

UNIWERSYTET KARDYNAŁA STEFANA WYSZYŃSKIEGO  
W WARSZAWIE

WYDZIAŁ MATEMATYCZNO-PRZYRODNICZY  
SZKOŁA NAUK ŚCISŁYCH

Marta Mech  
56862  
Nauki Ścisłe

## **MIGRACJA INTERFEJSU PHP DO KATALOGU GWIAZD**

Praca licencjacka wykonana  
pod kierunkiem naukowym  
mgr Katarzyny Małek

WARSZAWA 2008

Marta Mech

imię i nazwisko studenta/studentki

nr albumu:: 56862

Nauki Ścisłe

kierunek:

Dziekan Wydziału Matematyczno-  
Przyrodniczego. Szkoła Nauk Ścisłych  
UKSW

prof. dr hab. Tadeusz Skośkiewicz

### Oświadczenie

Świadomy(a) odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego z żadnej uczelni.

Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

(podpis studenta)

Marta Mech

imię i nazwisko studenta/studentki

nr albumu:: 56862

Nauki Ścisłe

kierunek:

Dziekan Wydziału Matematyczno-  
Przyrodniczego. Szkoła Nauk Ścisłych  
UKSW

prof. dr hab. Tadeusz Skośkiewicz

### Oświadczenie

Oświadczam, że niniejsza praca napisana przez Pana/Panią ..... nr albumu ..... została przygotowana pod moim kierunkiem i stwierdzam, że spełnia ona warunki do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

(podpis promotora)

# SPIS TREŚCI

<b>SPIS TREŚCI</b> .....	<b>3</b>
<b>CEL</b> .....	<b>4</b>
<b>ROZDZIAŁ 1 EKSPERYMENT „II OF THE SKY”</b> .....	<b>5</b>
<b>ROZDZIAŁ 2 ZASTOSOWANE TECHNOLOGIE</b> .....	<b>17</b>
2.1 PHP, JAKO SKRYPTOWY JĘZYK PROGRAMOWANIA .....	17
2.2 NIEZWYKŁE MOŻLIWOŚCI POSTGRESQL .....	18
2.2.1 <i>Historia PostgreSQL</i> .....	18
2.2.2 <i>Architektura PostgreSQL</i> .....	18
2.3 RELACYJNA BAZA DANYCH - DB2.....	19
2.3.1 <i>Enterprise Server Edition</i> .....	20
<b>ROZDZIAŁ 3 TESTOWY INTERFEJS DZIAŁAJĄCY Z POSTGRESQL I DB2</b> .....	<b>21</b>
<b>ROZDZIAŁ 4 MIGRACJA INTERFEJSU KATALOGU GWIAZD - DOSTOSOWANIE DO WSPÓŁPRACY Z BAZĄ IBM DB2</b> .....	<b>24</b>
<b>PODSUMOWANIE</b> .....	<b>30</b>
<b>PODZIĘKOWANIE</b> .....	<b>31</b>
<b>BIBLIOGRAFIA</b> .....	<b>32</b>

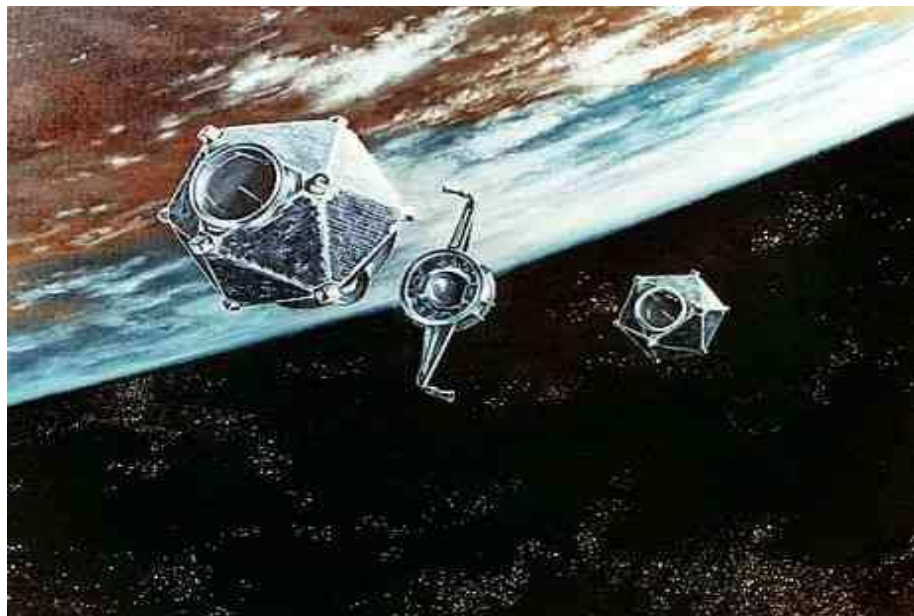
## **CEL**

Celem niniejszej pracy była modyfikacja interfejsu do przeglądania katalogu gwiazd w eksperymencie „ $\pi$  of the Sky”. Dotychczas interfejs ten działał wyłącznie z danymi przechowywanymi w bazie PostgreSQL. Ponieważ projekt „ $\pi$  of the Sky” zamierza w najbliższym czasie zamianę silnika PostgreSQL silnikiem DB2 konieczna była modyfikacja istniejącego już interfejsu w taki sposób, aby mógł on korzystać z obu baz w sposób automatyczny, bez ingerencji użytkownika. Dodatkowo istnienie dwóch różnych systemów baz danych powinno być niewidoczne dla użytkownika interfejsu.

## ROZDZIAŁ 1 Eksperyment „ $\pi$ of the sky”

Głównym celem projektu „ $\pi$  of the Sky” jest poszukiwanie i badanie poświat optycznych stowarzyszonych z błyskami gamma (GRB) oraz innych szybko zmiennych zjawisk astrofizycznych z rozdzielczością czasową 10 sekund. Projekt ten zrzesza naukowców z Centrum Fizyki Teoretycznej Polskiej Akademii Nauk (CFT PAN), Instytutu Problemów Jądrowych im. Andrzeja Sułtana, zespół pracowników i doktorantów z Instytutu Fizyki Doświadczalnej Wydziału Fizyki Uniwersytetu Warszawskiego oraz Politechniki Warszawskiej (Wydział Fizyki oraz Instytut Systemów Elektronicznych) oraz Centrum Badań Kosmicznych Polskiej Akademii Nauk. Projekt ten jest koordynowany przez dr hab. Lecha Mankiewicza z CFT PAN oraz dr hab. Grzegorza Wrochnę z IPJ. Projekt „ $\pi$  of the Sky” powstał z inspiracji dwóch znakomitych astrofizyków, nieżyjącego już prof. Bogdana Paczyńskiego z Princeton University, a także dr Grzegorza Pojmańskiego z OA UW.

Rozbłyski gamma po raz pierwszy zostały zaobserwowane pod koniec lat 60 XX wieku przez wojskowe satelity VELA, które zostały wystrzelone na orbitę okołoziemską, w celu kontroli zakazu prób jądrowych w kosmosie, czyli tzw. „nuklearnej ciszy”.



**Rysunek 1.** Satelity Vel – wizja artystyczna.

**Źródło:** <http://vesta.astro.amu.edu.pl/>

W wyniku tych obserwacji astronomowie zaczęli zadawać pytania, gdzie powstają rozbłyski gamma oraz jakie jest ich źródło. W celu odpowiedzi na pytanie, zbudowali wiele teorii, a każda była inna od poprzedniej. Rok 1991 był prawdziwym przełomem w zrozumieniu istoty błysków gamma, ponieważ wtedy prace rozpoczął instrument BATSE, umieszczony na pokładzie satelity CRGO (Compton Gamma Ray Observatory). Dzięki BATSE został zmierzony dokładnie kształt oraz czas trwania impulsów promieniowania. To dzięki jego początkowej analizie można rozróżnić rozbłyski krótkie, czyli trwające mniej niż 2 sekundy i długie - trwające dłużej. Przyczyna długich rozbłysków upatrywana jest w nagłych eksplozjach, które związane są z ostatnią fazą rozwoju gwiazdy, nazywaną hipernową, czyli bardzo masywną supernową. W wyniku rozbłysków emitowane jest promieniowanie gamma, które uważane jest za najsilniejsze źródło promieniowania we wszechświecie. W przypadku krótkich rozbłysków, przyczyną ich powstawania jest zlewanie się dwóch gwiazd w ciasnym układzie dwóch gwiazd, który może składać się z dwóch gwiazd neutronowych, albo gwiazdy neutronowej i czarnej dziury. Jednak najważniejszą informacją pochodzącą od BATSE było stworzenie mapy błysków, z których wynikało izotropowe rozłożenie błysków na niebie, brak koncentracji w naszej Galaktyce, a co za tym idzie przesłanka o pozagalaktycznym pochodzeniu tych błysków. Na długo przed opublikowaniem wyników pochodzących od BATSE Bogdan Paczyński jako jedyny głosił, że błyski  $\gamma$  występują poza naszą galaktyką [1].

