

# **Integracja systemów w praktyce uczelni wyższej**

Janina Mincer-Daszkiewicz

[jmd@mimuw.edu.pl](mailto:jmd@mimuw.edu.pl)

CPI, Warszawa, 15.03.2011

# MUCI i USOS

- JMD

Instytut Informatyki Uniwersytetu Warszawskiego, Lider projektu USOS z ramienia MUCI

- MUCI

**Międzyuczelniane Centrum Informatyzacji;** konsorcjum 35 polskich uczelni

<http://muci.edu.pl>

- USOS

**Uniwersytecki System Obsługi Studiów;** system do obsługi spraw studiów, własność MUCI

<http://usos.edu.pl>



**Użytkownicy USOS**

# Plan prezentacji

- **Usługi sieciowe** czyli jak pobierać wyniki matur do systemu rekrutacji na studia
- **Hurtownia danych do celów zarządczych**, czyli jak łączyć dane z systemu rekrutacyjnego i bazy studentów
- **Automatyczna synchronizacja danych** czyli jak wymieniać dane między bazą centralną i bazami aplikacji webowych
- **Korporacyjna szyna danych** czyli jak uniknąć dublowania osób w systemach uczelnianych
- **UCZELNIA-API** czyli jak dać dostęp do danych w systemie uczelnianym (autoryzowanym) aplikacjom zewnętrznym

# Krajowy Rejestr Matur – KReM

- Dostępny pod adresem: <https://krem.uw.edu.pl/>
- Powstał w wyniku umowy między rektorem **Uniwersytetu Warszawskiego** a dyrektorami ośmiu **Okręgowych Komisji Egzaminacyjnych**
- Od **2006** roku gromadzi wyniki matur kandydatów na studia
- Wielką wagę przykładą się do zapewnienia **bezpieczeństwa**
  - Szyfrowana komunikacja za pomocą protokołu HTTPS
  - Certyfikat SSL systemu KReM, który użytkownik musi zweryfikować podczas pierwszego połączenia ze stroną
  - Podpisy elektroniczne do podpisywania danych (OpenPGP)
  - Uwierzytelnianie użytkowników za pomocą identyfikatorów i haseł
  - Logowanie do systemu tylko z zarejestrowanych komputerów (filtrowanie po numerze IP)
  - Logowanie wszystkich zdarzeń

## Statystyki systemu KReM

Aktywne Uczelnie:	92
Aktywne OKE:	8
KReM i CKE:	2
Olimpiady przedmiotowe:	20
Arkusze egzaminacyjne:	121
Szkoły w słowniku:	10286
Administratorzy aktywni/wszyscy:	167 / 225 (74.22%)
Administratorzy nieaktywni/wszyscy:	51 / 225 (22.67%)
Administratorzy usunięci/wszyscy:	7 / 225 (3.11%)
Maturzyści z ocenami/wszyscy:	1685672 / 1690805 (99.7%)
Maturzyści z dokumentami/wszyscy:	1406956 / 1690805 (83.21%)
Maturzyści z osiągnięciami/wszyscy:	4478 / 1690805 (0.26%)
Maturzyści ze szkołą/wszyscy:	1028292 / 1690805 (60.82%)
Maturzyści, którzy wycofali zgodę:	5129
Pliki pobrane z systemu:	66238
Pliki przekazane do systemu:	65860
Zapytania anulowane/wszystkie:	9045638 / 10703050 (84.51%)
Zapytania nieanulowane/wszystkie:	1657412 / 10703050 (15.49%)
Zapytania gotowe do pobrania/nieanulowane:	311903 / 1657412 (18.82%)
Zapytania nie gotowe do pobrania/nieanulowane:	1345509 / 1657412 (81.18%)
Zapytania pobrane/nieanulowane:	778392 / 1657412 (46.96%)
Zapytania nie pobrane/nieanulowane:	879020 / 1657412 (53.04%)
Zapytania posiadające dane/nieanulowane:	1090037 / 1657412 (65.77%)
Zapytania do aktualizacji/nieanulowane:	258 / 1657412 (0.02%)

# Liczba maturzystów, których dane trafiły do KReM w roku 2010

Rok:

<i>Nazwa</i>	<i>Matura z tego roku</i>	<i>Aneks z tego roku</i>	<i>Pierwsze dane w tym roku</i>	<i>Konflikty rozstrzygnięte</i>	<i>Konflikty nierozstrzygnięte</i>
OKE w Gdańsku	31888	2063	31996	0	0
OKE w Jaworznie	36846	2100	36899	0	0
OKE w Krakowie	70366	4009	70505	0	0
OKE w Łodzi	31840	1880	31893	0	0
OKE w Łomży	23245	1430	23310	0	0
OKE w Poznaniu	49262	2713	49385	0	0
OKE w Warszawie	40215	3286	40325	0	0
OKE we Wrocławiu	28508	1852	28569	0	0
	312170	19333	312882		

# Ilość danych otrzymanych z OKE w roku 2010

Od: 2010-01-01 do: 2010-12-30

Szukaj

Nazwa	Wysłane wyniki	Wysłane dokumenty	Wysłane osiągnięcia	Wysłane szkoły	Wysłane wycofania	Wysłane pliki
OKE w Gdańsku	33976	33978	163	33977	35	59
OKE w Jaworznie	38969	38970	113	41627	62	69
OKE w Krakowie	74407	74382	278	74383	205	64
OKE w Łodzi	96322	96980	280	96551	1	67
OKE w Łomży	24691	24675	119	24666	0	36
OKE w Poznaniu	52007	52008	158	47838	46	161
OKE w Warszawie	43530	43530	149	43498	56	30
OKE we Wrocławiu	428043	284715	749	228189	106	49



- Początkowo przekazywanie danych między OKE I KReM oraz KReM i uczelniami odbywało się poprzez **pliki**
- Na sugestie uczelni wprowadzono możliwość przekazywania danych poprzez wywołania **usług sieciowych** (ang. *web services*)
- KReM pełni rolę **serwera** udostępniającego około **80 usług sieciowych**
- Rolę klienta pełnią uczelniane systemy typu **IRK (Internetowa Rejestracja Kandydatów)**
- Na stronie informacyjnej KReM są dostępne:
  - Dokumentacja techniczna usług sieciowych KReM
  - Plik **WSDL** ze specyfikacją interfejsu usług sieciowych
  - Przykładowy klient w **Javie** i w **PHP**
- Zastosowane technologie: SOAP, WS-Security, SSL, OpenPGP

