeby pisania kodu. DCM zapewnia aplikacji dostęp do danych poprzez DCE lub monitor transakcji, wykonując jednocześnie operacje czytania i pisania do prawie każdej bazy danych lub systemu plików.





UNIFACE Six - Deployment Manager

### Dostęp do baz danych.

UNIFACE współpracuje z ponad trzydziestoma systemami zarządzania bazami danych i systemami plików. Aplikacje tworzone w UNIFACE Six są niezależne od baz danych, a dostęp do wybranych baz danych odbywa się poprzez specyficzne dla nich drajwery UNIFACE'a, z uwzględnieniem wersji bazy danych i wykorzystaniem mechanizmów właściwych dla danego systemu bazodanowego. Jeśli system obsługi bazy danych nie zapewnia potrzebnych mechanizmów, np. blokowania, procedur pamiętanych, więzów integralności, UNIFACE przejmuje ich obsługę.

UNIFACE oferuje również możliwość dostępu do baz danych poprzez drajwer do ODBC.

### Praca zespołowa.

System Kontroli Wersji UNIFACE'a:

- umożliwia przechowywanie różnych wersji aplikacji (rozwijane, testowe, eksploatacyjne),
- dostarcza funkcji do przeglądania, pielęgnowania i raportowania zawartości systemu kontroli wersji,
- zapewnia kontrolę wersji na poziomie obiektów repozytorium i pozwala na blokowanie obiektu na czas jego aktualizacji przez członka zespołu projektowego (mechanizm check-in i check-out).

UNIFACE zawiera otwarty i udokumentowany interfejs, co pozwala na zastosowanie dowolnego, wybranego przez projektanta systemu kontroli wersji, zamiast wewnętrznego systemu UNIFACE'a.



UNIF 6 umożliwia również nadawanie uprawnień dla członków zespołu projektowego. Prawa dostępu mogą być nadawane dla pojedynczych osób lub dla grup. Na przykład, ktoś należący do grupy „Projektanci” może mieć dostęp do wszystkich narzędzi UNIFACE’a, a ktoś należący do grupy „Programiści”, może mieć prawo dostępu tylko do Rapid Application Builder.

Prawa dostępu mogą być nadawane na poziomie narzędzi oraz na poziomie trygierów obiektów, co daje pewność, że globalnie zdefiniowane własności obiektów są przestrzegane w całej aplikacji.

Elementem zwiększającym wydajność zespołu jest system preferencji. Każda grupa lub pojedynczy członek zespołu wykonawczego może określić swoje preferencje dotyczące środowiska pracy. Na przykład, praca z wybranym systemem kontroli wersji lub wskazanie tych obiektów, które powinny zostać automatycznie wybrane z repozytorium przed rozpoczęciem pracy.

### **Personal Series - narzędzia dla użytkownika końcowego.**

Narzędzia Personal Series dają końcowym użytkownikom łatwy dostęp do danych.

- *Personal Query* - służy do tworzenia raportów ad hoc i generowania nawet bardzo złożonych zapytań do bazy danych przy pomocy interfejsu graficznego. Nie jest potrzebna znajomość języka zapytań (SQL) - wystarczy wskazać i wybrać (point-and-click).
- *Personal Access* - umożliwia przeformatowanie wyszukanych danych do postaci akceptowanej przez inne narzędzia (Microsoft Word, Excel, WordPerfect, dBASE, ASCII).
- *Business Graphics* - jest narzędziem do tworzenia różnego rodzaju wykresów (trójwymiarowe kołowe, liniowe, słupkowe, itd.).

### **Podsumowanie.**

Technologia Klient/Serwer ewoluuje bardzo dynamicznie. Przetwarzanie rozproszone, heterogeniczne, środowiska przedsiębiorstw, zwiększający się zakres usług informatycznych, efektywność masowej obsługi transakcji - wszystko to stwarza nowe możliwości, ale stanowi także wielkie wyzwanie dla systemów wspomagających tworzenie aplikacji Klient/Serwer.

W tych warunkach konstrukcja sterowana modelem i technologicznie niezależne wdrożenie stanowiące główne cechy UNIFACE Six, gwarantują sukces w budowaniu nawet bardzo złożonych aplikacji o kluczowym znaczeniu dla biznesu.

**UNIFACE Six - Wybrane platformy sprzętowo-programowe****Systemy operacyjne:**

VMS  
UNIX  
IIP MPE/IX  
MS-Windows  
Macintosh  
OS/2  
DOS

**Bazy danych:**

ADABAS  
ALLBASE/SQL  
BASISplus  
C-ISAM  
DB2/2  
DB2 6000  
dBASE III  
INFORMIX  
INGRES  
Rdb  
RMS  
SYBASE  
ORACLE

**Komunikacja:**

DECnet  
TCP/IP  
LAN Manager  
Novell

**Interfejs użytkownika:**

MS-Windows  
Macintosh  
OSF/Motif  
OS/2 Workplace Shell  
Interfejs znakowy

**Narzędzia CASE:**

SiP - IDE  
System Engineer - LBMS  
System Architect - Popkin  
CDD/R - DEC  
I-SEE - Westmount  
Oracle\*Case  
ADW - KnowledgeWare  
Excelerator - Intersolv  
Design/1 - Andersen Consulting  
IEF - Texas Instruments  
Teamwork - CADRE  
S-Designor - SDP Technology  
SILVERRUN - CSA



COMPUWARE.

**POLNET TECHNOLOGIES**  
**I N T E R N A T I O N A L**



**SOLID<sup>®</sup> Server**  
**dla**

**UNIFACE<sup>SIX</sup>**

Compuware Corporation należy do światowej czołówki firm dostarczających kompleksowe rozwiązania software'owe obejmujące cały cykl życia aplikacji (systemów informatycznych), włączając w to budowanie aplikacji, implementację, pielęgnację i rozwój oraz zarządzanie systemami informatycznymi.

Misją Compuware jest być światowym liderem dostarczającym produkty software'owe wysokiej jakości, zaprojektowane dla zwiększenia produktywności.

UNIFACE Six - produkt Compuware, jest to narzędzie o wysokiej wydajności, pozwalające na tworzenie, wdrażanie i pielęgnację złożonych systemów informatycznych o kluczowym znaczeniu dla przedsiębiorstwa, w sposób niezależny od infrastruktury informatycznej, a w szczególności niezależnych od zastosowanych systemów bazodanowych.

*Obecnie Compuware oferuje - jako uzupełnienie do UNIFACE - SOLID Server dla UNIFACE - SQL'owy system zarządzania bazą danych firmy Solid Information Technology Ltd.*



Wprowadzając produkt SOLID Server dla UNIFACE Compuware może zaoferować swoim klientom kompletne środowisko umożliwiające tworzenie i rozwój aplikacji, w tym także bazę danych służącą do przechowywania danych i stanowiącą repozytorium aplikacji.

Jednocześnie należy podkreślić, że podstawowa cecha UNIFACE - budowanie aplikacji umożliwiających dostęp do większości systemów bazodanowych i systemów plików, pozostaje niezmienną. Ogromna ilość klientów będzie mogła w dalszym ciągu używać systemów bazodanowych najlepszych i największych firm produkujących je, a wspomaganie tego ze strony UNIFACE pozostanie optymalne.

Nie jest bowiem intencją Compuware stać się dostawcą systemów bazodanowych. Celem Compuware jest natomiast dostarczanie klientom kompleksowych rozwiązań dla ich potrzeb. Porozumienie z Solid Information Technology Ltd. jest przykładem takiego kompleksowego rozwiązania.

### ***Co to jest SOLID Server?***

SOLID jest pełnowartościowym systemem zarządzania relacyjną bazą danych. SOLID Server dla UNIFACE jest idealnym komponentem bazodanowym dla masowo produkowanych profesjonalnych aplikacji Klient/Serwer. Posiada większość cech jakie oferuje najnowsza technologia baz danych. Podstawowe cechy produktu SOLID Server dla UNIFACE to:

- Wysoka wydajność i zgodność ze standardami SQL 2 i ODBC.
- Procedury zapamiętywane w bazie, przechowywanie danych typu tekstowego i dużych obiektów binarnych (multimedia) oprócz normalnych typów danych.
- Zarządzanie transakcjami i współbieżnym dostępem.
- Wspomaganie baz rozproszonych - replikacja danych.
- Skalowalność od pojedynczego użytkownika do symetrycznych wieloprocesorowych maszyn RISC.
- Praca na wszystkich podstawowych platformach operacyjnych i protokołach sieciowych.
- Praca 24 godziny na dobę przez 7 dni w tygodniu i automatyczne odtwarzanie bazy danych po awarii.