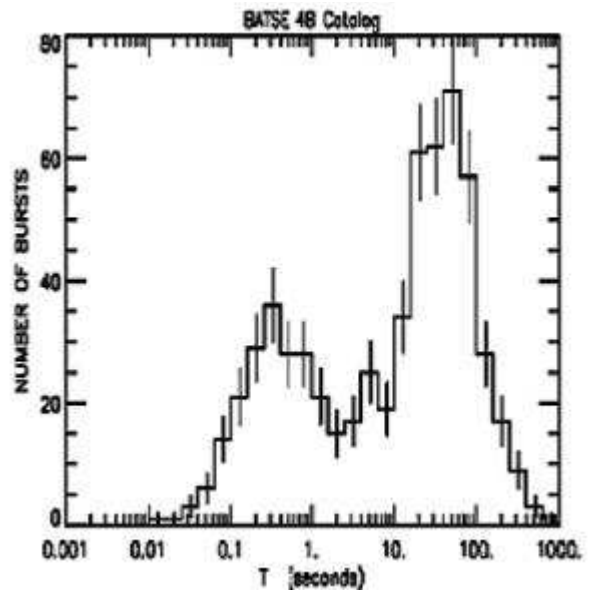
W 1996 roku nastąpił kolejny przełom dzięki wystrzeleniu satelity BeppoSAX. Ten włosko - holenderski satelita wyposażony był w detektor promieniowania gamma, kamerę rentgenowską szeroką (40°) i precyzyjną (rozdzielczość 3 minut kątowych). Dnia 28 lutego 1997 BeppoSAX zaobserwował błysk gamma w zakresie  $\gamma$  oraz X. W przeciągu kilku godzin obliczył kierunek błysku GRB970228<sup>1</sup>. W 12 godzin od obserwacji błysku informacja ta została przekazana do eksperymentów naziemnych, które nakierowały się na ten punkt na niebie, co w rezultacie pozwoliło na zaobserwowanie przez teleskopy naziemne poświaty radiowej i poświaty w zakresie widzialnym.

---

<sup>1</sup> GRB970228 – błysk gamma z roku 1997 28 lutego.

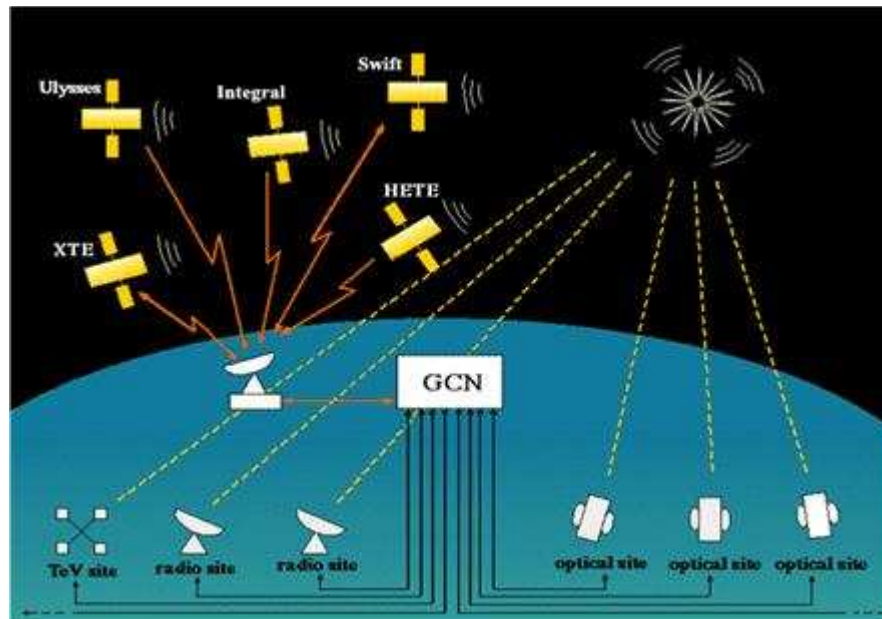


**Rysunek 2.** Satelita BeppoSAX.  
Źródło: <http://heasarc.gsfc.nasa.gov>



**Rysunek 3.** Rozkład długości trwania GRB.  
Źródło: <http://gammaray.msfc.nasa.gov/batse/>

W celu ułatwienia komunikacji między satelitami rejestrującymi błysk gamma, a instrumentami naziemnymi, powstała działająca do dzisiaj, sieć dystrybucji współrzędnych GRB (Grb Coordinate Network, GCN). Jej zadaniem jest informowanie o zaobserwowaniu błysku gamma wszystkich zainteresowanych. Alarmy te, inaczej zwane trygerami pochodzą z wielu satelitów (Rys 4). Najnowszy satelita do badania błysków gamma, to SWIFT wystrzelony w roku 2004 przez NASA. Składa się on z czterech podstawowych elementów: detektora promieniowania gamma, detektora promieniowania rentgenowskiego oraz teleskopu działającego w zakresie widzialnego, a także ultrafioletowego, dzięki czemu po raz pierwszy satelita może sam zmierzyć poświatę optyczną pozostawioną przez GRB. Samo działanie tego teleskopu polega na tym, że na podstawie danych z detektora gamma i promieniowania rentgenowskiego, potrafi on samodzielnie w przeciągu kilku sekund zlokalizować miejsce błysku. Następnie detektor optyczny jest nakierowywany na to położenie i w ciągu kilku – kilkudziesięciu sekund wykonywane są zdjęcia w zakresie widzialnym. Ograniczenie teleskopu spowodowane jest zbyt długim czasem nakierowywania powodującym czas martwy w obserwacjach. Mimo tego jest on najlepszym źródłem trygerów dla innych eksperymentów, a w szczególności instrumentów naziemnych.



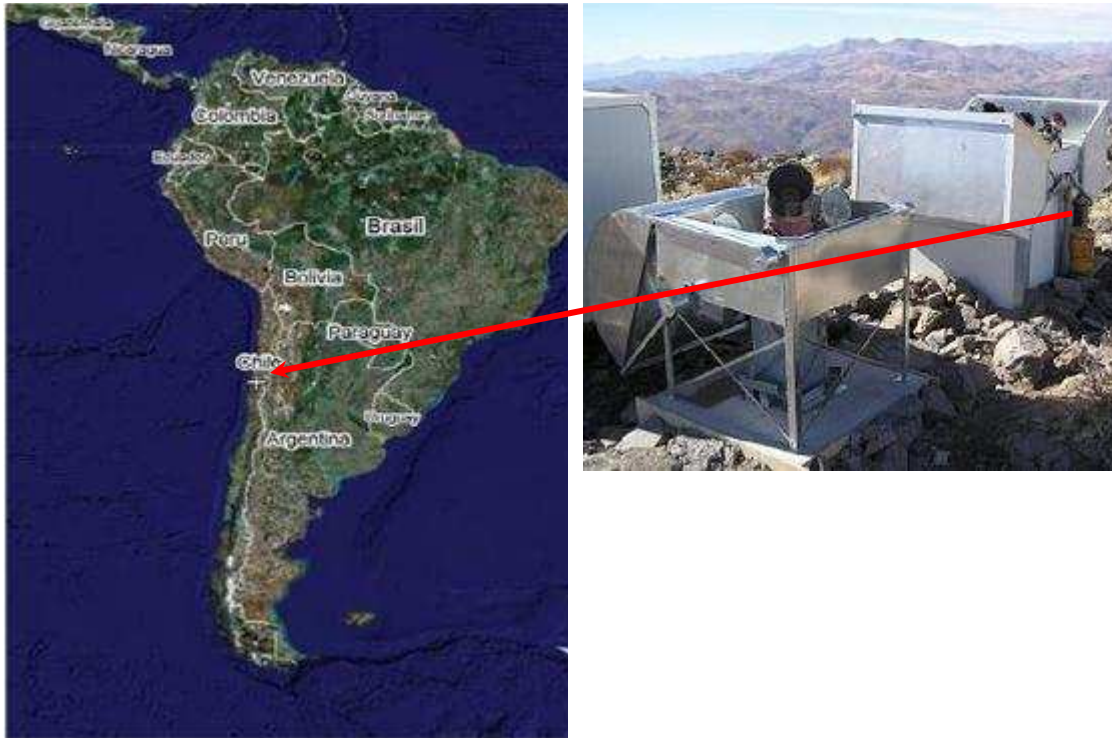
Rysunek 4. Struktura sieci GCN.

Źródło: [gcn.gsfc.nasa.gov/](http://gcn.gsfc.nasa.gov/)

Pomimo, że zjawisko to pojawia się na całej sferze niebieskiej, detekcja rozbłysku jest utrudniona. Utrudnienie to spowodowane jest przypadkowością występowania rozbłysku, a także czasem jego trwania wynoszącym od kilku ułamków sekund do nawet godziny, podczas którego wypromieniowana energia w momencie błysku równa jest w przybliżeniu energii wypromieniowanej przez Słońce przez okres kilku miliardów lat.