# KReM – użytkownicy usług sieciowych

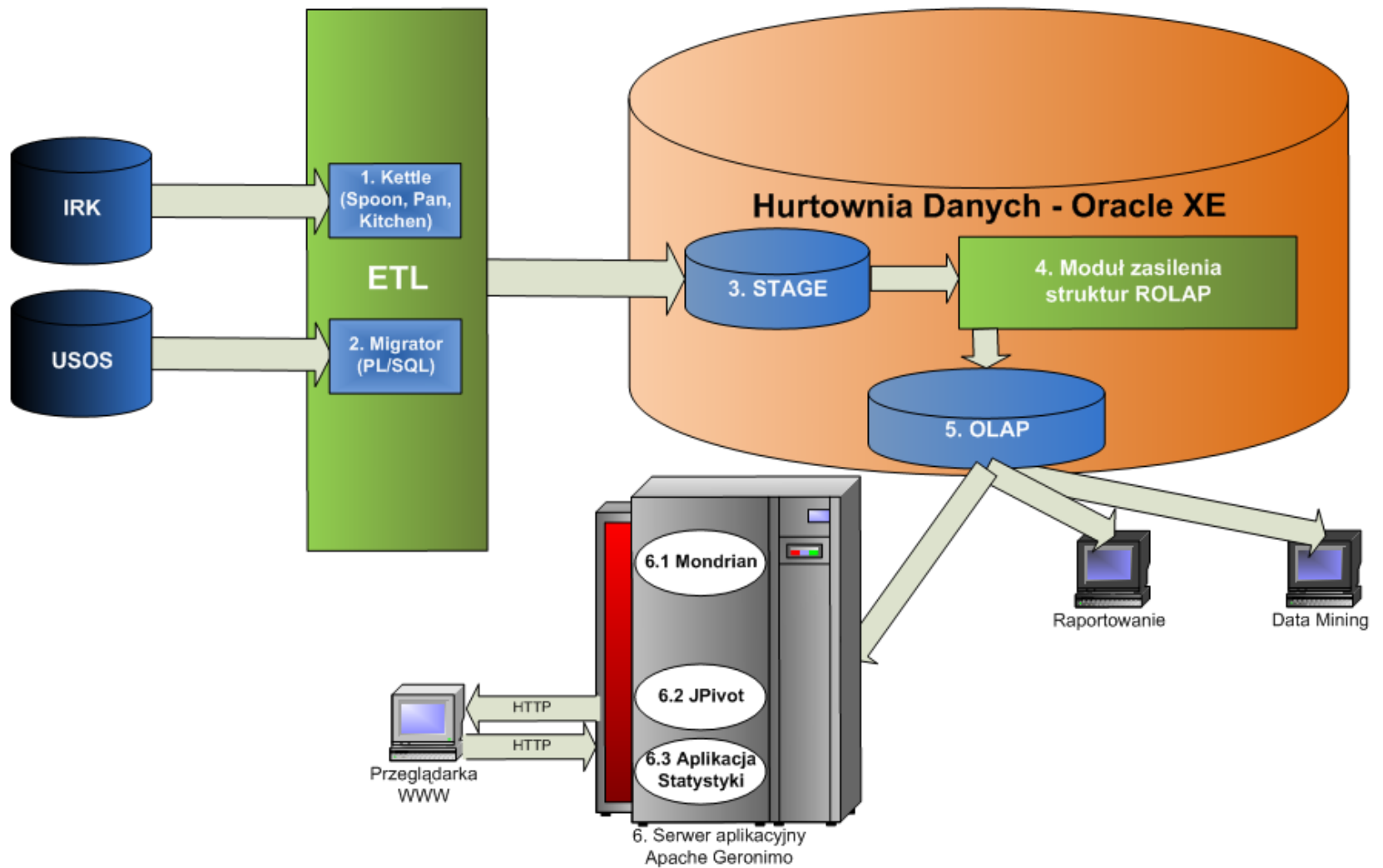
- Spośród **90 uczelni** zarejestrowanych w KReM z usług sieciowych korzysta **18** (do roku 2010):

1. Akademia Ekonomiczna im. Karola Adamieckiego w Katowicach
2. Akademia Górniczo-Hutnicza w Krakowie
3. Akademia Techniczno-Humanistyczna w Bielsku-Białej
4. Akademia Wychowania Fizycznego i Sportu w Gdańsku
5. Państwowa Wyższa Szkoła Informatyki i Przedsiębiorczości w Łomży
6. Państwowa Wyższa Szkoła Zawodowa w Białej Podlaskiej
7. Państwowa Wyższa Szkoła Zawodowa w Gnieźnie
8. Państwowa Wyższa Szkoła Zawodowa im. Jana Amosa Komeńskiego w Lesznie
9. Państwowa Wyższa Szkoła Zawodowa w Raciborzu
10. Politechnika Częstochowska
11. Politechnika Lubelska
12. Politechnika Poznańska
13. Politechnika Rzeszowska im I. Łukaszczyka
14. Uniwersytet Jagielloński
15. Uniwersytet Kardynała Stefana Wyszyńskiego
16. Uniwersytet Technologiczno-Przyrodniczy w Bydgoszczy
17. Warszawski Uniwersytet Medyczny
18. Wyższa Szkoła Filozoficzno-Pedagogiczna "Ignatianum" w Krakowie

- Spośród **8 OKE** z usług sieciowych nie korzysta żadna

# Hurtownia danych – procesy ETL

- **Procesy zarządcze na uczelni wyższej** wymagają wsparcia informatycznego. Do tego służą hurtownie danych.
- Przykład: hurtownia danych z własną bazą danych (Oracle Express) zasilana **z dwóch źródeł**:
  - **System rekrutacyjny** (IRK, baza danych MySQL)
  - **System uczelniany** do obsługi spraw studiów (USOS, baza danych Oracle)
- Zasilanie hurtowni danymi (**moduł ETL**):
  - Transformacje między tabelami obu baz za pomocą narzędzia Pentaho Data Integration (Kettle)
  - Jednokierunkowy migrator danych, skrypt (w PL/SQL) wykonywany po stronie bazy docelowej, dostęp do bazy źródłowej za pomocą **Oracle db\_link**
- Technologie: Mondrian jako serwer ROLAP, baza danych Oracle Express, język zapytań MDX, narzędzie ETL Pentaho Data Integration, JPivot



Hurtownia danych zasilana z dwóch źródeł danych

(źródło: praca mgr G. Kukawskiego, UW 2008)

# Lista kategorii i statystyk

The screenshot shows the USOS STATYSTYKI interface. At the top, there is a blue header with the text "USOS STATYSTYKI" and a "pomoc" link. Below the header, there are three navigation tabs: "KATALOG", "MODEL DANYCH", and "WYLOGUJ". On the left side, there are two main menu boxes. The first box is labeled "Katalog" and contains a search bar with a "/" symbol. The second box is labeled "Menu" and contains the options "Dodaj kategorię" and "Dodaj statystykę". On the right side, there is a list of categories and statistics. The categories are represented by yellow folder icons and include "Studenci", "Pracownicy", "Dyplomy", "Rekrutacja", "Maturzyści", and "Przyjęci". A red line points from the text "Kategoria" to the "Maturzyści" folder. Below the categories, there is a bar chart icon representing a statistic, with a red line pointing from the text "Statystyka" to it. The statistic is titled "Liczba pracowników w podziale na stanowiska w poszczególnych latach".

# Dodawanie statystyk

The screenshot shows the USOS STATYSTYKI interface with the "Dyplomy" category selected. The header and navigation tabs are the same as in the previous screenshot. The left side menu now shows the "Dyplomy" category selected, with a description: "Kategoria zawiera statystyki dotyczące dyplomów wydanych przez uczelnie". Below it, the "Menu" box contains the options "Dodaj kategorię", "Modyfikuj kategorię", "Dodaj statystykę" (circled in red), and "Wyszukaj statystykę". On the right side, there is a list of statistics, each represented by a bar chart icon. The statistics are: "Liczba wydanych dyplomów w podziale na realizowany program w kolejnych latach", "Liczba wydanych dyplomów z wyróżnieniem i bez, wraz z wynikiem studiów w podziale na lata", "Oceny otrzymane z egzaminu dyplomowego w podziale na lata", "Średnie ze studiów studentów, którzy otrzymali dyplom w podziale na lata", "Oceny z prac dyplomowych w podziale na lata", and "Procentowy udział liczby wydanych dyplomów w podziale na typ w poszczególnych latach".