Ponadto SOLID posiada następujące unikalne cechy:

- Łatwość instalacji i eksploatacji - prawie zerowe wymagania dla administracji bazą danych.
- Bardzo mała konsumpcja zasobów: minimalne wymagania to 1 MB RAM i 2 MB dysku.
- Brak ograniczeń na ilość tablic, wierszy i pól w bazie danych.
- Rewelacyjnie niska cena - od kilkudziesięciu USD na jednego użytkownika w przypadku systemów jednostanowiskowych w środowisku Windows, do kilkuset USD na użytkownika, w przypadku wieloprocesorowych maszyn RISC UNIX.

## Historia Produktu

Technologia Relacyjnych Baz Danych rozwija się już od ćwierćwiecza. Profesjonalne i przemysłowe systemy zarządzania relacyjnymi bazami danych, które są obecnie liderami rynku, zaczęły powstawać na przełomie lat 70 i 80-tych. Ich technologia była projektowana dla bardzo odmiennych typów aplikacji i innego sprzętu niż te, które są obecnie najczęściej spotykane. Ciągłe rosnąca konkurencja wśród producentów, ewolucja wymuszająca wdrażanie nowych koncepcji, przy jednoczesnej konieczności zachowania ciągłości cech funkcjonalnych, spowodowały, że produkty budowane w oparciu o tradycyjną technologię rozrastały się przez dodawanie coraz to nowych warstw programowych. W efekcie dzisiejsze systemy zarządzania bazami danych, jako wynik długotrwałej i intensywnej ewolucji, mają wysokie wymagania dotyczące sprzętu i są bardzo kosztowne w eksploatacji oraz administrowaniu.

Tymczasem techniki konstruowania bardzo efektywnych i wydajnych „software’owych maszyn bazodanowych” przestały być tajemnicą i będąc przedmiotem studiów ośrodków rozwojowych i uniwersyteckich, znacznie wyprzedzają to, co oferuje coraz bardziej konserwatywny przemysł.

Stwarza to szansę dla firm, które podejmują wysiłek zbudowania systemów bazodanowych od nowa, w oparciu o szczegółową analizę wymagań stawianych przez współczesnego użytkownika informatyki.

Te wymagania to:

- Łatwość scalenia z aplikacją i łatwość masowych instalacji. Baza danych powinna sprawiać jak najmniej kłopotu użytkownikom - powinna mieć minimalne wymagania administracyjne.
- Zgodność ze wszystkimi standardami technologii baz danych i ponadto zdolność do działania serwera bazy danych na wszystkich podstawowych platformach.
- Przenośność na poziomie plików binarnych.
- Skalowalność od zera do praktycznej nieskończoności.
- Zwartość i łatwość dystrybucji serwera bazy danych przy utrzymaniu funkcjonalności i wydajności zgodnej z wymaganiami stawianymi profesjonalnym i przemysłowym systemom bazodanowym.

Firma Solid Information Technology Ltd. z Finlandii zbudowała swój system zarządzania relacyjną bazą danych mając na uwadze właśnie te wymagania. Solid Serwer jest ukierunkowany na wciąż rozwijającą się technologię systemów Klient/Serwer.

Firma Solid Information Technology Ltd. powstała w roku 1992 jako wydzielona grupa z KT-Datacenter - jednej z trzech największych firm software’owych w Finlandii. W roku 1995 KT-Datacenter osiągnęła obrót 200 milionów USD i zatrudniała przeszło 700 pracowników.



## **Korzyści z użycia SOLID Server dla UNIFACE**

SOLID Server dla UNIFACE może być wykorzystywany jako miejsce przechowywania repozytorium systemu do budowania i rozwoju aplikacji UNIFACE Six i/lub jako docelowa baza danych dla aplikacji UNIFACE.

Użytkownicy produktu SOLID Server dla UNIFACE uzyskują wymierne korzyści z jego użycia. Oto kilka przykładów:

### **Projektanci aplikacji UNIFACE**

- Otrzymują skalowalne, elastyczne i łatwe w obsłudze środowisko do pracy zespołowej.
- Wymagania sprzętowe stacji roboczej są niskie - dla stanowiska projektanta na pojedynczym PC (UNIFACE+SOLID Server) wystarczy 8 MB RAM i ok. 45 MB przestrzeni na dysku.
- Dzięki zaawansowanej technologicznie architekturze SOLIDA projektanci mogą testować rozwiązania dostępne tylko w dużych i kosztownych systemach bazodanowych (procedury zapamiętywane, zarządzanie transakcjami itp.).

### **VAR-owie UNIFACE**

- Producenci dedykowanego i specjalizowanego oprogramowania mogą tworzyć wersje swoich programów, w których baza danych jest osadzona w aplikacji i działa jak „czarna skrzynka” - w sposób niewidoczny dla użytkownika. Jest to możliwe dzięki niezwykle małym wymaganiom odnośnie administrowania i pielęgnacji.
- Dzięki temu, że UNIFACE pozwala tworzyć aplikacje niezależne technologicznie, to samo oprogramowanie może być sprzedawane w konfiguracjach wykorzystujących inne popularne systemy bazodanowe.

### **Przedsiębiorstwa rozpoczynające proces informatyzacji**

SOLID Server dla UNIFACE jest idealnym rozwiązaniem dla firm rozpoczynających proces informatyzacji przedsiębiorstwa lub przechodzących z aplikacji w Clipperze, dBase do systemów o architekturze Klient/Server:

- Niska cena Serwera SOLID i małe wymagania sprzętowe, przy jednoczesnej bogatej funkcjonalności i skalowalności, pozwolą na uniknięcie wysokich kosztów informatyzacji oraz umożliwią ewolucyjny rozwój systemów w firmie. Daje to możliwość budowania infrastruktury informatycznej stopniowo - przez zastępowanie starych systemów nowoczesnymi rozwiązaniami technologicznymi.
- Małe wymagania odnośnie administrowania i pielęgnacji umożliwią uniknięcie kosztów związanych ze szkoleniem administratorów baz danych.

### **Przedsiębiorstwa dokonujące masowej dystrybucji oprogramowania**

Firmy używające aplikacji UNIFACE w połączeniu ze strategicznym systemem bazodanowym, muszą czasem dystrybuować swoje aplikacje wraz z bazą danych do różnych ośrodków terenowych lub wręcz umieszczać je na mobilnych stacjach roboczych (notebookach). SOLID Server dla UNIFACE jest tu doskonałym i ekonomicznym rozwiązaniem.



## ***Kompleksowe wdrożenie technologii informatycznej firmy Compuware***

Niektóre przedsiębiorstwa wybierają Compuware jako głównego dostawcę rozwiązań technologicznych dla kompleksowej informatyzacji firmy. Wybór ten jest zwykle podyktowany tym, że produkty Compuware pokrywają cały cykl życia systemów, jak również poziomem usług informatycznych (konsultingu, szkoleń i pomocy technicznej) od projektowania i rozwoju, aż do wspomagania eksploatacji. Firmy te kupują od jednego dostawcy takie dodatkowe produkty jak: narzędzia do raportowania (UNIFACE Personal Series), narzędzia ułatwiające tworzenie aplikacji w Internecie (UNIFACE Web Enabler), narzędzia wykorzystujące złożone usługi w sieciach komputerowych (UNIFACE Distributed Computing Management) oraz zaawansowane systemy bazodanowe (SOLID Server dla UNIFACE).

### ***Kto używa SOLID Server dla UNIFACE?***

Mimo krótkiego czasu istnienia produktu SOLID Server, wiele firm w Europie i na świecie używa go jako systemu bazodanowego w ważnych dla firmy systemach informatycznych, a także jako repozytorium UNIFACE, w przypadkach konstruowania systemów i pakietów aplikacyjnych. Poniżej przedstawionych jest kilka przykładów takich firm:

**ARKE** - jedna z największych agencji turystycznych w Holandii używa aplikacji napisanej w UNIFACE i wykorzystującej serwery SOLID w zintegrowanym systemie informacyjnym, w około 200 swoich biurach podróży.