Naziemne obserwacje nie mogą być przeprowadzane w zakresie promieniowania gamma, z tego powodu, że zakres ten jest całkowicie tłumiony przez ziemską atmosferę. Na powierzchni Ziemi możliwe są obserwacje mniej energetycznych pasm takich jak radiowe, optyczne czy w zakresie TeV. Obserwacje te są bardzo istotne, ponieważ są uzupełnieniem zarejestrowanego obrazu w promieniowaniu gamma. Detekcja poświat dokonywana jest przy pomocy detektorów optycznych, które muszą być nakierowane na konkretny punkt na niebie, z tego względu, że mają wąskie pole widzenia. Współrzędne przekazywane są z satelity podczas generowania trygera. W momencie, gdy zostanie wykryty błysk gamma, rozpoczyna się cały proces podejmowania decyzji o wygenerowaniu trygera, nakierowaniu teleskopu i zarejestrowaniu obrazu. W rezultacie, od momentu pojawienia się błysku do momentu zarejestrowania go, może upłynąć nawet kilkadziesiąt minut. Lukę w obserwacjach mają zapełnić małe zautomatyzowane teleskopy, które posiadają szerokie pole widzenia i krótką ogniskową. Gdy otrzymują one tryger z sieci GCN [2], wtedy w krótkim czasie nakierowują się na dany punkt na niebie. Takim właśnie małym teleskopem – robotem jest polski detektor „ $\pi$  of the Sky”.





**Rysunek 5.** Aparatura projektu „ $\pi$  of the Sky” w Las Campanas w Chile.

**Źródło:** <http://grb.fuw.edu.pl/>.



**Rysunek 6.** Droga Mleczna zarejestrowana przez kamery „ $\pi$  of the Sky”.

Miejsce, w którym został umieszczony prototyp detektora zostało wybrane bardzo starannie. Czynniki, które zadecydowały, o umieszczeniu aparatury w Las Campanas Observatory w Chile, to przede wszystkim ukształtowanie terenu i czyste powietrze, dzięki czemu możliwa jest obserwacja praktycznie przez cały rok, czyli około 320 nocy w roku. Dodatkowym aspektem, decydującym o umieszczaniu aparatury w tym

miejscu, jest także bliska obecność innych polskich projektów ASAS (All Sky Automated Survey) oraz OGLE (The Optical Gravitational Lensing Experiment) [3].



**Rysunek 7.** Las Campanas w Chile.

**Źródło:** <http://upload.wikimedia.org/wikipedia/>

Przed umieszczeniem aparatury w Chile, przeprowadzane zostały wstępne testy techniczne, które odbyły się w Brwinowie pod Warszawą. Na początku badań korzystano z kamery o rozdzielczości 786 x 512 pikseli. Aparatura ta gromadziła dane przez okres 10 miesięcy, a po upływie tego okresu została zamieniona na kamerę o lepszej rozdzielczości 2000 x 2000 pikseli, która pracowała przez mniej więcej jeszcze jeden miesiąc. W tym czasie została zbudowana kamera CCD442A, z chipem 2032 x 2032 pikseli, każdy o wielkości 15 x 15  $\mu\text{m}^2$ . Sczytanie całego chipu zajmuje zaledwie 2 sekundy. Sygnał pochodzący z kamery jest wzmacniany i digitalizowany, czyli przetwarzany z analogowego na cyfrowy przez 16 bitowy ADC, a następnie zapisywany w pamięci. Wszelkie sygnały, które pochodzą z kamery są odbierane i kontrolowane przez komputer, poprzez łącze USB. Każda z kamer wyposażona jest w migawkę, specjalnie zaprojektowaną do długotrwałego działania, wytrzymałą co najmniej  $10^7$  bezawaryjnych otwarć [4]. Została ona zaprojektowana przez inżynierów z Uniwersytetu Warszawskiego. Wykorzystywane obiektywy mają ogniskową  $f= 85$  mm oraz aperturę  $d= f/1.42$ . Pole widzenia każdej kamery to  $20^\circ \times 20^\circ$ . Wykorzystywane obiektywy to Planer- T\* firmy Canon. Dwie kamery znajdujące się na jednym montażu pracują w koincydencji. Montaż może obracać się w kierunku zaobserwowanego zjawiska na niebie w czasie krótszym niż minuta



**Rysunek 8.** Aparatura używana w Brwinowie.

Źródło: <http://grb.fuw.edu.pl/>



**Rysunek 9.** Po lewej stronie znajduje się prototyp detektora umieszczony w otwieranej kopule, natomiast po prawej stacja kontrolna z komputerami.

Źródło: <http://grb.fuw.edu.pl/>

Po fazie testowej przeprowadzonej w Brwinowie, system został przygotowany do nowych warunków, w jakich miał pracować. Budowany był z myślą o niezawodności, czyli całkowicie zautomatyzowany, samodiagnostozujący problemy oraz wszelkie usterki. Z Brwinowa system został przewieziony do Las Campanas Observatory w Chile, gdzie w lipcu 2004 roku ruszyły badania.

Detektor obsługiwany jest poprzez zestaw programów i skryptów działających pod systemem operacyjnym Linux. Oprogramowanie to składa się z modułów, odpowiedzialnych za różne urządzenia, np. montaż, kamery, pozyskiwanie danych DAQ (Data Acquisition System). Moduły te mają ze sobą łączność dzięki systemowi CORBA (Common Object Request Broker Architecture), który zapewnia stałą komunikację pomiędzy obiektami pracującymi w zupełnie innych systemach. Algorytmy poszukujące błysków gamma, gwiazd nowych oraz innych szybkozmiennych obiektów, a także te służące do szybkiej fotometrii, zostały stworzone przez członków zespołu „ $\pi$  of the Sky” [5]. W przypadku precyzyjnej fotometrii<sup>2</sup> oraz astrometrii<sup>3</sup>, algorytmy zostały zapożyczone z projektu ASAS, kierowanego przez dr Pojmańskiego, jednakże sukcesywnie są one zastępowane przez nowe, dedykowane specjalnie dla tego projektu. Jak już zostało wspomniane system zbudowano tak by był niezawodny. Oba komputery obsługujące detektor podłączone są do sieci internetowej, co pozwala na stałą komunikację zarówno z komputerami jak i kamerami bezpośrednio z Warszawy. Oprócz tego komputery

<sup>2</sup> Fotometria- proces polegający na odnalezieniu na otrzymanych klatkach gwiazd oraz odczytaniu ich jasności a także pozycji (x, y) na chipie.

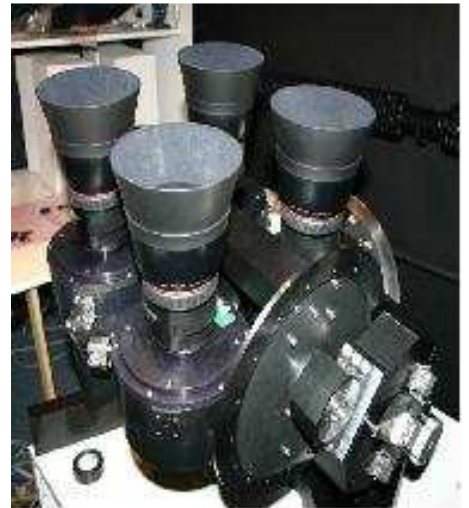
<sup>3</sup> Astrometria- transformacja współrzędnych instrumentalnych (x, y) na współrzędne astronomiczne (ra, dec).



znajdujące się w Las Campanas Observatory w Chile połączone są bezpośrednio do detektora „ $\pi$  of the Sky”, połączone są również między sobą, dzięki czemu możliwy jest szybki przepływ danych między nimi. Połączenie to służy również tzw. backup’owi, czyli tworzeniu kopii bezpieczeństwa, w momencie przerwania jednego z połączeń. Komputery posiadają również opcje „Wake on LAN” i „BOOT on LAN”, co umożliwia uruchomienie ich przez sieć w przypadku awarii plików systemowych. Obydwa PC mogą być resetowane, a także włączane i wyłączane przez przekaźniki zainstalowane na obydwu komputerach oraz w listwach zasilających. Komunikacja z systemem odbywa się za pośrednictwem modułu PiShell. Automatyczne kopiowanie informacji o systemie następuje, co 15 minut na serwer WWW w Warszawie. W przypadku wybranych skompresowanych zdjęć nieba, przesyłanie odbywa się, co 15 minut [2]. W momencie, gdy wystąpią jakiegokolwiek problemy w działaniu któregoś z modułów, wysyłana jest wiadomość SMS/e-mail, do osoby dyżurującej.



**Rysunek 10.** Wnętrze pomieszczenia kontrolnego w Las Campanas Observatory w Chile.  
**Źródło:** <http://grb.fuw.edu.pl/>



**Rysunek 11.** 4CCD na ruchomym montażu.  
**Źródło:** <http://grb.fuw.edu.pl/>

Podczas obserwacji, system działa automatycznie, według przygotowanego skryptu nocnego, utworzonego dla każdej nocy indywidualnie. Skrypt taki zawiera wszelkie informacje, które niezbędne są do przeprowadzenia obserwacji, takie jak np. ruch montażu, początek i koniec obserwacji. Jeżeli w momencie trwania obserwacji zostanie zarejestrowany błysk  $\gamma$ , montaż zmienia pozycje z zaplanowanej w skrypcie, na nową, której współrzędne zostały przekazane w wysłanej z sieci koordynującej trygerze. Montaż na zmianę pozycji potrzebuje kilku - kilkudziesięciu sekund, dzięki czemu możliwa jest obserwacja poświaty pozostawionej przez GRB. Obserwacja taka trwa około 30 minut, po czym montaż wraca do zaplanowanej pozycji.

Dwukrotnie podczas obserwacji wykonywany jest skan całego nieba (skan wieczorny i poranny). Podczas takiego skanu dostępne pola fotografowane są trzykrotnie z czasem ekspozycji 10 sekund.