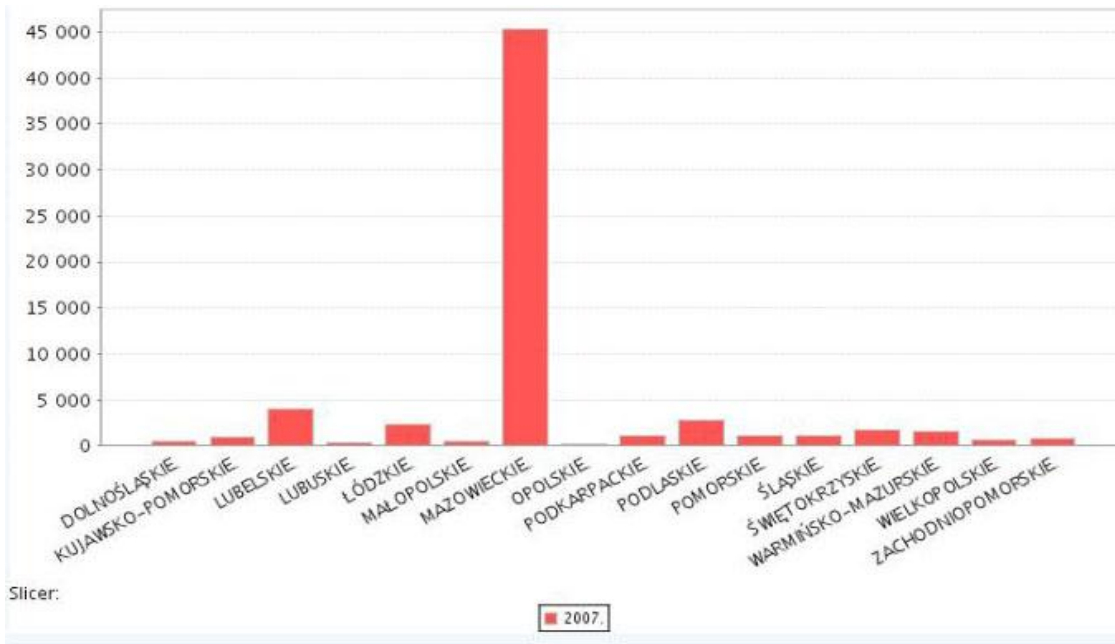
(źródło: praca mgr G. Kukawskiego, UW 2008)

	Data danych	
	2007	
	Measures	
Program	• Liczba mężczyzn	• Liczba kandydatów
+Niestacjonarne - eksternistyczne	130	626
+Niestacjonarne - wieczorowe	2270	5815
+Niestacjonarne - zaoczne	1406	5171
-Stacjonarne	15995	45476
-DZ-INF	784	906
Informatyka, stacjonarne, pierwszego stopnia	784	906
+DZ-JSEM	255	447
+DZ-KU-ESW	193	743
+DZ-KU-IB	123	919
+DZ-KU-JA	132	417
+DZ-KU-OBTA	54	319
+DZ-KU-SL	143	595
+DZ-KU-WK	121	633
+DZ-MAT	358	598
+DZ-MSOS	79	216
+DZ-MZ	25	74
+DZ-OS	60	175
+DZ-PE	143	1443
+DZ-PO-IPSIR	240	714
+DZ-SC	80	166

## Popularność kierunków z liczbą kandydatów mężczyzn

(źródło: praca mgr G. Kukawskiego, UW 2008)

## Liczba kandydatów z poszczególnych województw

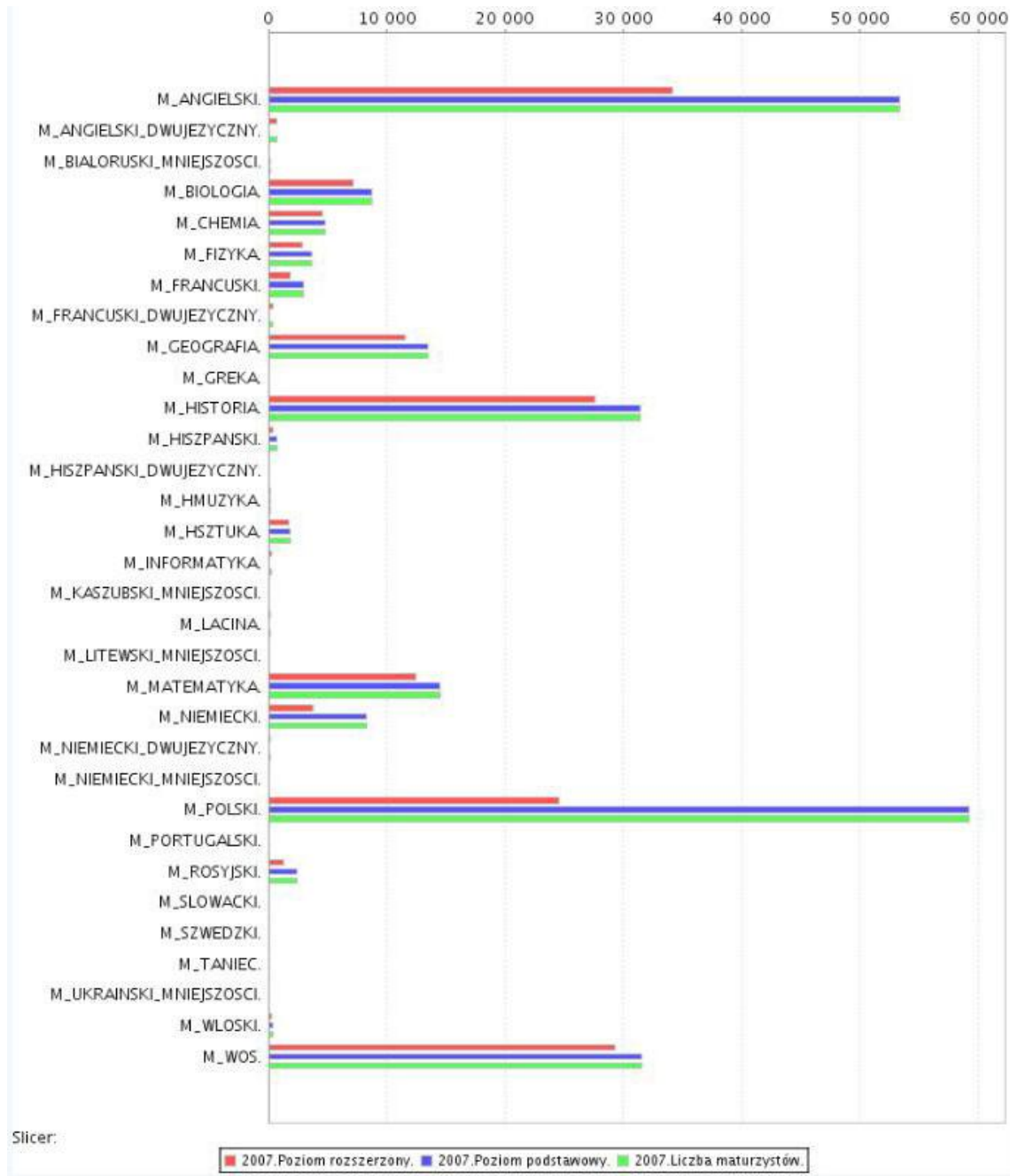


Geografia	Data danych
+DOLNOŚLĄSKIE	514
+KUJAWSKO-POMORSKIE	958
+LUBELSKIE	3998
+LUBUSKIE	292
+ŁÓDZKIE	2405
+MAŁOPOLSKIE	431
+MAZOWIECKIE	45265
+OPOLSKIE	196
+PODKARPACKIE	1111
+PODLASKIE	2851
+POMORSKIE	1118
+ŚLĄSKIE	1136
+ŚWIĘTOKRZYSKIE	1692
+WARMIŃSKO-MAZURSKIE	1631
+WIELKOPOLSKIE	715
+ZACHODNIOPOMORSKIE	762