**Compu-Quote Inc.** - dostarcza usług ubezpieczeniowych w dziedzinie ubezpieczeń nieruchomości na terenie całej Kanady. Wszystkie aplikacje działające w 90 biurach regionalnych napisane są przy użyciu UNIFACE Six. Obecnie powstaje nowa generacja integrująca wszystkie aplikacje. Compu-Quote wybrał SOLID Server dla UNIFACE jako centralną platformę bazodanową, która będzie działać na maszynie Pentium. Wielkość bazy danych wyniesie 2 GB, zaś aplikacja będzie obsługiwać docelowo 1500 użytkowników.

**Fundia Wire** - firma przemysłu metalurgicznego używa aplikacji kadrowej napisanej w UNIFACE z bazą danych SOLID. Baza ma 500 MB (największa tablica ma 3.000.000 wierszy) i działa na Digital VAX 3100/90 24 VUPS.

**KONE Elevators** - jedna z największych na świecie firm produkujących windy. Używa Serwerów SOLID do przechowywania parametrów eksploatacyjnych i optymalizacyjnych w serwisie swoich klientów.

**Miasto Raisio** - zatrudnia przeszło 1700 pracowników. Aplikacje działają na 250 stacjach OS/2. Serwerem jest maszyna Intel 133 MHz Pentium z OS/2. Wielkość bazy danych wynosi 2 GB. Wybór SOLIDA dla aplikacji UNIFACE został dokonany po analizie możliwości zastosowania ORACLE. Analiza wykazała, że użycie ORACLE, jako systemu zarządzania bazą danych, wymagałoby zastąpienia serwera Pentium maszyną RISC, a wielkość bazy danych wzrosła by do 3 GB.

---

\* W Polsce usługi profesjonalne w tym zakresie prowadzi Generalny Dystrybutor produktów Compuware - POLNET Technologies.

**Minihouse Groningen** - VAR UNIFACE'a w Holandii. Jednym z wiodących produktów tej firmy jest znany na świecie pakiet INFOSALE - system obsługi sprzedaży i marketingu. SOLID Server jest używany w większości sprzedawanych wersji INFOSALE - od jednostanowiskowych, działających na notebookach, do konfiguracji wielostanowiskowych, integrujących się z innymi aplikacjami w skali przedsiębiorstwa.

**Tekla** - jest największym producentem systemów GIS w Finlandii. Ostatnio zbudowała wielki system dla fińskiego przemysłu drzewnego. Istotnym elementem aplikacji są mobilne stacje robocze wykorzystujące lokalne bazy danych SOLID. SOLID jest także centralną bazą danych działającą na platformie operacyjnej HP-UX.

**Teleste** - specjalizuje się w budowie laboratoriów językowych. Firma oferuje ostatnio nową generację środowisk do nauki języków, wykorzystujących intensywnie multimedia (Internet, TV, kasety video itp.). Jedną z podstawowych części laboratorium jest system administrowania danymi oparty na SOLIDZIE dla UNIFACE i na serwerze Windows NT. Średnia wielkość instalacji 50 - 100 użytkowników.

### **Komponenty SOLID Server dla UNIFACE**

Produkt SOLID Server dla UNIFACE składa się z następujących części składowych:

**SOLID Server** - system zarządzania bazą danych.

**SOLID Remote Control** - jest to program administrowania serwerami SOLID. Pozwala on na:

- administrowanie wszystkimi bazami danych SOLID w sieci z jednej stacji roboczej,
- generację kopii awaryjnych (Backups) bądź w trybie „on-line”, bądź według określonego harmonogramu czasowego,
- otrzymywanie informacji statusowej,
- konfigurowanie i strojenie serwera(ów).

**SOLID SQL Editor** - jest to narzędzie (zgodne ze standardem ODBC) pozwalające na wykonywanie SQLowych zapytań do bazy danych SOLID. Posiada graficzny (lub znakowy) interfejs użytkownika. SQL Editor umożliwia:

- interakcyjny lub wsadowy tryb pracy,
- otwarcie komunikacji z wieloma bazami SOLID równocześnie,
- zapisywanie wyników do plików lub ich drukowanie.

**SOLID ODBC Driver** - umożliwia dostęp do danych SOLID Serwer dla UNIFACE z aplikacji i narzędzi zgodnych ze standardem ODBC (Open Database Connectivity). SOLID ODBC Driver jest dostępny dla platform MS-Windows 3.x, Windows 95 i Windows NT.

**SOLID UNIFACE Driver** - własny drajwer UNIFACE umożliwiający bezpośredni dostęp do danych z aplikacji UNIFACE do bazy SOLID. Jest on obecnie dostępny dla następujących platform:

- MS-Windows 3.1x.
- MS-Windows 95.



POLNET Technologies prowadzi działalność w Polsce od 1990 roku. Jesteśmy średniej wielkości firmą informatyczną dysponującą szeregiem specjalistów o wysokich kwalifikacjach (projektanci, analitycy, programiści), posiadającą wieloletnie, poparte solidnymi referencjami doświadczenie w budowaniu systemów informatycznych. Nasi klienci to między innymi: Narodowy Bank Polski, Powszechna Kasa Oszczędności, Dyrekcja Generalna Lasów Państwowych, Wydawnictwo Sigma NOT, Główny Inspektorat Pracy, Centralny Zarząd Zakładów Karnych.

Nasza firma oferuje swoim klientom usługi w dziedzinie:

- konsultingu, doradztwa informatycznego i integracji systemów informatycznych,
- wykonywania oprogramowania na zamówienie,
- szkoleń w zakresie metodyki prowadzenia przedsięwzięć informatycznych,
- sprzedaży profesjonalnych narzędzi dla informatyków.

POLNET proponuje również rozwiązania wykorzystujące technologię kart inteligentnych w systemach kontroli dostępu fizycznego i logicznego, kontroli czasu pracy i portmonetki elektronicznej.

W ciągu ponad pięciu lat działalności na polskim rynku dopracowaliśmy się sprawdzonych w praktyce metod działania. Podstawowymi przesłankami kierującymi naszą firmą przy współpracy z klientem są:

- indywidualne podejście do każdego przedsięwzięcia,
- uniezależnienie klienta od dostawców sprzętu i oprogramowania,
- pełna kontrola klienta nad przebiegiem prac.

POLNET jest firmą nastawioną na wdrażanie nowoczesnych technologii informatycznych. Dzięki szerokim kontaktom międzynarodowym możemy na bieżąco śledzić najnowsze tendencje i kierunki rozwoju informatyki oraz jej zastosowań. Obecnie promujemy najnowsze osiągnięcia i technologie informatyczne, takie jak:

- pakiet *Software through Pictures* (StP) amerykańskiej firmy *Interactive Development Environments* - wielodostępne, zintegrowane narzędzie typu CASE, wspomagające zarówno obiektowe, jak i strukturalne metodyki analizy i projektowania systemów informatycznych.
- zintegrowane środowisko projektowe drugiej generacji *UNIFACE Six*, firmy *Compuware Corporation*, służące do szybkiego rozwoju i konstruowania skalowalnych i technicznie niezależnych systemów informatycznych o architekturze Klient/Serwer.
- karty inteligentne francuskiej firmy *Gemplus*.



# DELPHI

## - narzędzie do szybkiego tworzenia aplikacji.

Mariusz Gibutowski, Grzegorz Wojamik

Instytut Cybernetyki Ekonomicznej i Informatyki, Uniwersytet Szczeciński

### 1. Podstawowe wiadomości na temat pakietu Borland Delphi.

Delphi jest nowym narzędziem programistycznym typu RAD. Nazwa pochodzi od antycznego miasta greckiego Delfy, gdzie według wierzeń ówczesnych miała znajdować się wyrocznia boga Apollina.

Jest to narzędzie charakteryzujące się techniką wizualnego tworzenia graficznego interfejsu użytkownika, generujące 16-bitowy kod dla środowiska MS Windows 3.1. Pakiet jest przeznaczony do tworzenia praktycznie wszystkich typów aplikacji, przy czym ma silne narzędzia do budowy systemów zarządzania bazami danych, a zwłaszcza graficznych interfejsów - nakładek na duże SZBD, np. ORACLE.