Wszystkie zdjęcia pochodzące z obserwacji, poddawane są analizie w czasie rzeczywistym pod względem poszukiwania błysków o czasie narastania rzędu kilku sekund, co oznacza, że klatki wprowadzane są do algorytmu tuż po zakończeniu ich odczytu z kamer. Jednak obserwacje nie skupiają się wyłącznie na poszukiwaniu błysków gamma, ale także na poszukiwaniu gwałtownych pojaśnień, gwiazd zmiennych, a także innych zjawisk o krótkim czasie zmienności [1].

Działanie algorytmu poszukującego błysków gamma jest dosyć skomplikowane. W pierwszej kolejności, po przeniesieniu zdjęć z obydwu komputerów do pamięci RAM, podejmowana jest analiza online polegająca na wykryciu błysków, którą można podzielić na fazy. W pierwszej fazie obraz jest preparowany, odnajdowane są wszelkiego rodzaju tła, a także wykonywane jest progowanie, czyli odrzucenie pikseli poniżej pewnego ustalonego progu. W ten sposób następuje zmniejszenie przypadków, gdzie można by zaobserwować błysk. W kolejnej fazie następuje porównanie błysków z katalogiem istniejących gwiazd oraz torami lotów znanych satelitów, powoduje to kolejne cięcia i w ten sposób liczba potencjalnych kandydatów na błyski zostaje zawężona. Zdjęcia, na których algorytm do poszukiwania błysków znalazł interesujący przypadek, są wycinane w formie kwadratu o bokach 100x100 pikseli i środkiem w miejscu wystąpienia pojaśnienia. Oprócz tego zostają zrobione wycinki na klatkach zarówno sprzed, jak i po obserwacji błysku. Następnie wszystko zapisywane jest na dysku, a także wysłane na serwery komputerowe znajdujące się w Warszawie, w celu wizualnego sprawdzenia tych przypadków przez człowieka. Kiedy zapisane dane zostają skopiowane na drugi komputer sumowane są one po 20, w celu podwyższenia stosunku sygnału do szumu, jak również zwiększenia zasięgu widzenia gwiazd o 2 magnitudo. Ponownie wykonywana jest analiza w celu wyszukania błysków, jednak tym razem z czasem narastania rzędu minut. Po zakończeniu obserwacji online rozpoczyna się analiza offline na obu komputerach. Na pierwszym komputerze przeprowadzana jest redukcja, szybka fotometria, astrometria, a także katalogowanie<sup>4</sup>. Na drugim natomiast na zsumowanych klatkach wykonana jest redukcja<sup>5</sup>, zwykła fotometria, a później podobnie jak na pierwszym komputerze astrometria i katalogowanie.

Redukcja składa się z dwóch części:

- a) Odjęcie „ciemnej klatki” po angielsku „dark frame”,
- b) Podzielenie przez klatkę „flat” po angielsku „flat field”.

---

<sup>4</sup> Katalogowanie – normalizacja jasności aparaturowych wyliczonych w procesie fotometrii, a następnie przypisanie otrzymanych danych do poszczególnych pomiarów w plikach astrometrii.

<sup>5</sup> Redukcja – odjęcie od klatki tła oraz jej optymalizacja, w ten sposób klatka przygotowana jest do dalszych procesów obróbki.

Ciemna klatka powstaje w wyniku ekspozycji tej samej długości i temperaturze, ale przy zamkniętej migawce. Odjęcie tej klatki pozwala na redukcję efektu tzw. prądu ciemnego, czyli ładunku, który gromadzi się w czasie ekspozycji na poszczególnych pikselach, który może powstać w wyniku np. emisji termicznej.

Niejednorodna optyka obiektywu przyczynia się do konieczności podzielenie obrazu przez „flat field”. Wykorzystywane obiektywy są prawie dwukrotnie jaśniejsze na środku aniżeli na rogach. W idealnej sytuacji „flat field” powinien być obrazem jednorodnie oświetlonej powierzchni, bądź fragmentu nieba [2].

Po przeprowadzonej redukcji następuje fotometria, która polega na utworzeniu listy gwiazd, pojawiających się na klatce, także ich współrzędnych na chipie. Obliczana jest również jasność gwiazd, która stanowi sumę wartości pikseli w pewnym zdefiniowanym otoczeniu gwiazdy tzw. aparaturze, od której odejmuje się wartość poziomu tła. Jak zostało już zasygnalizowane wcześniej istnieją dwa rodzaje fotometrii: szybka oraz precyzyjna. Pierwszy rodzaj fotometrii przeprowadzany jest na nieposumowanych klatkach, na komputerze, który odpowiedzialny jest za kontrolę kamer. Drugi rodzaj fotometrii wykonywany jest na klatkach posumowanych po 20, co zwiększa dokładność pomiaru poprzez zmniejszenie szumu.

Kolejnym etapem jest astrometria, która porównuje otrzymaną listę gwiazd otrzymanych z fotometrii z katalogiem gwiazd stałych. W tym etapie przeprowadzana jest transformacja ze współrzędnych instrumentalnych na współrzędne astronomiczne (rektascensję i deklinację).

Ostatnim etapem jest proces katalogowania podczas, którego następuje normalizacja jasności aparaturowych otrzymanych w procesie fotometrii, a następnie otrzymane dane przypisywane są do poszczególnych pomiarów w plikach astrometrii. Ostatnią fazą tego procesu jest umieszczenie wszystkich danych, które dotyczą pomiaru w bazie danych.

Algorytm poszukiwania gwałtownych zmian pojaśnień, działa już po zakończeniu obserwacji i skatalogowaniu wszystkich danych. Głównym jego założeniem jest analiza różnic jasności w funkcji czasu, a następnie, poprzez odpowiednie procedury, szukane są fluktuacje krzywych blasku<sup>6</sup> [1].

Podczas nocy zbieranych jest około 30GB danych. Na stałe po przeprowadzeniu dziennych analiz pozostaje około 10% z nich. Gdy dysk zostanie zapełniony, czyli zostanie ubierane 200GB danych, zostaje on zastąpiony nowym, a zapełniony dysk zostaje przewieziony do Warszawy. Wymiana taka następuje raz na 2-3 miesiące.

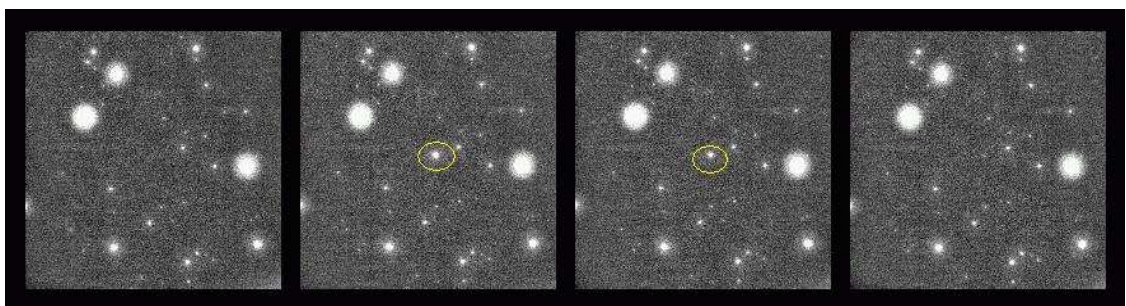
Naukowcy z projektu „ $\pi$  of the Sky” mogą pochwalić się licznymi odkryciami, w tym na największą uwagę zasługują dwa najnowsze odkrycia: Nową VSX J111217.4-

---

<sup>6</sup>Krzywa blasku – zależność między obserwowaną jasnością, a czasem.

353828 typu WZ Sagittae (XII 2007) oraz GRB 080319B (19 III 2008). Rozbłysk obiektu VSX J111217.4-353828 został zauważony przez algorytm do identyfikacji pojaśnienia na niebie i uzyskał potwierdzenie dzięki niezależnym obserwacjom. Po wykonanej analizie został określony jako nowa karłowata typu WZ Sagittae. Nowe typu WZ Sagittae to układy podwójne, w których skład wchodzi biały karzeł i późna gwiazda ciągu głównego (chłodna i mało masywna). Tuż poniżej temperatury jonizacji znajduje się gaz odbierany gwiazdzie towarzyszącej przez białego karła i opadający na jego powierzchnię. W wyniku zagęszczenia dochodzi do wzrostu temperatury, jonizacji gazu i istotne zmiany warunków przepływu gazów przez dysk akrecyjny. Do rozbłysków nowej (wybuchów) dochodzi, gdy gaz uderza w powierzchnie białego karła. Zjawisko powtarza się cyklicznie, jednak z różną częstotliwością.