(źródło: praca mgr G. Kukawskiego, UW 2008)



# Popularność przedmiotów maturalnych



(źródło: praca mgr G. Kukawskiego, UW 2008)

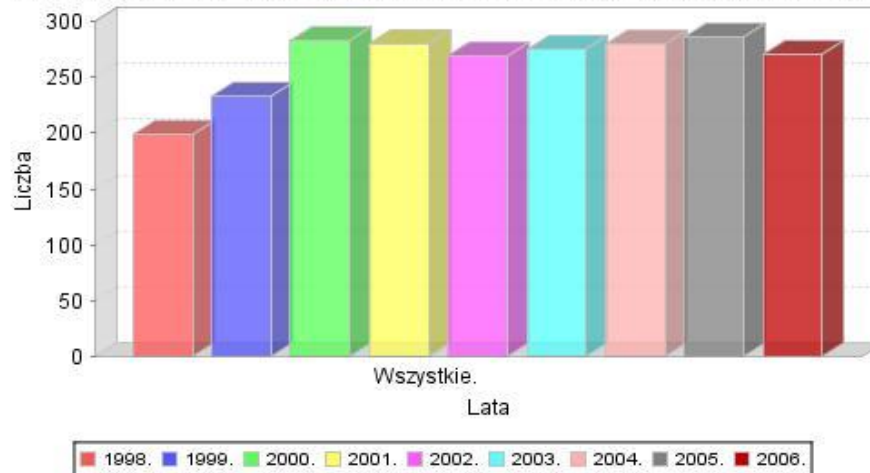


# Liczba pracowników Wydziału MIM

## Liczba pracowników w podziale na stanowiska w poszczególnych latach

Liczba pracowników w podziale na stanowiska w poszczególnych latach

### Liczba pracowników na MIM w poszczególnych latach



Stanowisko	Data danych								
	1998	1999	2000	2001	2002	2003	2004	2005	2006
Wszystkie	199	233	283	281	270	275	280	286	271

[Jednostka=Wydział Matematyki, Informatyki i Mechaniki]

- **Systemy uczelniane** obejmują często wiele różnych aplikacji:
  - System internetowej rekrutacji kandydatów
  - System typu dziekanat dla uczelnianej administracji
  - Aplikacje dla studentów i pracowników (wirtualny dziekanat)
  - Informator o studiach dla kandydatów na studia, studentów z zagranicy i z innych uczelni (informator ECTS)
  - Platforma do zdalnego nauczania (e-learning)
  - Programy pocztowe, system biblioteczny
  - Portale uczelni i jej jednostek
  - Centralny System Uwierzytelniania z pojedynczym logowaniem
- Z różnych względów (skalowalność, różni dostawcy, historyczne, organizacyjne) te systemy często mają **własne lokalne bazy danych**

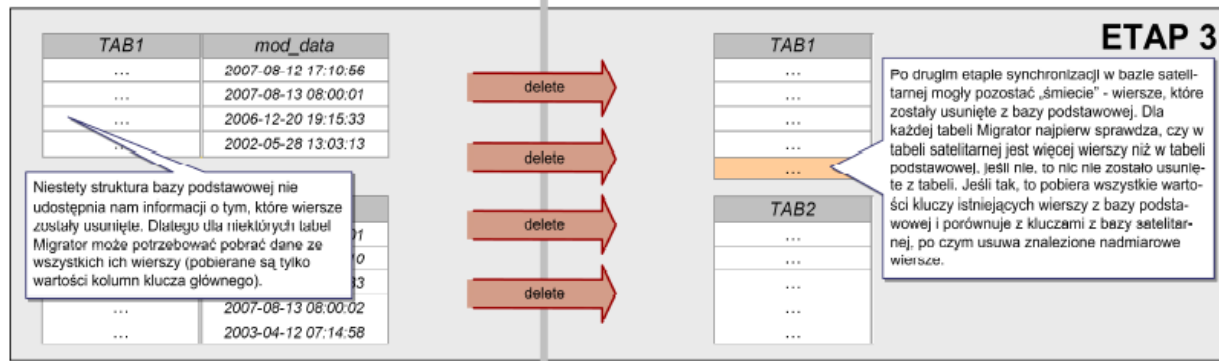
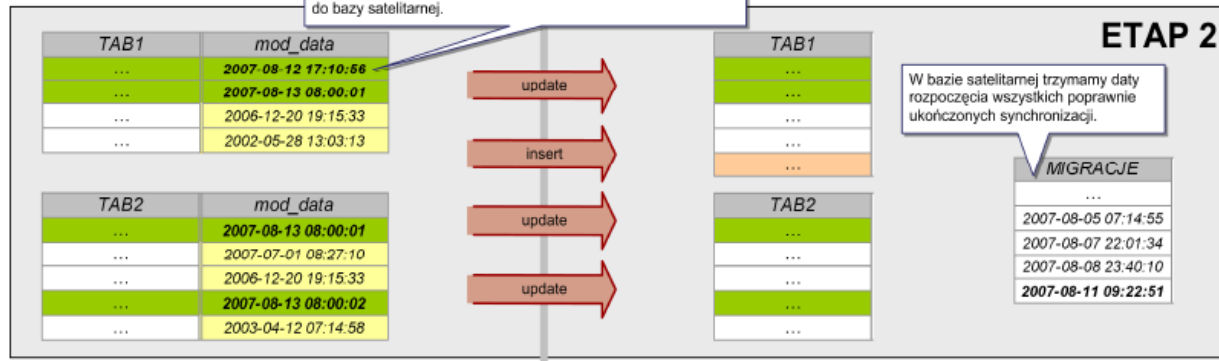
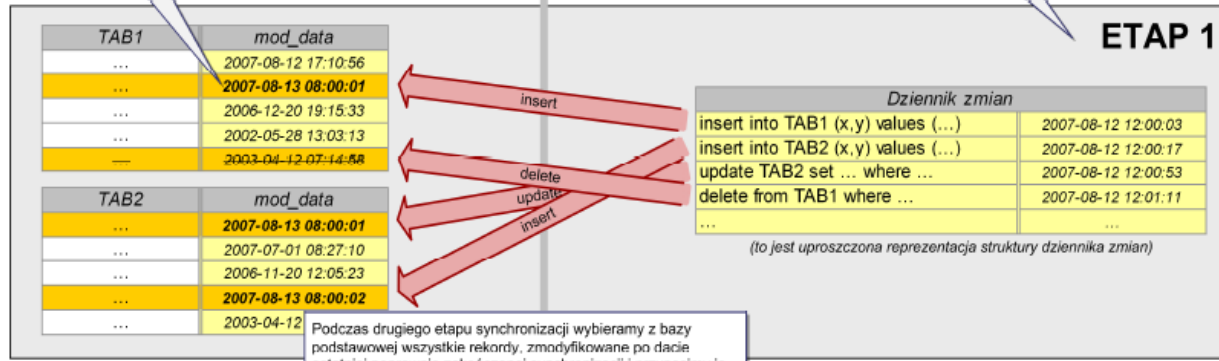
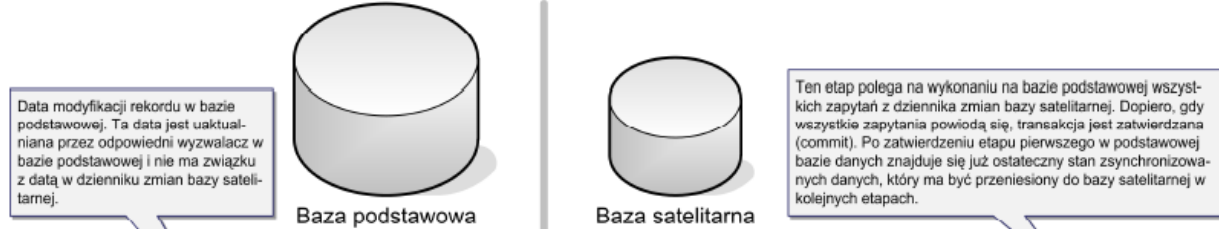
# Przykład – Uniwersytet Warszawski