Pakiet ten generuje pliki z rozszerzeniem .EXE, zawierające kod maszynowy, w odróżnieniu od niektórych innych narzędzi tej klasy które generują kod interpretowany. Dzięki temu aplikacja zyskuje kilkunastokrotnie na szybkości działania.

W rezultacie takiego podejścia programy wynikowe są dużych rozmiarów, gdyż zawierają w sobie biblioteki procedur, podczas gdy programy interpretowane z innych pakietów mimo że są stosunkowo krótkie, wymagają modułu interpretującego.

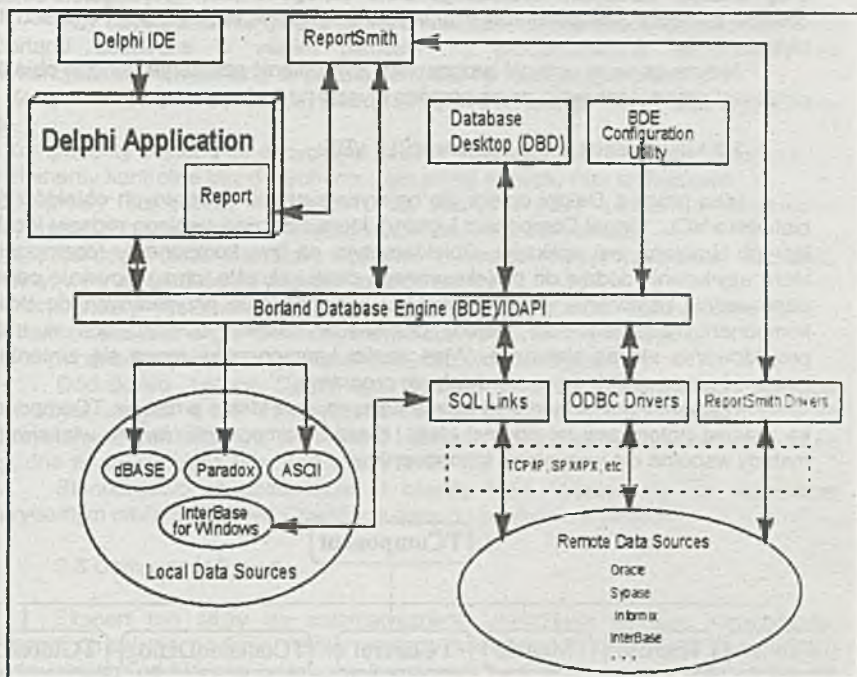
Oprócz budowy samych aplikacji, pakiet umożliwia tworzenie bibliotek .DLL, które mogą być później wielokrotnie wykorzystywane nie tylko przez Borland Delphi, ale również przez C++, Visual Basic, Power Builder i inne systemy.

Delphi oparty jest na języku Object Pascal, wywodzącym się w prostej linii od Turbo Pascala tej samej firmy. Jest on zgodny z Borland Pascalem 7.0, dzięki czemu aplikacje stworzone w tym ostatnim kompilują się w Delphim. Jest to język w pełni obiektowy, umożliwiający dziedziczenie, enkapsulację, czy polimorfizm. Silną stroną pakietu jest dziedziczenie, czyli tworzenie własnych klas pochodnych na podstawie klas ogólniejszych.

Na uwagę zasługuje bogaty interfejs użytkownika. Programista ma przed sobą zintegrowane środowisko pracy, tzw. IDE (Integrated Development Environment), na które składają się

- paleta komponentów (Component Palette), czyli skrzynka z narzędziami, gdzie na każdej zakładce umieszczonych jest kilka wizualnych kontroltek, np. pole edycji, etykieta, lista rozwijalna,
- pasek narzędzi do zarządzania projektem, typu : dołączanie i usuwanie pliku do projektu, uruchamianie i zatrzymywanie programu, zapis i odczyt projektu z dysku.
- okno formularza, na którym umieszcza się wizualne elementy pobrane z palety komponentów,
- okno kodu programu z zakładkami na których znajdują się kolejne moduły,
- okno właściwości (Object Inspektor), gdzie wyświetlone są właściwości wybranego elementu,
- okno menadżera projektu (Project Manager),
- oraz kilka innych, czasowo włączanych okien.

Delphi posiada dostęp do dużych baz danych np. ORACLE, Sybase, Informix, oraz Microsoft SQL Server poprzez Borland SQL Links, mechanizm właściwy dla firmy Borland. Oprócz tego pakiet zawiera sterowniki ODBC (Open Database Connectivity), mimo że jest to standard konkurencyjnego Microsoftu. Dzięki nim ma dostęp do baz : Access, Betrieve, Excel, DB2, AS/400, Ingres, HP ALBASE/SQL.



Rys. 1. Architektura dostępu do baz danych w Delphi.

Dla potrzeb komunikacji z firmowymi produktami Borlanda, typu dBase, czy Paradox, wykorzystuje własny mechanizm BDE (Borland Database Engine). Oprócz tego zawiera własny serwer bazy: Interbase.

Jedną z silniejszych stron pakietu jest jego debugger. Zintegrowany z pakietem, uruchamiany jest z poziomu menu i udostępnia podstawowe opcje:

- śledzenie zawartości zmiennych,
- krokowe uruchamianie aplikacji,
- możliwość tworzenia kontrolnych punktów przerwań (breakpoints).

Debugger działa tutaj na podobnych zasadach jak w Turbo Pascalu. Co ciekawe, podczas kompilacji Delphi informuje o niewykorzystanych zmiennych, niezainicjowanych wskaźnikach, czy pustych pętlach.

Delphi posiada dobrze zorganizowaną Pomoc On-Line. Informacje na żądany temat można znaleźć zarówno poprzez Spis Treści, jak i opcję Szukaj, tak jak w innych programach środowiska Windows. Zaletą jest dokładny opis funkcji systemowych API (Application Programming Interface).



## 2. Cechy pakietu umożliwiające szybkie tworzenie aplikacji.

### 2.1 Drag & Drop.

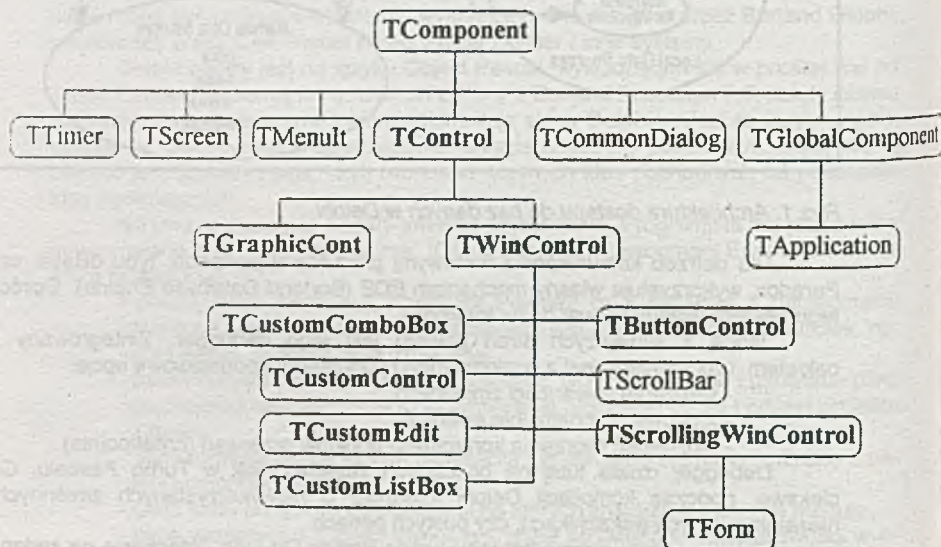
Praca z Delphi przy tworzeniu interfejsu użytkownika opiera się na technice drag & drop. Zadaniem tworzącego okno dialogowe jest wybór odpowiedniego obiektu z Palety Komponentów i umieszczenie go w odpowiednim miejscu tego okna.

Jedyną co musi uczynić programista aby zmienić położenie takiego obiektu w oknie jest przemieszczenie go za pomocą myszki na żądaną pozycję.