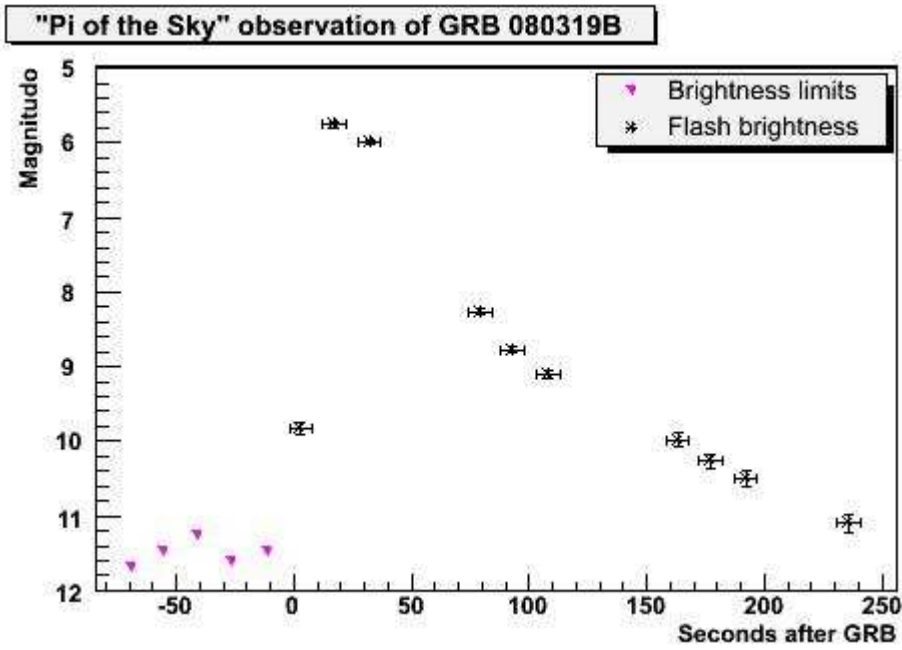
Detekcja poświaty optycznej błysku GRB080319B to największe z odkryć eksperymentu „ $\pi$  of the Sky”. Był to najjaśniejszy błysk optyczny stowarzyszony z błyskiem gamma widziany przez człowieka. Do eksplozji doszło 7,5 miliardów lat świetlnych od Ziemi, w połowie odległości do krańców widzialnego Wszechświata. Rozbłysk obserwowany przez " $\pi$  of the Sky" został zaobserwowany jednocześnie w widmie gamma przez satelitę "Swift", dzięki czemu został zidentyfikowany jako tzw. rozbłysk gamma (ang. Gamma Ray Burst - GRB) i oznaczony jako GRB 080319B<sup>7</sup> (drugi spośród zarejestrowanych tej nocy). Najprawdopodobniej źródłem błysku była śmierć masywnej gwiazdy, która dała początek czarnej dziurze. Obserwacje optyczne prowadzone w czasie pierwszej minuty wybuchu są kluczowe dla zrozumienia mechanizmu zachodzącego zjawiska. Obserwacje optyczne "Pi of the Sky" w połączeniu z danymi w widmie gamma otrzymanymi przez satelitę "Swift" po raz pierwszy potwierdzają z kilkusekundową precyzją, że w czasie wybuchu takiego jak obserwowany, emisja optyczna zachodzi równocześnie z emisją promieniowania gamma” [3].



**Rysunek 12.** GRB 080319B –kolorem żółtym zaznaczono Pojawie się optycznego odpowiednika błysku gamma. Zdjęcia pochodzą z archiwum eksperymentu „ $\pi$  of the Sky”.

Źródło: <http://grb.fuw.edu.pl/pi>

<sup>7</sup> GRB 080319B – błysk gamma z 19 marca 2008 roku.



**Rysunek 13.** Obserwowany przebieg zmian jasności błysku GRB030819B w jednostkach jasności gwiazdowej (magnitudo). Magnitudo jest skalą odwrotną gdzie "zero" odpowiada najjaśniejszym widocznym gwiazdom, a 6 jest granicą widzialności gołym okiem. Zaobserwowany przez "Pi of the Sky" rozbłysk GRB 080319B był w chwili kulminacji wystarczająco jasny, by dostrzec go bez pomocy przyrządów optycznych.

**Źródło:** <http://grb.fuw.edu.pl/>



## **ROZDZIAŁ 2 Zastosowane technologie**

### **2.1 PHP, jako skryptowy język programowania**

PHP jest to obiektowy, skryptowy język programowania, dzięki któremu można tworzyć dynamiczne strony WWW. Pierwsza wersja pojawiła się pod nazwą PHP/FI (Personal Home Page /Forms Interpreter). Została ona stworzona przez duńskiego programistę Rasmusa Lerdorfa. Lerdorf napisał zestaw skryptów Perla, które służyły do monitorowania internautów odwiedzających jego witrynę. W momencie, gdy ruch się wzmógł przepisał je w języku C, przy czym aplikacja została rozszerzona. Po pewnym czasie użytkownicy zaczęli prosić go o udostępnienie tych aplikacji, by również mogli korzystać z tych narzędzi na swoich stronach. 8 czerwca 1995 roku autor udostępnił ich kod źródłowy pod nazwą PHP Tools 1.0, natomiast w 1997 roku pojawiło się PHP/FI 2.0, jednak wersja ta przez większość czasu była testowana. Oficjalne wydanie miało miejsce w listopadzie 1997 roku. W 2002 roku Zeev Suraski oraz Andi Gutmans zaczęli ponowną modernizację silnika PHP w celu dodania do tego języka modelu obiektowego na poziomie dorównującym innym językom obiektowym. W lipcu 2004 roku pojawiła się stabilna wersja oznaczona numerem 5.0.0. Wersja ta charakteryzowała się całkowicie nowym modelem programowania obiektowego, w wyniku, czego została utracona pełna kompatybilność z poprzednimi wersjami PHP. Znaczącą zmianą wprowadzona w tym modelu jest sposób reprezentacji obiektów. We wcześniejszych wersjach obiekt był jednocześnie zmienną, co przysparzało sporo trudności, w tej wersji, zmienna obiektowa stała się wyłącznie referencją do właściwego obiektu, podobnie jak w Javie. Operacja przypisania powoduje powstanie drugiej referencji, która wskazuje na ten sam obiekt. Do tego jeszcze udostępnia zbiór interfejsów, które w znaczący sposób rozszerzają możliwości klas użytkownika. Oprócz tego zauważalne i znaczne zmiany oraz nowości nastąpiły w systemie modułów PHP. Wszelkie funkcje obsługi języka XML zostały przypisane i wbudowane bezpośrednio w jądro aplikacji. Wersja 5.0 wprowadziła mechanizm wyjątków. Tylko część modułów powiadamia jednak użytkownika o wystąpieniu błędów za ich pośrednictwem, większość umieszcza je na wyjściu programu w miejscu ich wystąpienia [6].

## 2.2 Niezwykłe możliwości PostgreSQL

PostgreSQL jest to system zarządzania bazą danych, w którym wykorzystywany jest język zapytań SQL, jak również wdrażany relacyjny model baz danych. Baza danych PostgreSQL posiada niezwykle możliwości, uważana jest za niemalże niezawodną, jak również charakteryzuje się bardzo dobrymi wskaźnikami wydajności. System ten działa na platformach UNIX oraz w systemach, które oparte są na platformach UNIX np. LINUX. PSQL jest dostępny na licencji BSD, czyli zgodna z zasadami Wolnego Oprogramowania<sup>8</sup>.

### 2.2.1 Historia PostgreSQL

Za początek relacyjnej bazy danych przyjmuje się lata między 1977, a 1985, kiedy to na Uniwersytecie Kalifornijskim w Berkeley opracowano relacyjną bazę danych Ingres. Zaistniała ona w wielu uniksowych komputerach zarówno społeczności akademickiej jak i badawczej. W 1984 – 1994 na Uniwersytecie w Berkeley kontynuowano pracę nad serwerem relacyjnych baz danych PostgreSQL. Zainteresowanie systemem PostgreSQL ciągle rosło, dało to początek otwarciu prac nad systemem w listach dyskusyjnych. Przy rozbudowywaniu i ulepszaniu tego systemu pracowało wielu ochotników. Nazwa również się zmieniła z Postgres 95 na PostgreSQL, która sugeruje język zapytań, wykorzystywany w tej bazie.

Obecnie programiści w sieci Internet pracują nad systemem PostgreSQL w podobny sposób jak ma to miejsce w przypadku innych systemów Open Source<sup>9</sup> takich jak: Perl, Apache oraz PHP. System ten charakteryzuje się tym, że użytkownik ma wgląd do kodu źródłowego i dzięki temu łatwiej mu wprowadzać wszelkie ulepszenia i poprawki istniejących już funkcji jak również tworzyć nowe [6].

### 2.2.2 Architektura PostgreSQL

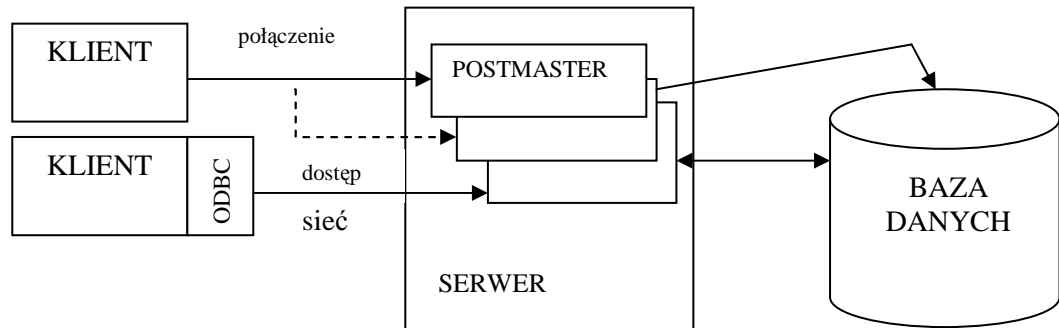
Architektura PostgreSQL opiera się na modelu klient – serwer. Kołem napędowym instalacji PostgreSQL jest proces serwera bazy danych, który uruchamiany jest na jednym serwerze. Baza ta na razie nie obsługuje systemów klastrowanych, czyli funkcji które pozwoliłyby obniżyć obciążenie, poprzez rozłożenie go na kilka serwerów. Taka

---

<sup>8</sup>Wolne Oprogramowanie - ruch programistów i użytkowników komputerów zaangażowanych w działania na rzecz wolnego dostępu do oprogramowania przez ogół użytkowników.