- Zachowanie spójności i poprawności danych wymaga, aby:
  - dane były wprowadzane **tylko raz**, w jednym miejscu ...
  - i automatycznie **dostarczane** wszędzie tam, gdzie są potrzebne
- Systemy i bazy danych na **Uniwersytecie Warszawskim**
  - IRK, IRK-BWZ, IRK-SJO (baza MySQL)
  - USOS (baza Oracle)
  - USOSweb, UL, APD, Ankieter (baza MySQL)
  - Informator ECTS (baza MySQL)
  - Moodle (baza MySQL)
  - CAS + LDAP (baza MySQL)
  - BUW (system komercyjny Virtua)
  - Portale (różnie), CMS-y (Drupal, Joomla) i wiele innych
  - Biurokarier.edu.pl (wspólna aplikacja wiele uczelni)

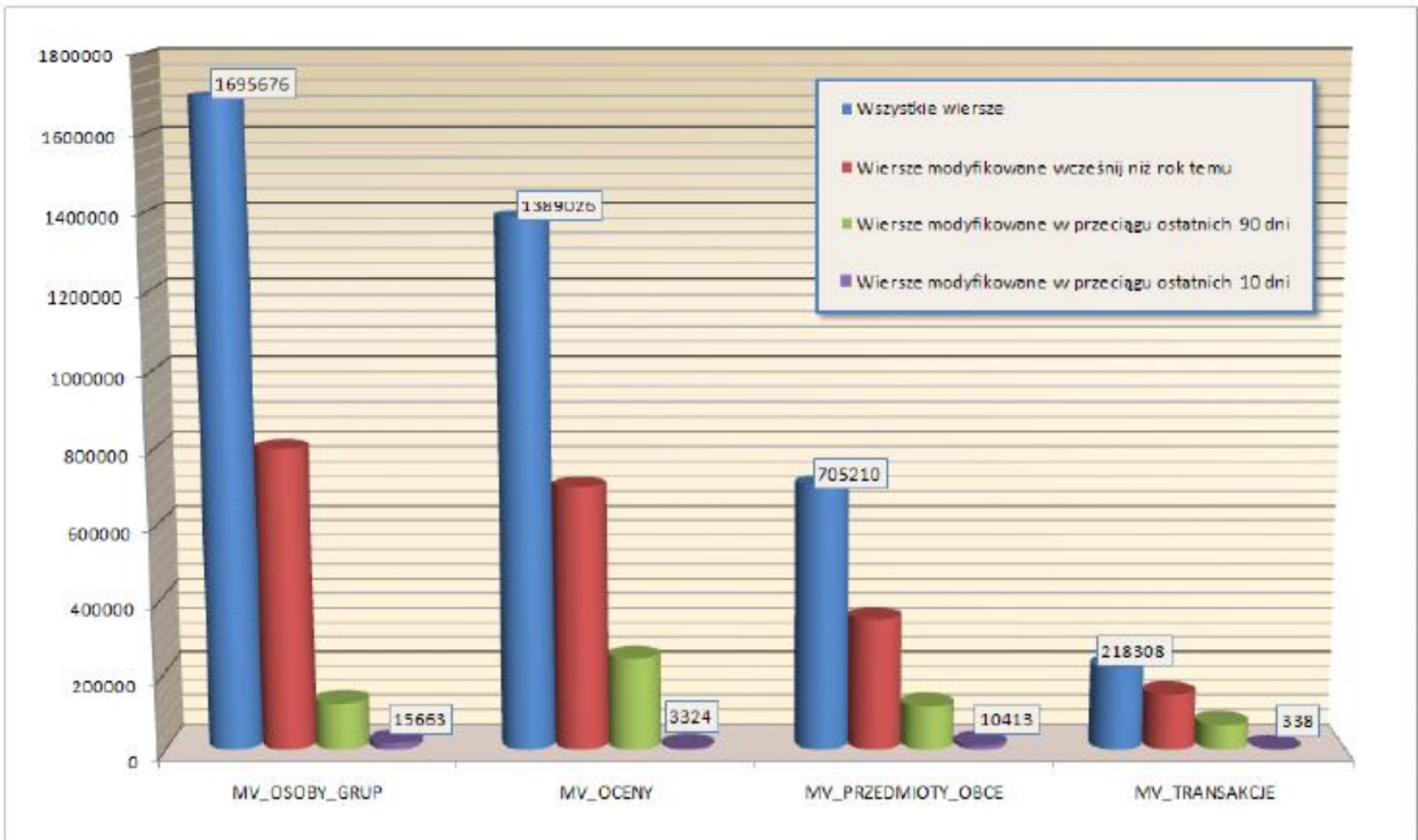


# Automatyczna synchronizacja danych

- Rozwiązanie problemu → **narzędzie do automatycznej dwukierunkowej replikacji danych**
  - Napisane w Pythonie
  - Wymaga sterowników do obsługi wspieranych baz danych (obecnie: Oracle, MySQL)
  - Migruje dane między bazą centralną i satelitarną na podstawie pliku konfiguracyjnego, który opisuje mapowanie tabel (widoków) z obu baz
  - Jeśli dane mają być przenoszone w obu kierunkach, to po stronie bazy satelitarnej wymagane jest logowanie operacji, których efekty mają być przeniesione do bazy centralnej
  - Działa przyrostowo, czyli sprawdza dane zmienione od poprzedniej migracji
  - Wszystkie wykonywane operacje są szczegółowo logowane



# Etapy synchronizacji danych



Wpływ daty modyfikacji na liczbę migrowanych wierszy podczas drugiego etapu synchronizacji

(źródło: praca mgr W. Rygielskiego, UW 2007)



- **Kierunki** migracji danych
  - Z IRK i USOS do Informatora ECTS
  - Między USOS a USOSweb, UL, APD, Ankieterem, Moodle
  - Z USOS do CAS + LDAP, BUW, portali
  - Z baz danych USOS różnych uczelni do bazy danych biurokarier.edu.pl
- **Szybkość** działania:
  - między USOS a USOSweb ok. 15-20 minut
  - między USOS a APD, BUW, BK ok. 5 minut
  - między USOS a LDAP, COME ok. 1-2 minut
- Bardzo ogólne narzędzie, może synchronizować **dowolne bazy danych** (pod warunkiem dostępności sterownika i pliku konfiguracyjnego definiującego mapowanie)