### 2.2 Kilkadziesiąt komponentów VCL i VBX.

Idea pracy z Delphi opiera się na wykorzystaniu specjalnych obiektów (tzw. biblioteka VCL: Virtual Component Library), które stanowią swojego rodzaju klocki, z których tworzona jest aplikacja. Obiektami tymi są tzw. komponenty (components), które użytkownik dodaje do projektowanego okna i określa ich zachowanie poprzez odpowiednie ustawienie ich właściwości i wykorzystanie przypisanych do takiego komponentu zdarzeń oraz metod. Oczywiście ustawienia dokonane w trakcie projektowania nie są statyczne. Właściwości komponentów mogą się zmieniać w zależności od potrzeb w trakcie działania programu.

Wszystkie komponenty są oparte na specjalnej klasie o nazwie TComponent i są właśnie potomkami tej ogólnej klasy. Klasa TComponent zawiera właściwości i metody wspólne dla wszystkich komponentów.



Rys. 2. Hierarchia obiektów VCL.



Użytkownik może definiować i wykorzystywać dwa rodzaje komponentów:

- komponenty wizualne; te które widoczne są po skompilowaniu na formacie ekranowej,
- komponenty niewizualne; które po skompilowaniu programu nie są widoczne na oknie dialogowym.

Aby użytkownik nie musiał tworzyć biblioteki komponentów od podstaw firma Borland dostarcza w swoim pakiecie 73 predefiniowane komponenty do wykorzystania. Komponenty te można podzielić na następujące grupy:

- Komponenty komunikacji z użytkownikiem (np. pole edycji, menu, tabela, checkbox itd.),
- Komponenty dostępu do danych (np. tablica, zapytanie, baza danych, raport itd.)
- Elementy kontrolne baz danych (np. pole edycji danych, lista lookup-owa, lookup-owy combobox, nawigator bazy danych, który umożliwia dodawanie, kasowanie, poruszanie się pomiędzy rekordami bazy danych itd.)
- Standardowe okna dialogowe systemu Windows (otwórz, zachowaj, czcionka, kolory, wydruk, ustawienia drukarki, znajdź, zamień)
- Komponenty systemowe (kontener OLE, obiekty obsługi mechanizmu DDE - Dynamic Data Exchange, elementy standardowych okien dialogowych)
- Komponenty przykładowe (np. kalendarz, drzewo katalogów itp.).

Dodatkowo twórcy Delphi-ego uwzględnili dużą popularność obiektów wykorzystywanych w MS Visual Basic, tzw. VBX-ów i dodali do Delphi możliwość korzystania z tych obiektów. Z obiektów tych można korzystać pod warunkiem, że są zgodne ze specyfikacją Microsoft Visual Basic 1.0.

Standardowo są dostarczone 4 obiekty VBX. Wydaje się, że najbardziej przydatnym może być obiekt ChartFx służący do tworzenia wykresów.

### 2.3 Component Expert.

Ekspert ten służy do automatycznego utworzenia nowego komponentu. Ekspert umożliwia podanie nazwy nowego komponentu, określenia komponentu nadrzędnego, od którego budowany komponent będzie potomkiem oraz strony na Paletce Komponentów, na której będzie umieszczony. Zadaniem programisty jest dodanie już ręcznie kodu, który będzie określał nowe cechy budowanego komponentu.

### 2.4 Two-Way-Tools.

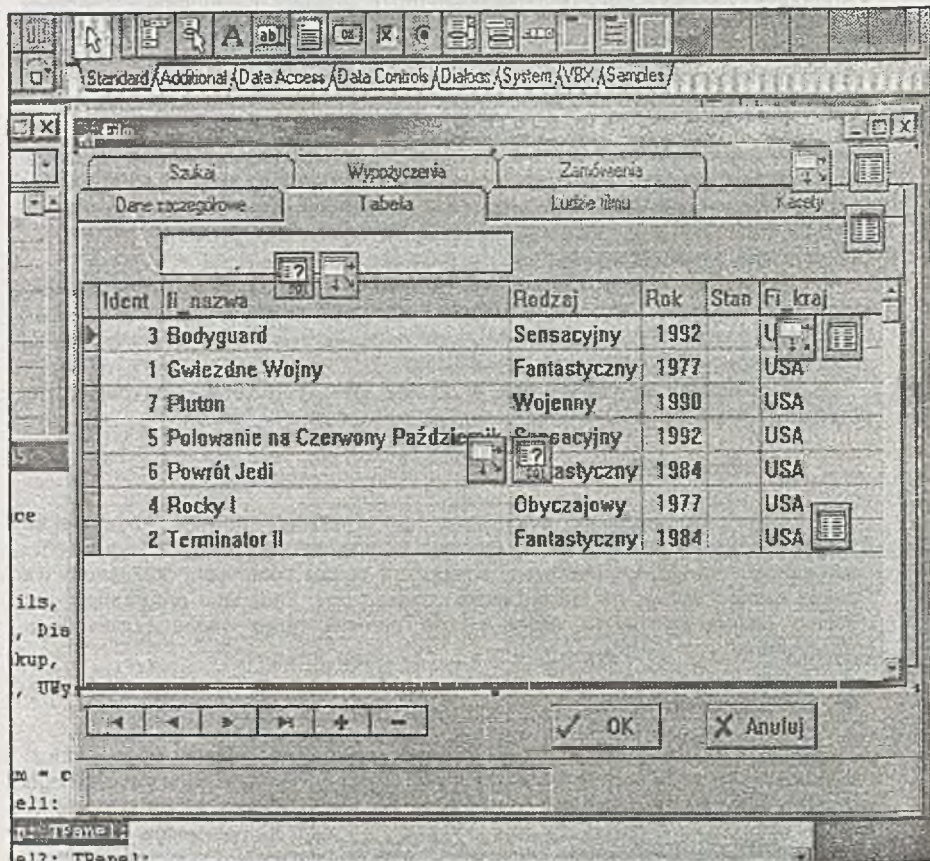
Jest to technika polegająca na automatycznym wprowadzaniu odpowiednich deklaracji i poleceń w kodzie programu w zależności od wyglądu okna zmienianego wizualnie poprzez dodawanie lub usuwanie obiektów. Polega to na tym, że w momencie dodania do okna dowolnego obiektu z Palety Komponentów deklaracja tego obiektu jest automatycznie generowana w kodzie programu. Dodatkowo zmiana nazwy takiego obiektu powoduje zmianę nazwy zadeklarowanego obiektu w kodzie programu. Oczywiście z chwilą usunięcia komponentu z okna jego deklaracja jest kasowana z kodu programu.

Podejście takie pozwala użytkownikowi na oszczędności czasowe związane z brakiem fazy generowania kodu aplikacji po każdej modyfikacji interfejsu.

## 2.5 Podgląd danych w trybie projektowania.

Kolejnym elementem ułatwiającym projektowanie aplikacji jest możliwość oglądania danych w postaci w jakiej będą widoczne po skompilowaniu programu już w trybie projektowania.

Przykładowo po wybraniu obiektu DBGrid (arkusz bazy danych) i podłączeniu go do odpowiedniej tablicy bądź zapytania użytkownik ma możliwość już w trakcie tworzenia okna dialogowego zobaczyć w jaki sposób realne dane będą się prezentowały w tym oknie po procesie kompilacji programu.



Rys.3. Okno w czasie projektowania wyświetla faktyczne dane.

## 2.6 Experts i Templates.

Delphi dysponuje dwoma ekspertami umożliwiającymi łatwą generację okien dialogowych.



- Database Form Expert (ekspert okien baz danych), który pozwala na tworzenie okien, które umożliwiają na manipulację danymi z baz danych obsługiwanych przez Delphi. Ekspert ten pomaga w tworzeniu dwóch typów okien bazodanowych:

- proste okno bazy danych,

- okno bazy danych master/detail, gdzie realizowane jest powiązanie pomiędzy tabelami typu "1 do n".

Ekspert okien baz danych automatyzuje połączenie z wybraną tablicą lub tworzy zapytanie SQL-owe w zależności od dokonanego wyboru, umieszcza automatycznie na oknie dialogowym elementy kontrolne baz danych powiązane z wybraną wcześniej tablicą lub utworzonym zapytaniem, umożliwia określenie kolejności dostępu do elementów kontrolnych bazy danych.