<sup>9</sup>Open Source – informuje, że oprogramowanie dostarczane jest wraz z kodem źródłowym.

funkcjonalność przeniosłaby się na dodatkową skalowalność, a także poprawiła niezawodność systemu. Klient i serwer są rozdzieleni, za sprawą sieci, a także opracowanych aplikacji klientów w środowisku wskazanym przez użytkownika



**Rysunek 14:** Rozproszona aplikacja PostgreSQL

**Źródło:** R. Stones, N. Matthew, "Od podstaw bazy danych i PostgreSQL", Helion, str. 35.

Na rysunku 14 widać kilka programów klienckich, może to być np.: Linux, UNIX, Windows, które łączą się z serwerem lokalnie bądź przez sieć. W momencie gdy klient łączy się z głównym procesem serwera bazy danych POSTMASTER, który tworzy nowy proces serwera specjalnie w celu obsługi żądania dostępu dla tego klienta. PostgreSQL może wydajnie utrzymywać integralność danych dla wielu użytkowników, dzięki koncentracji przetwarzania danych na serwerze, w zamian podjęcia próby kontroli tych klientów korzystających z tych samych baz danych, które przechowywane są w jednym katalogu [6].

## 2.3 Relacyjna baza danych - DB2

Powstanie DB2 upatruje się w latach 70, w momencie, gdy IBM stworzyło relacyjny model baz danych, a także język SQL. Baza ta uważana jest za pierwszą komercyjną implementację relacyjnej bazy danych, która była rozwijana w IBM w ramach projektów z rodziny System R<sup>10</sup>. Rok 1982 dał rozwiązanie o nazwie SQL/DSTM, zostało zintegrowane w ramach systemu serwerowego SYSTEM/38. Następnie w roku 1983 wyodrębnił się formalnie projekt ochrzczonej nazwą DB2 na systemie MVSTM - pierwotnie określany mianem DATABASE 2 [9]. Za początek relacyjnych baz danych przyjmuje się rok 1996, kiedy IBM udostępnił DB2 UDB (wersja) na platformy systemów otwartych. Wprowadzenie tej wersji dało możliwość przechowywania danych nie tylko relacyjnych, ale także audio, video, jak również dokumenty tekstowe. Bez wątpienia była to

<sup>10</sup> System R - przełomowy system bazodanowy zbudowany w latach 70. jako projekt badawczy IBM w San Jose Research Center (obecnie IBM Almaden Research Center). W systemie tym wprowadzono język zapytań SQL i wykazano wydajność przetwarzania transakcyjnego w relacyjnym modelu baz danych.

pierwsza wersja przystosowana do wykorzystywania w sieci WWW. Optymalizator zapytań DB2 - LEO (ang. Learning Optimizer) stanowi najwydatniejszą strategię realizacji zapytań w oparciu o doświadczenia z przeszłości. Optymalizator zapytań DB2 wybiega znacznie poza możliwość konkurencyjnych baz. Pozwala nie tylko optymalizować zapytania skierowane do bazy DB2, ale także rozproszone zapytania z wykorzystaniem innych systemów baz danych (np. Oracle, MsSQL, itd.). Tabele innych systemów baz danych mogą być obsługiwane jak lokalne tabele DB2.

Baza danych DB2 kumuluje w tzw. globalnym katalogu dane o zdalnych tabelach, brane są również pod uwagę aspekty zdalnej bazy (np. indeksy założone na tabeli, wielkość tabel, wydajność zdalnej bazy, buforowanie zdalnych obiektów).

### **2.3.1 Enterprise Server Edition**

DB2 Enterprise Server Edition (ESE) stanowi funkcjonalny serwer bazy danych. Wersja ta charakteryzuje się dobrą skalowością, dostępnością i opcją rozszerzeń. Jest ona znana i używana przez wiele firm. Dzięki DB2 UDB WSE oraz DB2 Connect możliwe jest łączenie się z tzw. systemami hostowymi. Daje również możliwość partycjonowania bazy danych, czyli podział danych w ramach jednego bądź wielu serwerów, z tym samym systemem operacyjnym, w rezultacie, czego rozmiar danych zależy wyłącznie od liczby dostępnych serwerów [7].

## ROZDZIAŁ 3 Testowy interfejs działający z PostgreSQL i DB2

W celu zapoznania się z językiem PHP, stworzyłam stronę pozwalającą monitorować temperaturę chipu w chwili wykonania zdjęcia nieba za pomocą detektora „ $\pi$  of the Sky”. W pierwszej fazie tworzenia opierała się ona wyłącznie na bazie PostgreSQL. Przygotowanie takich skryptów dało możliwość poznania samej struktury budowania stron internetowych oraz zapytań bazowych języka PHP. W momencie, gdy skrypty działały poprawnie, zostały one zmienione, w taki sposób, aby działały zarówno z bazą DB2 jak i PostgreSQL. Oczywiście wprowadzane zmiany miały być czytelne i zrozumiałe, jak również niewidoczne dla użytkownika. Wybór bazy, do której wysyłane jest zapytanie - odbywa się poprzez zmienną \$base, do której przypisywana jest liczba 1, odpowiadająca za bazę PostgreSQL bądź 2, która odpowiada za DB2. Najistotniejszym elementem tej zmiany to zapytanie do bazy. Ogólna zasada tworzenia zapytania jest identyczna jak w PostgreSQL, jednak nie należy zapomnieć o drobnej różnicy, każda nazwa tabeli w DB2 użyta w zapytaniu musi zostać poprzedzona słowem ‘main.’, w tym celu została stworzona dedykowana zmienna do której zostało przypisane słowo ‘main.’. W momencie, gdy strona pracuje z bazą PostgreSQL zmienna ta jest pusta, natomiast w przypadku pracy z DB2 nazwa tabeli zostaje poprzedzona „main.”. Główne zapytanie kierowane do bazy, pozwala na wyszukanie wyników, które nas interesują. Zbudowane przeze mnie zapytanie (tzw. SELECT) wraz z wysłaniem go do bazy ma następującą postać:

```
$sql_main= "SELECT    spathtofile,    ".$main."frame.id_frm,
fchiptset,    fchiptemp, ttime_ut, astrook FROM    ".$main."frame,
".$main."frame_det    WHERE    ".$main."frame.id_frm=
".$main."frame_det.id_frm    AND    idaynight    =$night    AND
abs(fchiptemp - fchiptset)=$delta";
if($base==1)
{
    $query_scan=pg_Exec($conn, $sql_main);
    if($query_scan){}
    else {echo "error1".pg_last_error();}
}
else if($base==2)
{
```

```

$query_scan=db2_prepare($conn,$sql_main);
db2_execute($query_scan);
if($query_scan){}
else {echo "Error:".db2_stmt_errormsg();}
}

```

Zmienna \$conn jest zbudowanym przeze mnie wcześniej połączeniem z bazą danych (w zależności od parametru \$base połączenie nawiązywane jest z bazą PostgreSQL lub DB2). pg\_Exec [8] odpowiada za wysłanie zapytania do bazy PostgreSQL natomiast db2\_prepare i db2\_execute to polecenia umożliwiające połączenie z bazą DB2 poprzez język skryptowy PHP.

Zapytanie to wyszukuje w bazie kolejno: spathtofile – nazwę klatki, id\_frm – jej numer identyfikacyjny, fchiptset – temperaturę jaka została ustawiona podczas wykonywania ekspozycji, fchiptemp – temperaturę rzeczywistą, odczytaną ze zdjęcia, a także ttime - czas początkowy ekspozycji w formacie timestamp. W przypadku gdy różnica pomiędzy temperaturą ustawioną przez detektor, a temperaturą rzeczywistą różni się o zadane \$delta stopni wynik zapytania zostaje wyświetlony na ekran w postaci tabeli. Wyniki te zostają pobrane z dwóch tabel: frame oraz frame\_det. Rezultatem wykonania skryptu jest następująca strona:

## NIGHT BASE

20061126

delta:

date:

[previous night](#)   [next night](#)

scan0708

FRAME	TIME_UT	ASTRO	FCHIPTSET	FCHIPTEMP	FCHIPTSET-FCHIPTEMP
<a href="#">k2a_061126_052.ast</a>	08:16:45	OK	-10	-11	1
<a href="#">k2a_061126_043.ast</a>	08:01:24	OK	-10	-11	1
<a href="#">k2a_061126_042.ast</a>	07:59:47	OK	-10	-11	1
<a href="#">k2a_061126_026.ast</a>	01:29:15	OK	-10	-11	1
<a href="#">k2a_061126_024.ast</a>	01:25:45	OK	-10	-11	1
<a href="#">k2a_061126_020.ast</a>	01:19:00	OK	-10	-11	1
<a href="#">k2a_061126_009.ast</a>	00:58:57	OK	-10	-11	1
<a href="#">k2a_061126_004.ast</a>	00:49:29	OK	-10	-11	1