# Wiele źródeł danych a unikatowość danych osobowych

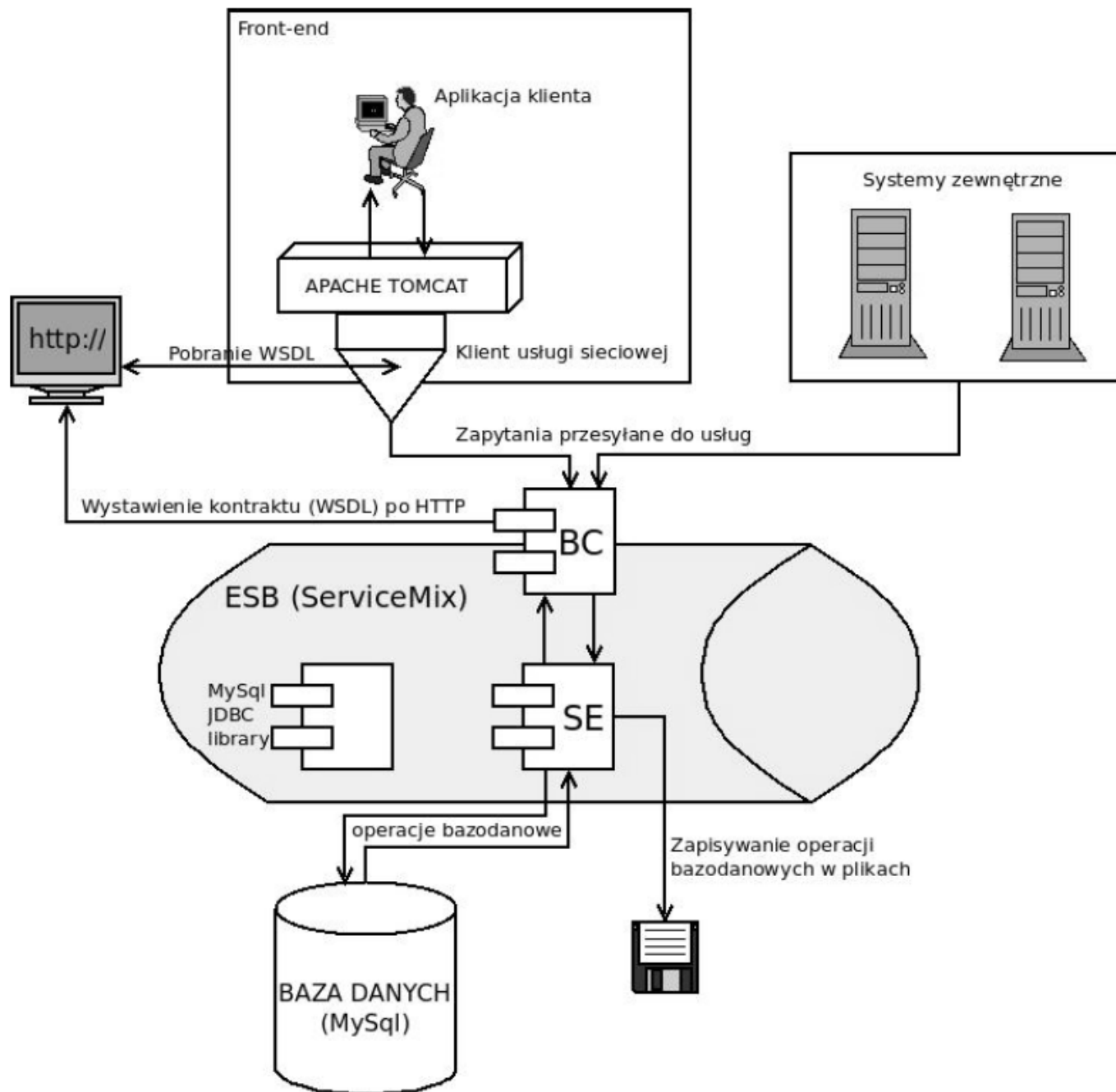
- Przykładem danych, dla których nie ma jednego naturalnego miejsca wprowadzania danych są **dane osobowe**:
  - Pracownicy są wprowadzani do **bazy kadrowej** (HMS)
  - Studenci są wprowadzani do **bazy studenckiej** (USOS)
- Studenci stają się pracownikami, pracownicy zaczynają studiować , brak naturalnych kluczy (obcokrajowcy) ...
- Osoba może występować w **wielu rolach**, ale dane osobowe powinny być **unikatowe** i jednoznacznie identyfikowane
- Rozwiązanie problemu → **Centralny Rejestr Osób (CRO)** nadający nowym osobom unikatowy **GUID** i automatycznie propagujący ich dane do wszystkich aplikacji
- Technologia – **korporacyjna szyna danych (ESB)** pełniąca rolę serwera aplikacyjnego, rutera komunikatów, szyny przekazującej dane do subskrybentów, konwertera formatów danych

# Centralny Rejestr Osób (CRO)

- **Zadania CRO:**
  - Dodawanie danych osobowych do centralnego rejestru wraz z nadaniem globalnego identyfikatora unikatowego w skali wszystkich uczelnianych baz danych
  - Umożliwienie wyszukiwania danych wg różnych kryteriów, na podstawie niepełnych danych
  - Propagowanie zmian do innych systemów
- **Minimalny zestaw danych** umożliwiający umieszczenie wpisu w bazie:
  - GUID
  - Imię, nazwisko, PESEL
  - Imię, nazwisko, data urodzenia, płeć
  - Imię, nazwisko, data urodzenia, miejsce urodzenia, kraj urodzenia
  - Imię, nazwisko, typ dokumentu, numer dokumentu , kraj wydania
  - Imię, nazwisko, nazwa aplikacji uczelnianej, identyfikator osoby w tej aplikacji

# Centralny Rejestr Osób (CRO)

- Dane **unikatowe** w ramach CRO:
  - GUID
  - PESEL
  - (Typ dokumentu, kraj wydania, numer dokumentu)
  - (Aplikacja uczelniana, identyfikator osoby w tej aplikacji)
- **WSDL** opisuje dopuszczalne operacje:
  - getData (SELECT) – umożliwia wyszukiwanie danych, dostępne dwie opcje:
    - Czy pole może być puste?
    - Czy pole musi odpowiadać dokładnie wartości przekazanego parametru?
  - setData (UPDATE)
  - setGUID (INSERT)
- Inne aplikacje powinny subskrybować zmiany
- Zastosowane technologie: szyna ServiceMix, Apache CXF, Apache Maven
- Inne zastosowania szyny: centralne generowanie indywidualnego numeru konta bankowego dla studenta



Architektura aplikacji CRO osadzonej na szynie danych

(źródło: praca mgr K. Eisenbarta, UW 2011)

# Dostęp do danych w repozytoriach uczelni

- Repozytoria uczelniane przechowują wiele użytecznych danych
- Sięgają do nich serwisy uczelniane, ale można by je także udostępnić zewnętrznym aplikacjom
- W wielu serwisach, takich jak Gmail, Facebook, czy Twitter, można dać aplikacji uprawnienie do korzystania z danych zgromadzonych w innym systemie. Wymaga to jednak zgody użytkownika.
- Według tych samych zasad działa **USOS-API**, program napisany w Pythonie, udostępniający dane z bazy USOS poprzez uzgodnione API
- Może pełnić rolę serwera aplikacyjnego dla wewnętrznych serwisów uczelnianych i dostawcy danych dla zewnętrznych programów
- USOS-API to **platforma integrująca** systemy informatyczne uczelni ze światem zewnętrznym



- **Introduction**
  - [Authorization](#)
  - [Working with API](#)
    - [Basic rules](#)
    - [Datatypes](#)
    - [Vocabulary](#)
    - [Diagrams](#)
    - [Error handling](#)
  - [Method Reference](#)
    - [apiref](#) Accessing API documentation
    - [apisrv](#) API server data
    - [courses](#) Information on courses
    - [geo](#) Geographical data
    - [oauth](#) OAuth Authorization
    - [tt](#) Accessing activity timetables
    - [users](#) Accessing user information
  - [Examples](#)
- [Sign up for an API key](#)

## *USOS API Reference*

**USOS API** is a simple and fat-free REST protocol, which allows developers to access academic database. In case of this installation, you will be accessing **The University of Warsaw's** data, but there are many other institutions which use USOS API.