- Dialog Ekspert - umożliwia automatyczną generację okien dialogowych nie związanych z bazami danych. Za pomocą tego eksperta użytkownik może określić czy tworzone okno będzie miało tzw. zakładki oraz w jaki sposób będą wyświetlone standardowe przyciski (OK, Cancel, Help) na oknie.

Delphi dostarcza gotowe do wykorzystania szablony okien i aplikacji. Są to tzw. Form Templates (szablony okien) i Project Templates (szablony aplikacji). Szablony okien umożliwiają szybkie wygenerowanie okien: tzw. "O..." (informacyjne), z podwójną listą (w którym z jednej listy można przesyłać pozycje do drugiej), dwa formaty okien z zakładkami, standardowe okno dialogowe (na którym są umieszczone tylko przyciski OK, Cancel, Help) oraz okno z hasłem. Warto dodać, że w zależności od potrzeb użytkownik ma możliwość samodzielnego tworzenia tego typu szablonów dla okien, które mają w jego programach taką samą postać. Szablony aplikacji umożliwiają szybkie wygenerowanie okna głównego dla aplikacji MDI (Multi Dialog Interface), SDI (Single Dialog Interface) oraz CRT (aplikacja dla tworzenia tekstowych programów w Pascalu). W przypadku szablonów aplikacji można również tworzyć szablony własne.

## 2.7 Zarządzanie aplikacją (Project Manager)

Project Manager jest narzędziem dającym użytkownikowi możliwość panowania nad tworzoną aplikacją. Za jego pośrednictwem w prosty sposób można włączać do i wyłączać z projektu jednostki kodu lub okna dialogowe, w zależności od naszych zamiarów.

Dzięki niemu możemy określić, które okno jest oknem głównym aplikacji, jakie okna są tworzone automatycznie w pamięci w czasie uruchomienia programu. Wszystkie zmiany, które dokonamy za pośrednictwem Managera zostaną uwzględnione w kodzie programu. Możemy również ustawiać dzięki niemu właściwości kompilatora.

Wyżej wymienione właściwości, których celem jest przyspieszenie procesu tworzenia aplikacji w Borland Delphi, nie są całkowicie nowatorskie. Istnieją już narzędzia posiadające powyższe zalety, jednak ich skupienie w Delphi jest tak duże, że produkt ten wydaje się w pełni spełniać wymogi nowoczesnego narzędzia RAD.

## 3. Narzędzia dodatkowe.

Pakiet Delphi wyposażony jest w wiele zewnętrznych narzędzi dodatkowych. Do zakładania, przeglądania i modyfikacji struktury danych służy Database Desktop, którego można wywołać z poziomu menu. Jest on wyposażony w podstawowe



funkcje operacji na danych, ale w zupełności wystarcza do założenia praktycznie każdej struktury. Pliki w formacie .PDX mogą być również założone przez Paradoxa.

Do tworzenia raportów służy program Report Smith, komunikuje się on z Delphim poprzez DDE i umożliwia tworzenie raportów praktycznie każdego rodzaju.

Pakiet posiada własny serwer InterBase. Dostęp do niego może być realizowany poprzez Borland Database Engine i InterBase SQL Link. Posiada wersję jednostanowiskową oraz wielodostępną: Borland InterBase Workgroup Server.

Zewnętrzny program WinSight umożliwia śledzenie otwieranych okien, klas, oraz wysyłanych komunikatów. Dzięki niemu można śledzić cały proces wywołań poszczególnych klas aplikacji, jednak nie można niczemu przeszkodzić. Jest to więc bierna przeglądarka.

WinSpector natomiast służy do sprawdzania zaistniałych błędów systemowych UAE (Unrecoverable Application Errors) oraz błędów ochrony GPE (General Protection Error). W przypadku zaistnienia błędu program zapisuje informację o nim do pliku .LOG, który później pomaga w usunięciu problemu.

Do znalezienia kodu dla wybranego obiektu służy Object Browser. Można w nim przeglądać pascalowe moduły, obiekty, symbole lub procedury i zmienne.

Wraz z pakietem dostarczana jest też przeglądarka Acrobat Reader. Dzięki niej możliwe jest przeglądanie na ekranie pełnej dokumentacji, którą producent dostarcza również w formie elektronicznej w formie plików .PDF. Elektroniczne podręczniki są wierną kopią papierowych odpowiedników. Można je przeglądać strona po stronie tak jak książki, gdyż zachowano nawet oryginalny podział na strony. Wersja elektroniczna ma kilka zalet w stosunku do książek. Oferuje szybkie wyszukiwanie ciągu znaków, wycinanie, kopiowanie i wklejanie dowolnych fragmentów tekstu, oraz szybkie poruszanie się między rozdziałami, punktami i tematami podręczników.

#### 4. Nowe elementy zawarte w wersji 2.0.

Upgrade 2.0 pakietu rozprowadzany jest w 3 wersjach:

- Delphi Desktop,
- Delphi Developer,
- Delphi Client/Server.

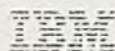
Delphi 2.0 przeznaczony jest dla środowiska Windows 95 i Windows NT. Generuje on w pełni 32-bitowy kod wykonywalny. Nowy program wzbogacony jest o dodatkowe komponenty wizualne, których jest tu ponad 90. W tym względzie firma Borland przestawiła się na standard OCX.

Nowa wersja wyposażona jest w repozytorium, który znacznie ułatwi pracę z zewnętrznymi bazami danych. Delphi 2.0 zawiera też nowy mechanizm OLE Automation, pozwalający na bardziej wydajną pracę z aplikacjami mogącymi pracować jako serwery OLE, np. Paradox, czy Borland C++.

Wersja ta wyposażona została też w wewnętrzny program do modyfikacji struktury danych Database Explorer, który jest integralną częścią środowiska programisty.

## Bibliografia :

1. Visual Basic 4.0 kontra Delphi 2.0: którą platformę wybrać. S. Ryan. Computerworld, Nr 8/228, 1996.
2. Kłopoty z narzędziami dla klienta. Test trzech pakietów wspomagających proces tworzenia aplikacji. Computerworld, Nr 36/208, 1995.
3. Visual Basic i Delphi – łeb w łeb. Stephen W. Plain. PC Magazine, Nr 12/34. 1995.
4. Delphi 2.0. 32-bitowa wyrocznia. Michał Gołębiwski. Enter, Nr 3, 1996.
5. Wyrocznia. Tilman Bömer. Chip, Nr 6, 1995.
6. Borland Delphi for Windows. Database Application Developer's Guide. Borland International 1995.
7. Borland Delphi for Windows. User's Guide. Borland International 1995.



IBM Polska Sp z o.o. z siedzibą w Warszawie

ul. Nowy Świat 15/17  
sk. 399  
00-950 Warszawa 1, Poland  
Telefon (+48-22) 625-10-10  
Telefax (+48-22) 625-70-03

## IBM POLSKA POSZUKUJE PARTNERÓW WYTWARZANIA APLIKACJI

Krzysztof Chelpiński



Gdy 6 lutego 1996 roku na scenie Centrum Kongresowego w San Diego pojawił się szef IBM Louis Gerstner obecne na sali 3500 osób wstrzymało oddech. Słowa, które usłyszeli wywołały mocniejsze bicie serca i wyraz zadowolenia na ich twarzach. Publiczność zgromadzona na sali to Partnerzy Handlowi IBM a słowa do nich skierowane obwieszczały nową strategię firmy IBM wobec partnerów.

Strategia opisana została w dokumencie nazwanym "Kartą Partnerstwa IBM". Dokument ten, będący swoistą konstytucją, ustanawia nowe zasady współpracy pomiędzy IBM a partnerami handlowymi. Partnerzy otrzymują pełne wsparcie technologiczne i sprzętowe, zyskują dostęp do olbrzymich zasobów wiedzy i doświadczeń Korporacji oraz mogą także korzystać z wypracowanych przez IBM metodologii.

IBM będzie wspierał partnerów oferujących własne rozwiązania, a także pomagał przy tworzeniu nowych, najlepiej spełniających indywidualne oczekiwania klientów.

Przy dużych, unikatowych projektach IBM będzie zapraszał swoich partnerów do współpracy w charakterze podwykonawców, natomiast przy projektach korzystających ze stosowanych już rozwiązań partnerzy przejmą całość realizacji.