Interfejs pokazuje wszystkie zdjęcia wykonane danej nocy, wraz z godziną, a także temperatury odpowiadające danej klatce, również zostaje wyliczana ich różnica. Poprzez

pole formularza możliwe jest decydowanie o granicznej różnicy temperatur dla jakiej zostaną wyświetlone wyniki. Lista rozwijana daje możliwość również wyboru daty pomiarów. Linki `previous night` i `next night`, jak sama nazwa wskazuje, pozwalają bezpośrednio wyświetlić dane z poprzedniego jak i następnego dnia. Związku z tym, że główna klatka w większości przypadków składa się z kilku klatek zsumowanych, został stworzony link pozwalający na sprawdzenie tych samych parametrów już dla pojedynczych klatek. Zapytanie dotyczące klatek pojedynczych znajduje się w oddzielnym skrypcie. Kliknięcie przyciskiem myszy na nazwę klatki otwiera okno, które wygląda następująco:

## PIDB

db: `scan0708` id\_frm: `11895` frame: `k2a_061126_043.ast`

frame	time_ut	fchiptset	fchiptemp	fchiptset-fchiptemp
<code>k2a_061126_01880.fitc</code>	<code>08:01:53</code>	<code>-10</code>	<code>-10</code>	<code>0</code>
<code>k2a_061126_01879.fitc</code>	<code>08:01:39</code>	<code>-10</code>	<code>-11</code>	<code>1</code>
<code>k2a_061126_01878.fitc</code>	<code>08:01:24</code>	<code>-10</code>	<code>-10</code>	<code>0</code>

Dane do tej tabeli pobierane są również z baz PostgreSQL i DB2. W momencie, gdy nie ma wyników pojawia się komunikat „no frame in pidb”.

Po zamieszczonych zrzutach ekranu widać, że użytkownik, który ogląda stronę, nie potrafi określić, na jakiej bazie działa. Jediną możliwością sprawdzenia, z jaką bazą łączy się skrypt PHP jest bezpośrednia weryfikacja tego w źródle strony. Dzięki temu został osiągnięty jeden z głównych celów towarzyszący tworzeniu tego interfejsu. Strona, którą wykonałam jest używana przez członków eksperymentu „ $\pi$  of the Sky” i została włączona do systemu monitoringu detektora.

## **ROZDZIAŁ 4 Migracja interfejsu katalogu gwiazd - dostosowanie do współpracy z bazą IBM DB2**

Interfejs WWW służący do wyszukiwania gwiazd został stworzony przez Marka Biskupa. Został on napisany z myślą współpracy jedynie z bazą PostgreSQL, a także w celu wyszukiwania gwiazd o zadanych parametrach takich jak: współrzędne niebieskie (rektascensja i deklinacja), jasność, liczba pomiarów, rodzaj zmienności itp. W pierwszej kolejności użytkownik wprowadza zakres parametrów, a następnie ma możliwość wyboru formy prezentowanych wyników. Rezultat może być wyświetlony w trojaki sposób, jako lista gwiazd, wykres a także jako liczba wszystkich gwiazd spełniające zadane kryteria.

Wybierając listę gwiazd, osoba korzystająca z interfejsu, ma możliwość posortowania wyników względem np. jasności, nazwy. Liczba wyświetlanych wyników jest ograniczona, ponieważ w momencie wyświetlenia kilkunastu tysięcy gwiazd o zadanych parametrach na raz, mogłoby dojść do obciążenia serwera. W celu uniknięcia tego przeciążenia wyniki wyświetlane są na kilku stronach.

Wyniki mogą być również prezentowane w formie graficznej, jako mapa nieba, bądź diagram zmienności. W przypadku mapy nieba na wykresie osie współrzędnych to Ra (rektascensja) i Dec (deklinacja), czyli współrzędne niebieskie. Każda z gwiazd na takim wykresie zaznaczona jest pojedynczym punktem. Oprócz mapy nieba wyświetlany jest diagram zmienności, który jest użyteczny w momencie wyszukiwania gwiazd zmiennych. Niestety ze względu na wydajność, na wykresach pojawia się wyłącznie 1000 wyników, natomiast reszta jest pomijana. Zarówno jeden jak i drugi wykres jest interaktywny, co oznacza, że jeśli zaznaczymy myszą interesujący nas obszar wykresu, to zostanie on powiększony. Powiększenie generuje zapytanie do bazy z nowymi parametrami, dzięki czemu zostaje zawężony obszar nieba, a prezentowane wyniki będą pochodziły tylko i wyłącznie z tego obszaru.

W momencie kliknięcia na nazwę gwiazdy bądź na pojedynczą gwiazdę na wykresie, przechodzi się na kolejną stronę WWW, dedykowaną temu obiektowi. Prezentowana jest tam krzywa blasku, która przedstawia zmiany jasności gwiazdy w funkcji czasu. W przypadku gwiazdy zmiennej na tej samej stronie pojawia się dodatkowy wykres tzw. fazowana krzywa blasku. Oprócz wykresów w tabeli zawarte są dodatkowe informacje o gwiazdzie, między innymi jej nazwa, identyfikator, średnia jasność, identyfikator w innych katalogach Tycho, GCVS, ASAS. W momencie, gdy klikniemy na identyfikator gwiazdy w innym katalogu, użytkownik zostaje przekierowany na stronę,



która jest poświęcona tej gwiazdzie z informacjami z bazy, które odpowiadają wybranemu katalogowi [2].

Interfejs do przeglądania gwiazd w bazie „ $\pi$  of the Sky ” składa się z kilkunastu skryptów, a każdy z nich ma przypisane zadanie do spełnienia. Interfejs tworzą następujące skrypty: `cssToolTip.css`, `dataform.php`, `functions.php`, `lightCurve.php`, `blank.html`, `curveOrSky.php`, `dbConfig.php`, `pointOrCurve.php`, `style.css`, `variability.php`, `customConfig.php`, `dbconnec.php`, `getFitsPart.php`, `superStarView.php`, `zoomCurve.php`, `config.php`, `scripts.js`, `zoom.js`, `customDbConfig.php`, `skyMap.php`, `index.php`, `starsAround.php`, `csshorizontalmenu.css`, `customPathConfig.php`, `deepPurple.php`, `templates_c`, `csshorizontalmenu.js`, `dhtmlHistory.js`, `lightCurveData.php` i `starView.ph`. Podczas pracy szczególną uwagę poświęcałam skryptom `dbconnect.php`, `functions.php`, `superStarView.php`, `starView`, `config.php`, `customDbConfig.php` oraz `index.php`. W momencie, gdy zabrakłoby jednego z nich bądź przez przypadek zostało by coś zostało wprowadzone nieprawidłowo, strona WWW nie działałaby w poprawny sposób.

Podobnie jak we wcześniejszym interfejsie, tak również i ten przystosowywałam do pracy z dwoma bazami danych, PostgreSQL i DB2. Swoją pracę rozpoczęłam od wprowadzenia zmian związanych z połączeniami z bazą danych. W tym celu użyłam zmiennej `$basetype`, której przypisany jest ciąg znaków „db2” w przypadku pracy z bazą DB2, natomiast w przeciwnym razie strona WWW pracuje z bazą PostgreSQL. W momencie, kiedy użytkownik chce zmienić bazę, musi to zrobić zarówno w skrypcie `functions.php`, jak i `customDbConfig.php`.

Główne zapytanie do bazy ma charakter dynamiczny i został stworzony w odrębnej funkcji „`createQuery`”. W celu zachowania przejrzystości kodu, wprowadziłam dwa oddzielne zapytania - jedno z nich odpowiada bazie DB2 natomiast drugie PostgreSQL.

Zapytanie używane w DB2 jest bardziej złożone. Zawiera ono w sobie wyrażenie `ROWNUMBER() OVER ()`, które jak już sama nazwa wskazuje w pierwszej kolejności numeruje kolejne wiersze. Wyrażenie to umieszczane jest w zagnieżdżonym zapytaniu, które znajduje się w klauzuli `FROM`. W celu posortowania wierszy wykorzystuje wyrażenie `ORDER BY`, które umieściłam w `OVER ()`. Zapytanie od bazy wygląda w następujący sposób:

```
if ($basetype=='db2')
{
    $query = 'SELECT '.$rowid.' '.$select.' FROM (SELECT
ROWNUMBER() OVER('.$query_orderby.') AS '.$rowid.'
```

```

'.$select_count.' FROM '.$main1.'Superstar WHERE (1=1)
'.$query_where.') AS temp '.$query_limit.' ' ;
}
else
{
    $query = 'SELECT '.$select.' FROM '.$main.'Superstar
WHERE (1=1) '.$query_where.' '.$query_orderby.'
'.$query_limit.' ' ;
}