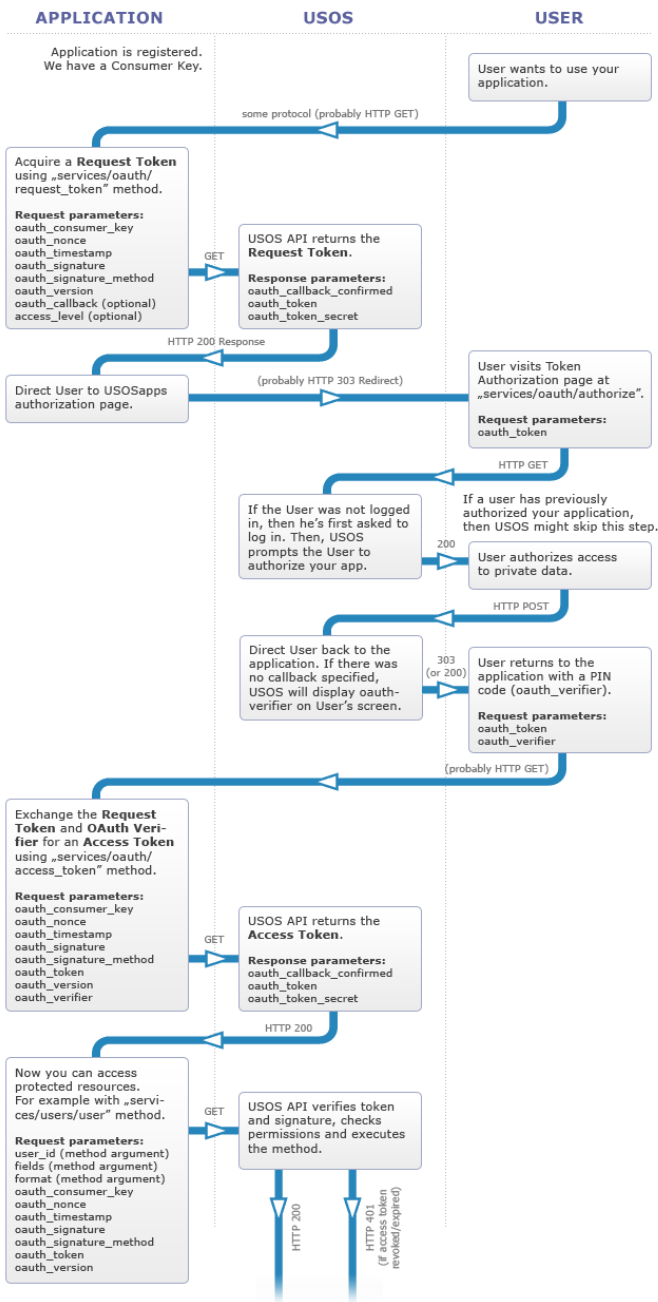
Basically, there are **three ways** to access USOS API:

- **Anonymously:** If you choose not to authenticate, you will be limited only to a subset (but still, fairly usable subset) of API methods. The main advantage of this solution is it's simplicity: just plain old simple HTTP requests.
- **With an API Key:** With one of those, you'll be able to **ask users to share their data** with you. Acquiring an API Key is easy, but you will also have to understand the Authorization stuff (we use a widely known OAuth standard, so there are many client libraries for you to find).
- **Administrative API Key:** This one is big. It might - for example - allow you to run any method as any user. You will need to contact us directly to get one, though!

USOS API Reference is a complete set of documentation needed in order to use the API.

- Read about our [Authorization model](#).
- Browse [available services](#) (we are still adding new ones!).
- Try some working [examples](#).

*Thanks for your interest!  
USOS API Team.*



# Do uwierzytelniania wykorzystuje się protokół **OpenAuth**

## Przykładowy scenariusz:

- Użytkownik korzysta z kalendarza w aplikacji Google
- Aplikacja ta chce ściągnąć z USOS dane o zajęciach studenta na uczelni
- Wywołuje metodę USOS-API
- Użytkownik zostaje przekierowany na stronę logowania centralnego systemu uwierzytelniania uczelni
- Loguje się
- Pojawia się strona z pytaniem o zgodę na dostęp do określonych danych
- Jeśli użytkownik wyrazi zgodę, to aplikacja dostaje dane z USOS
- Wyświetla je na kalendarzu Google



## DEVELOPER CENTER

USOS API is an experimental project which allows external developers to incorporate various USOS-related data into their applications.

USOS is a computer system that manages most study-related activities in over 30 higher education institutions in Poland.

[Read USOS API Introduction to get started!](#)

Sign up for an API key:

\*Application Name:

Application's Webpage:

Application name and webpage will be visible to the users. If your application doesn't have a webpage yet, leave this field blank.

\*Your email address:

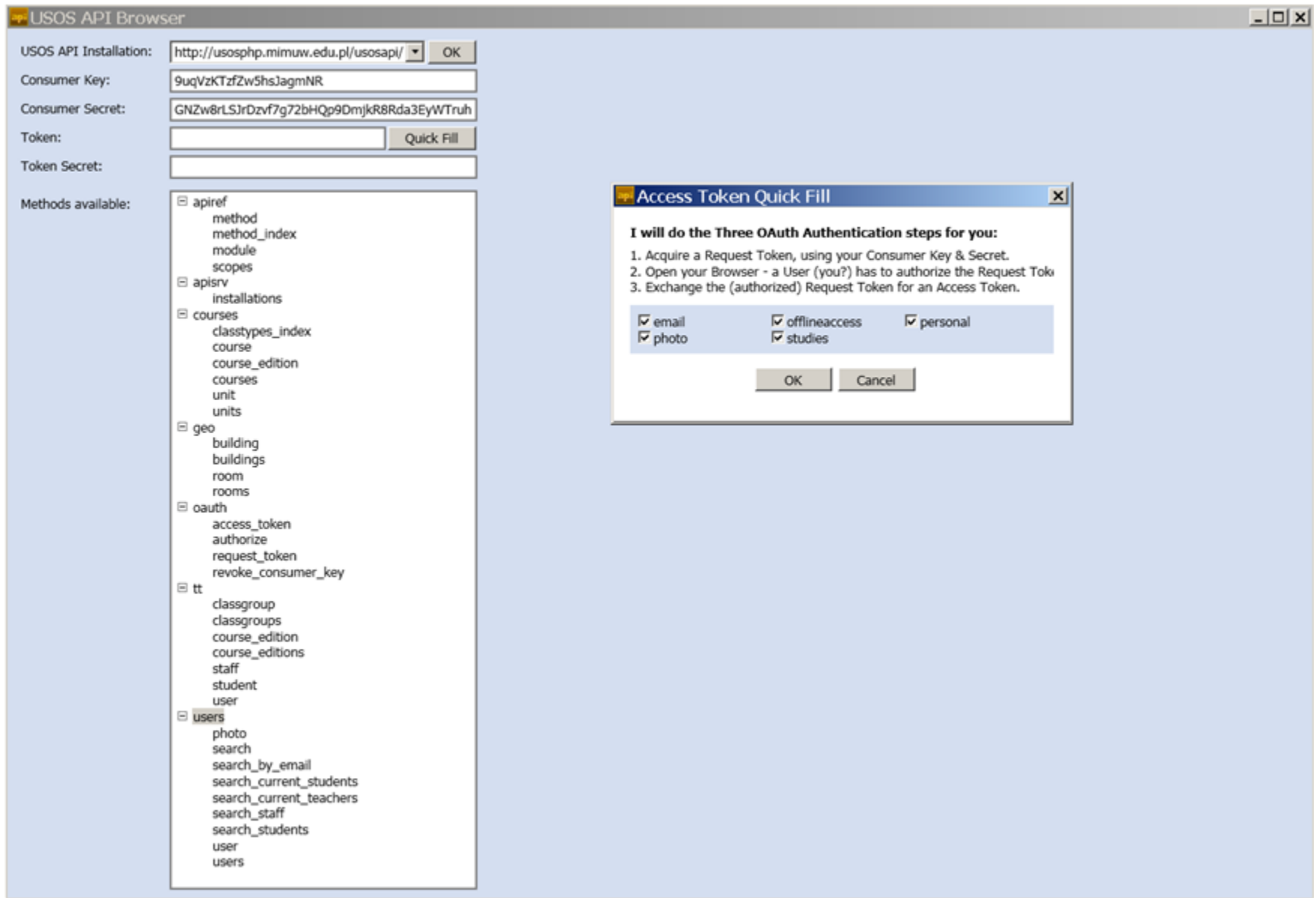
Email address will be hidden from the users. We won't send you activation emails nor any unwanted updates, but it still is just refer to a valid account at which we may contact you. In future USOS API revisions it might be used for some special actions (like, for example, Consumer Key secret-resetting process).

\* - required fields.

Submit

- Aplikacja nie posługująca się **żadnym kluczem** może sięgać jedynie do publicznie dostępnych danych (np. katalog przedmiotów)
- **Klucz administracyjny** pozwala na pełny dostęp do danych bez zgody użytkownika
- Aplikacja posługująca się **zwykłym kluczem** musi uzyskać zgodę użytkownika na dostęp do danych





API Browser to program napisany w WPF, który korzysta z USOS-API, a zarazem pozwala na testowanie jego metod

The image shows a desktop environment with two windows. On the left is the 'USOS API Browser' window, which has a configuration section for 'USOS API Installation' with fields for 'Consumer Key' (9uqVzKTzfZw5hsJagmNR), 'Consumer Secret' (GNZw8rLSJrDzvf7g72bHQp9DmjkR8Rda3EyWTruh), 'Token', and 'Token Secret'. Below this is a tree view of 'Methods available' including categories like 'apiref', 'apisrv', 'courses', 'geo', 'oauth', 'tt', and 'users'.

On the right is a Mozilla Firefox browser window titled 'Logowanie - Uniwersytet Warszawski'. The address bar shows 'https://logowanie.uw.edu.pl/cas/login?local'. The page features the University of Warsaw logo and the text 'Centralny Serwer Uwierzytelniania'. It contains a login form with fields for 'PESEL' and 'Hasło', a 'ZALOGUJ' button, and a checkbox for 'Ukryj mój PESEL'. The status bar at the bottom of the browser shows 'Zakończono' and search results for 'rygielski'.

Użytkownik zostaje przekierowany na stronę logowania centralnego systemu uwierzytelniania uczelni



**API-browser** asks your permission to access data on your USOS account.

Do you allow this application to perform the following actions?

**Access your basic information**

Such as your name, information whether you're a student and/or a staff member, public information available on your USOSweb profile.

**Send you emails**

Access your email address. This application may send messages directly to your email account.

**Access your data at any time**

This application can access your data even when you're logged off and not using the application (so called "offline access").

**Access your personal data**

Such as your PESEL number, date of birth, etc.

**Access your photo**

This application may download and use your USOS ID photo.

**Access general information on your studies**

Which programmes, courses and class groups you're attending (or attended).

Allow

Don't allow

Użytkownik po zalogowaniu jest proszony o wyrażenie zgody na dostęp do określonych danych

**USOS API Browser**

USOS API Installation:

Consumer Key:

Consumer Secret:

Token:

Token Secret:

Methods available:

- [-] apiref
  - method
  - method\_index
  - module
  - scopes
- [-] apisrv
  - installations
- [-] courses
  - classtypes\_index
  - course
  - course\_edition
  - courses
  - unit
  - units
- [-] geo
  - building
  - buildings
  - room
  - rooms
- [-] oauth
  - access\_token
  - authorize
  - request\_token
  - revoke\_consumer\_key
- [-] tt
  - classgroup
  - classgroups
  - course\_edition
  - course\_editions
  - staff**
  - student
  - user
- [-] users
  - photo
  - search
  - search\_by\_email
  - search\_current\_students
  - search\_current\_teachers
  - search\_staff
  - search\_students
  - user
  - users

**Get public staff member's activities.**

[view full description of this method](#)

*user\_id:*

*start:*

*days:*

*fields:*

*format:*

*callback:*

*as\_user\_id:*

Use SSL

Sign with Consumer Key (optional)

Sign with Token (optional)

Try to make it more human-readable

```
[
  {
    "start_time": "2011-03-10 12:15:00",
    "end_time": "2011-03-10 14:00:00",
    "name": {
      "en": "Distributed Systems - Master's seminar",
      "pl": "Systemy rozproszone - Seminarium magisterskie"
    }
  },
  {
    "start_time": "2011-03-08 14:15:00",
    "end_time": "2011-03-08 16:00:00",
    "name": {
      "en": "Advanced Topics in Operating Systems - Lecture",
      "pl": "Zaawansowane systemy operacyjne - Wykład"
    }
  },
  {
    "start_time": "2011-03-08 17:00:00",
    "end_time": "2011-03-08 18:30:00",
    "name": {
      "en": "Designing large databases - Lab",
      "pl": "Projekt dużej bazy danych - Laboratorium"
    }
  }
],
```

Gdy wyrazi zgodę, program może sięgać po dane w bazie USOS i dostarczać je aplikacji (tutaj dane te są po prostu wyświetlane w oknie API Browsera)

# Przykłady zastosowań USOS-API

- Integracja systemu uczelnianego z Live@Edu firmy Microsoft
- Wsparcie dla grup studenckich w serwisach społecznościowych
- Synchronizacja planu zajęć na uczelni z kalendarzem Google czy w komórce
- Aplikacja udostępniająca przetworzone wyniki ankiet oceniających zajęcia
- Dowolna inna aplikacja webowa czy desktopowa

- Wraz ze wzrostem poziomu informatyzacji uczelni rośnie potrzeba integracji systemów
- Istnieje wiele dobrych darmowych narzędzi do integracji
- Przyszłość należy do rozwiązań opartych na usługach sieciowych, szynach korporacyjnych, architekturach zorientowanych na usługi (SOA)
- Systemy uczelniane będą się integrować nie tylko ze sobą, lecz także ze światem zewnętrznym (w tym serwisami społecznościowymi)