Dynamiczny rozwój polskiego rynku stawia przed firmami komputerowymi coraz wyższe wymagania. Klienci potrzebują różnorodnych rozwiązań informatycznych, dopasowanych do ich potrzeb i dostarczonych na dogodnych warunkach.

IBM na całym świecie od wielu lat tworzy i dostarcza najwyższej klasy rozwiązania przeznaczone dla wszystkich dziedzin gospodarki: przemysłu, handlu, administracji, bankowości, ubezpieczeń, szkolnictwa itp. Wieloletnie doświadczenie IBM na rynkach światowych stanowi gwarancję także dla polskich klientów, iż firmy rekomendowane przez IBM Polska nie zawiodą ich oczekiwań.

Każdy nowy projekt będzie przygotowywany wspólnie przez specjalistów IBM i partnerów handlowych, aby wybrać rozwiązanie w pełni odpowiadające konkretnym potrzebom klienta.

Nowa globalna strategia IBM wobec partnerów handlowych skierowana jest także do polskich partnerów i polskiego rynku. W naszym kraju IBM współpracuje z 87 partnerami handlowymi. Proponowane przez nich rozwiązania oparte są na komputerach typu AS/400 oraz RS/6000, a także na platformach komputerów osobistych. Z wieloma z nich IBM zrealizował szereg modelowych kontraktów, których rezultaty w pełni zadowolili naszych klientów.

Wsparcie IBM dla firm partnerskich przyczyni się do ich rozwoju a niektórym z nich da szansę promocji własnych rozwiązań. Zadaniem IBM Polska jest zbudowanie solidnej oferty oprogramowania aplikacyjnego dla polskiego rynku, oprogramowania wykorzystującego platformy technologiczne IBM i zbudowanego z wykorzystaniem naszych metodologii. Będziemy zachęcać polskie firmy zajmujące się wytwarzaniem oprogramowania do udziału we wspólnych projektach, jak również do tworzenia lub lokalizacji standardowych pakietów aplikacyjnych mogących w przyszłości stanowić część oferty IBM Polska.



# CENTRUM TECHNIKI OKRĘTOWEJ

## Ship Design and Research Centre

Rok założenia 1971

Kontynuator dorobku okrętowych biur projektowo-konstrukcyjnych  
powstałych w 1945 i 1951 roku

Wały Piastowskie 1  
80-958 GDANSK

tel.: 58 374201  
374203

fax: 58 311683  
313935

e-mail: research@cto.gda.pl

### Dyrektor

Zbigniew KARPIŃSKI  
tel.: 58 310042

Z-ca dyrektora  
ds. badań i rozwoju  
Jan DUDZIAK  
tel.: 58 316946

Z-ca dyrektora ds.  
ekonomicznych  
Główny księgowy  
Mieczysław DAWICKI  
tel.: 58 313291

Kierownik Zakładu  
Badawczo-Rozwojowego  
Wojciech WSZĘDYBYŁ  
tel.: 58 316011

**C**entrum Techniki Okrętowej jest specjalistyczną placówką badawczo-rozwojową, projektową i ogólnotechniczną pracującą przede wszystkim na rzecz przemysłu okrętowego. Swoją działalność wykorzystuje także do rozwiązywania problemów technicznych innych dziedzin gospodarki: przemysłu maszynowego, energetyki, budownictwa lądowego i wodnego, rolnictwa, przemysłu spożywczego i chemicznego, komunikacji i łączności.

### CTO prowadzi badania i pomiary:

- modelowe właściwości oporowo-napędowych, manewrowych i morskich statków i innych obiektów pływających
- podczas prób zdawczych statku na morzu: prędkości statku, mocy, obrotów silników głównych i pomocniczych, właściwości manewrowych i morskich, drgań, hałasu, naprężeń (odkształceń) konstrukcji oraz mocy indykowanej silników spalinowych
- odpowiedzi dynamicznych konstrukcji
- właściwości fizyko-mechanicznych i sanitarno-chemicznych materiałów
- korozji i ochrony przed korozją
- prototypowych urządzeń i mechanizmów
- mechanizmów pokładowych np.: wciągarek, dźwigów itp.

### CTO projektuje

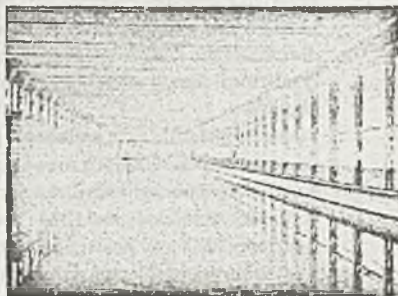
- kształty kadłuba, pędników i urządzeń sterowych dla statków
- izolacje akustyczne
- napędy, rozdzielnice i instalacje elektryczne
- systemy, mechanizmy i urządzenia siłowni okrętowych, systemy i urządzenia chłodnicze i klimatyzacyjne, ochrony p.poż. i ochrony środowiska

### CTO wykonuje instalacje

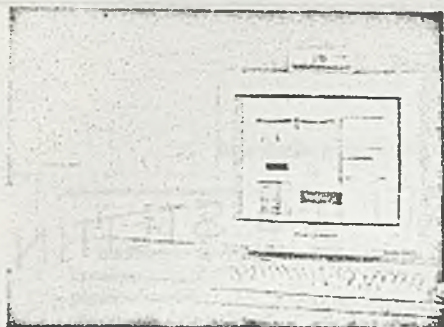
- przeciwpożarowe
- ochrony środowiska
- ochrony antykorozyjnej statków i konstrukcji morskich

### CTO świadczy usługi ogólnotechniczne w zakresie:

- informacji naukowo-technicznej i ekonomicznej
- udostępnienia zbiorów norm, przepisów i konwencji
- sprzedaży norm Polskiego Komitetu Normalizacji







**Centrum Techniki Okrętowej** jest liderem zaplecza badawczo-rozwojowego w zastosowaniach informatyki do prac badawczo-rozwojowych i projektowych wykonywanych na rzecz przemysłu okrętowego.

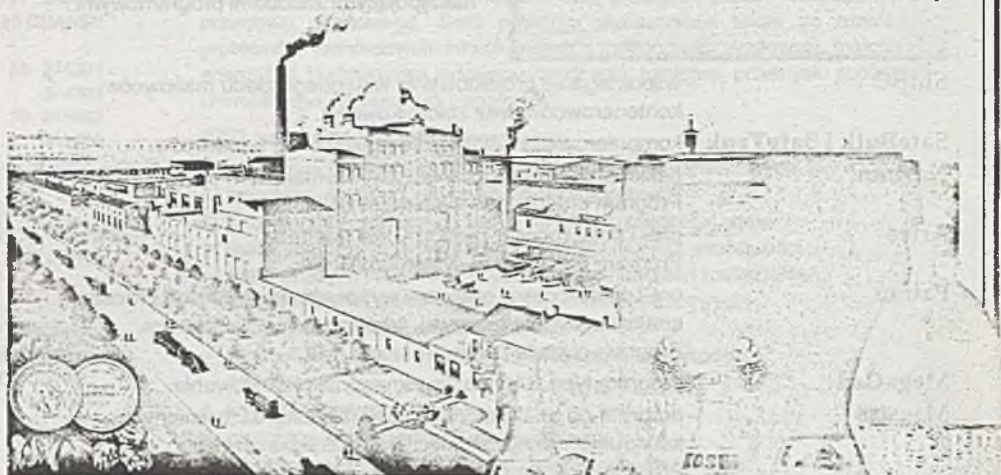
**Centrum Techniki Okrętowej** dysponuje następującymi zasobami programowymi:

- ShipC** - wspomaganie projektowania wstępnego zładu masowców, kontenerowców oraz zbiornikowców.
  - SafeBulk i SafeTank** - komputerowe projektowanie masowców i tankowców.
  - Nastran** - pakiet programów do analizy wytrzymałości w pełnym zakresie. Program pracuje pod systemem UNIX.
  - Aries** - pre- i postprocesor graficzny dla Nastrana. Program pracuje pod systemem UNIX.
  - Patran** - pre i postprocesor graficzny wyposażony we własne procedury analizy wytrzymałościowej, współpracuje z Nastranem. Program pracuje pod systemem UNIX.
  - MegaCad** - program typu CAD do wspomagania projektowania.
  - Maestro** - program do analizy, oceny i optymalizacji usztywnionych konstrukcji cienkościennych typu kadłuba okrętu. Program pracuje pod systemem UNIX.
  - PamPTS, MonitorMTS** - akwizycja danych pomiarowych z maszyny wytrzymałościowej poprzez kartę przetwornika analogowo-cyfrowego "AMBEX", zautomatyzowane przetwarzanie wyników, tworzenie wykresów i raportów.
  - GRAPHCOR** - program do wyznaczania parametrów korozyjnych.
  - PDS** - komputerowy system informacji o materiałach do budowy statków.
  - NAPA** - program do wstępnego projektowania statków w zakresie projektowania o odwzorowanie kształtu kadłuba i obliczeń hydrostatycznych. Program pracuje pod systemem UNIX.
  - BOS-L** - program do numerycznego wyznaczania opływu statku wraz z rozkładem falowym.
  - SYSTEM** - zintegrowany system informatyczny do wspomagania zarządzania
- Ponadto w CTO jest użytkownikiem Trójmiejskiej Akademickiej Sieci Komputerowej (TASK), umożliwiającą korzystanie z usług i zasobów sieci INTERNET.

Ośrodki CTO działają w systemie jakości spełniającym wymagania normy PN-EN 45001 dla laboratoriów badawczych, będącej odpowiednikiem norm serii ISO 9000 dla zakładów produkcyjnych. Podstawowe usługi pomiarowe i badawcze CTO objęte są certyfikatami akredytacyjnymi wydanymi przez Polskie Centrum Badań i Certyfikacji.



# BOHRISCH BRAUEREI A.G. STETTIN



BROWAR SZCZECIN



Browar Szczecin S.A. dysponuje najnowocześniejszymi technicznymi - technologicznymi rozwiązaniami gwarantującymi wytwarzanie piwa o wysokiej jakości i maksymalnej czystości mikrobiologicznej. Ciąg technologiczny jest zautomatyzowany i sterowany za pomocą komputerów, co pozwala na kompleksową kontrolę produkcji i wyeliminowanie ingerencji niepożądanych czynników zewnętrznych. W pełni zautomatyzowana stacja mycia (CIP) gwarantuje sterylną czystość piwa. Od czerwca '93 stosowana jest potrójna filtracja piwa; wirówka - filtr na ziemię okrzemkową - filtr klarujący (kartonowy). Używane do produkcji piwa surowce: słód, chmiel, drożdże i woda (wydobytą z własnych źródeł osadów czwartorzędowych) są najwyższej jakości. Dlatego piwo Browaru Szczecin odznacza się wysokimi wartościami smakowymi, jest dobrze nasycone, z wyraźną chmielową goryczką, o trwałym bukiecie.

Barwa jasna, słomkowa - wynosi 6 - 8 jednostek EBC.

Klarowność bardzo dobra - piwo ma tzw. „połysk”

Piana jest gęsta, biała, trwała.

Zapach ułciwicy, czysty i typowy dla piwa jasnego - wyraźnie chmielowy.

Parametry fizyko-chemiczne, a przede wszystkim stężenie brzojski podstawowej, charakteryzującej rodzaj piwa i zawartość alkoholu doskonale mieści się w obowiązujących normach. Poszczególne gatunki piwa różnią się zawartością alkoholu oraz intensywnością goryczki chmielowej.

# PTI-L

## *lista dyskusyjna Polskiego Towarzystwa Informatycznego*

---

Informatyka jest dziedziną aktywności ludzkiej, która wymaga od jej uczestników permanentnego samokształcenia. Gałęzie nauki i techniki tworzące informatykę, rozwija się z prędkością, skutecznie utrudniającą pogodzenie nauki własnej z pracą zawodową. Pojawienie się medium komunikacji, które dostarcza wyselekcjonowany materiał może okazać się bezcenne dla rozwoju zawodowego. Bazowa technika podnoszenia własnych kwalifikacji, w bardzo wielkim uproszczeniu, sprowadza się do obserwacji (w wymiarze zawodowym) mas mediów. Nic jednak nie zastąpi kontaktu ze specjalistą. Konferencje, seminaria, sympozja, szkolenia, to imprezy o największym ciężarze gatunkowym w zakresie zdobywania wiedzy. Jest jednak pewne "ale". Najczęściej nie ma czasu na odwiedzenie wszystkich interesujących spotkań, a poza tym w ogóle trzeba wiedzieć, że takie spotkania się odbywają, że o kwestii funduszy na wyjazd nie wspomnę.

Jedną z form integracji zawodowej jest działalność stowarzyszeń informatycznych. Polskie Towarzystwo Informatyczne jest stowarzyszeniem o najdłuższej historii (powstało w burzliwym okresie wiosny 1981 roku). W PTI skupili się i do dzisiaj działają najaktywniejsi krajowi informatycy, którzy tworzyli i tworzą historię jej rozwoju w Polsce. Powołana do życia lista dyskusyjna to kolejny element historii polskiej informatyki i Towarzystwa. Od ponad 6 lat istnieje sieć EARN/BITNET, trochę krócej mamy dostęp do Internetu. Te względnie nowe środki transferu informacji pozwalają na znacznie szybszą ich wymianę. Czyżby realizacja wizji świata, jako globalnej wioski - niespełna 55 milionów korespondentów w Internecie to obraz, który może zrobić wrażenie. Od początku istnienia tych sieci, ich użytkownicy, prócz możliwości wysyłania do siebie nawzajem listów (w tej, czy innej formie) łączą się w grupy, wiedzeni chęcią wymiany poglądów na interesujące ich tematy. W każdej sieci jest co najmniej jedno takie narzędzie, które pozwala na automatyczną dystrybucję poczty do określonych grup korespondentów. W Internecie jest to np. USENET, w EARN/BITNET listy dyskusyjne obsługiwane przez jedno z najsilniejszych tego typu narzędzi, jakim jest LISTSERV. Ogrom różnych pomysłów na tematykę list dyskusyjnych niech zobrazuje fakt, i w węzle krajowym: EARN/BITNET, czyli PLEARN znanych jest ich ponad 5000. Polskie Towarzystwo Informatyczne wyszło z inicjatywą wspomóc swoich członków, ale nie tylko, także wszystkich zainteresowanych tematyką szeroko rozumianej informatyki. Od lipca 1994 roku działa PTI-L: listę dyskusyjną Polskiego Towarzystwa Informatycznego. PTI-L funkcjonuje w popularnej nie tylko w kraju, akademickiej sieci EARN/BITNET (dostępna oczywiście w Internecie). Lista dyskusyjna PTI tworzy nową jakość w działaniu Towarzystwa. Jak będzie działała, zależy to od jej uczestników.

PTI bardzo serdecznie zaprasza wszystkich chętnych do współtworzenia listy. Pamiętać bowiem należy, że cokolwiek dzieje się wokół nas, my o tym wiemy, ale może warto by wiedzieli także inni, bo jest to informacja, na którą czekają - to hasło, a może raczej idea, która przyświecała uruchomieniu PTI-L.

Leszek Bogusławski (administrator listy)  
e-mail: jerome@warman.com.pl  
Zdzisław Szyjewski  
e-mail: zszyjew@uoo.univ.szczecin.pl



3214254

0-90

## PTI-L

### lista dyskusyjna Polskiego Towarzystwa Informatycznego

---

Adres list: PTI-L@PLEARN.BITNET

na powyższy adres wysyłamy pocztę, którą chcemy przekazać innym prenumeratorom listy.

#### *Jak się zapisać?*

1. Dla użytkowników systemu MVS na maszynie IBM (EARN) poprzez wydanie komendy

TELL LISTSERV@PLEARN.BITNET SUBSCRIBE PTI-L Imię Nazwisko  
<firma>

przykład:

1. tell listserv@plearn.bitnet subscribe pti-l Leszek Bogusławski
2. tell listserv@plearn.bitnet subscribe pti-l Leszek Bogusławski Polskie Towarzystwo Informatyczne

2. Dla użytkowników internetu poprzez wysłanie na adres

LISTSERV@PLEARN.EDU.PL

listu (bez tematu) z jedną linią zawierającą polecenie:

SUBSCRIBE PTI-L Imię Nazwisko < firma>