```

Najlepiej różnicę pomiędzy budową jednego jak i drugiego zapytania pokazuje przykład zapytania, które wyświetla wyniki dla gwiazd o zadanej liczbie pomiarów (no\_measurements), współrzędnych niebieskich (ra, dec), a także jasności (mag):

```

SELECT rowid, id, ra, dec, mag, sigma_mag, name,
no_measurements, amp, period, star_type, star_class,
other_id, other_class, pi_id FROM (SELECT ROWNUMBER() OVER()
AS rowid, id, ra, dec, mag, sigma_mag, name, no_measurements,
amp, period, star_type, star_class, other_id, other_class,
pi_id FROM main.Superstar WHERE (1=1) AND ( ra BETWEEN 12 AND
24 ) AND ( dec BETWEEN -70 AND 90 ) AND ( mag BETWEEN 12 AND
15) AND ( no_measurements BETWEEN 80 AND 1000000) ) AS temp
WHERE ( rowid BETWEEN 0 AND 100 )

```

Takie samo zapytanie do bazy PostgreSQL wygląda następująco:

```

SELECT id, ra, dec, mag, sigma_mag, name, no_measurements,
amp, period, star_type, star_class, other_id, other_class,
pi_id FROM Superstar WHERE (1=1) AND ( Superstar.ra BETWEEN
12 AND 24 ) AND ( Superstar.dec BETWEEN -70 AND 90 ) AND
(Superstar.mag BETWEEN 12 AND 15) AND (
Superstar.no_measurements BETWEEN 80 AND 1000000) ORDER BY
superstar.mag LIMIT 100 OFFSET 0

```

Zmienne które widoczne są w pierwotnej formie zapytania, odpowiadają wyrażeniom, zawartym w funkcji createQuery, które oczywiście stanowią część zapytania. Forma tworzenia zapytania pozwoliła na skonstruowanie \$query w prawidłowy sposób, a także wpłynęło na przejrzystość kodu.

Nie wszystkie funkcje PHP współdziałające z bazą PostgreSQL mają swoje odpowiedniki także dla bazy DB2. Przykładem takiej funkcji może być funkcja `pg_fetch_all`, która zwraca wszystkie rekordy, otrzymane z wyniku zapytania [8]. Związku z tym należało zbudować funkcję, która będzie odpowiadać zadanej funkcji. Stworzona przeze mnie - funkcja `db2_fetch_all`, wygląda w następujący sposób:

```
function db2_fetch_all($result)
{
    $i=0;
    while ($row_dabeData = db2_fetch_assoc($result))
    {
        $row_dabeData = lowercaseIndexing($row_dabeData);
        $dbData[$i] = $row_dabeData;
        $i++;
    }
    return $dbData;
}
```

Głównym zadaniem tej funkcji jest tworzenie wektora, w którym znajdują się wszystkie odpowiedzi pochodzące z zapytania `$result`. Wektor ten nie jest numerowany liczbami lecz nazwami kolumn. Niestety nazwy te wpisane są automatycznie poprzez wbudowaną funkcję `db2_fetch_assoc` dużymi literami. Stanowi to pewną trudność, ponieważ interfejs do katalogu gwiazd zbudowany był w oparciu o indeksowanie małymi literami, gdyż właśnie tak formatuje indeksy funkcja `pg_fetch_all` współpracująca z bazą PostgreSQL. Jednakże i tutaj znalazło się wyjście, dzięki funkcji `lowercaseIndexing` mogą zamieniać duże litery na małe [8].

Tak wygląda kod po zastosowaniu funkcji `db2_fetch_assoc` bez użycia `lowercaseIndexing`:

```
{[0]=>array(24){["ID"]=>int(280062)["RA"]=>float(0.0020753)
["DEC"]=>float(-
1.5072836)["MAGNITUDE"]=>float(5.4185141819288E-
315)["SIGMA_MAG"]=>float(5.2196167766767E-315)["NAME"]=>
string(13)"000007-
130.4["NO_MEASUREMENTS"]=>int(142)["CAMID"]=>
int(2)["PERIOD"]=>NULL["AMP"]=>NULL["HJD_T0"]=>NULL["SSTAR_ID
"]=>int(445563)["MIN_MAG"]=>float(5.4068556358333E-315)
["FD_NAME"]=> float(5.4265322794226E-315) ["STAR_CLASS"]=>
```

```

NULL [ "OTHER_ID" ]=>
NULL[ "OTHER_CLASS" ]=>NULL[ "TYCHO_ID" ]=>int(-
1)[ "ASAS_ID" ]=>int(1)[ "GCVS_ID" ]=>NULL[ "OTHERSTAR" ]=>int(2567
098)[ "LAST_CHECKED" ]=>NULL [ "FIELD" ]=>NULL[ "QUALITY" ]=>NULL
}}

```

Natomiast dzięki funkcji lowercaseIndexing indeksy zmieniają się automatycznie w sposób następujący:

```

{[0]=>array(24){ [ "id" ]=> int(280062) [ "ra" ]=>
float(0.0020753) [ "dec" ]=>float(-
1.5072836)[ "magnitude" ]=>float(5.4185141819288E-315)
[ "sigma_mag" ]=>float(5.2196167766767E-315)[ "name" ]=>
string(13)"000007-0130.4"[ "no_measurements" ]=>int(142)
[ "camid" ]=>int(2)[ "period" ]=>NULL[ "amp" ]=>NULL[ "hjd_t0" ]=>NULL
[ "sstar_id" ]=>int(445563)[ "min_mag" ]=>float(5.4068556358333E
-315) [ "fd_name" ]=>float(5.4265322794226E-
315)[ "star_class" ]=>NULL
[ "other_id" ]=>NULL[ "other_class" ]=>NULL[ "tycho_id" ]=>int(-1)
[ "asas_id" ]=>int(-1)[ "gcv_s_id" ]=>NULL[ "otherstar" ]=>
int(2567098) [ "last_checked" ]=> NULL [ "fiels" ]=> NULL
[ "quality" ]=> NULL } }

```

Stworzenie funkcji db2\_fetch\_all oraz użycie funkcji lowercaseIndexing ułatwiło modyfikację interfejsu, ponieważ w początkowej wersji miały nastąpić zmiany w pliku Smarty<sup>11</sup>, który jest bardzo wrażliwy na wielkość liter. Tak więc znalezienie sposobu na zmianę indeksów w stworzonej przez mnie funkcji miało ogromny wpływ na działanie całego interfejsu w przypadku pracy z DB2.

Podczas migracji interfejsu do katalogu gwiazd oprócz funkcji db2\_fetch\_all musiałam dopisać wiele linii kodu umożliwiających pracę z bazą IBM DB2. W większości przypadków nie wystarczyła sama zamiana funkcji PHP, lecz należało wprowadzić odpowiednie zmienne, przekształcić strukturę pliku itp. Była to praca bardzo złożona i wymagająca zachowania jednolitej konstrukcji we wszystkich plikach składających się na całość interfejsu.

---

<sup>11</sup> Smarty – obiektowa biblioteka skryptów, które służą do tworzenia szablonów dla aplikacji PHP.

Go to another database Pi Stars Browser Database: Pełna baza 2006-2007 (# of objects = 9.7 mln)

Show star:   % - replaces any string  
\* - replaces one character

### Searching:

<input type="checkbox"/> Ra	min <input type="text" value="00:00:00"/>	max <input type="text" value="24:00:00"/>	?
<input type="checkbox"/> Dec	min <input type="text" value="-90:00:00"/>	max <input type="text" value="90:00:00"/>	?
<input type="checkbox"/> Search around	Ra <input type="text" value="12:00:00"/>	Dec <input type="text" value="00:00:00"/>	?
		radius [arcsec] <input type="text" value="1"/>	?
<input type="checkbox"/> Magnitude	min <input type="text" value="0"/>	max <input type="text" value="15"/>	?
<input type="checkbox"/> Error [mag]	min <input type="text" value="0"/>	max <input type="text" value="1"/>	?
<input type="checkbox"/> No. observations	min <input type="text" value="80"/>	max <input type="text" value="1000000"/>	?
<input type="checkbox"/> Amplitude [mag]	min <input type="text" value="0"/>	max <input type="text" value="100"/>	?
<input type="checkbox"/> Period [days]	min <input type="text" value="0"/>	max <input type="text" value="100"/>	?
<input checked="" type="checkbox"/> Advanced			?

<< previous page      Sort by: magnitude      Stars per page: 100      next page >>

List of Stars						
Id	Magnitude	Error [mag]	Amplitude [mag]	Period [days]	No. observations	Class
<a href="#">121451-1530.3</a>	14.056	0.173			80	

Copyright (C) PI of the Sky Experiment, Warsaw, 2005-2007 Web programming: Marek Białopiotrow

Rysunek 15: Interfejs WWW do wyszukiwania gwiazd.

### Light Curve

<b>Experiment</b>	<i>Pi of the Sky</i>
<b>Database</b>	Pełna baza 2006-2007 (# of objects = 9.7 mln)
<b>Superstar Id</b>	2774966
<b>Star Id</b>	8315842
<b>Star Name</b>	121451-1530.3
<b>Ra</b>	12.2475h (12h:14m:51s)
<b>Dec</b>	-15.5009° (-15°30'03")
<b>Magnitude [ min / avg / max ]</b>	15.082 / 13.961 / 13.131
<b>Error [mag]</b>	0.368 <sup>m</sup>
<b>Delta magnitude</b>	1.951 <sup>m</sup>
<b>Number of measurements</b>	52
<b>Period</b>	0.000 days (0.000 h)
<b>Amplitude [mag]</b>	0.000 <sup>m</sup>
<b>t<sub>0</sub></b>	2453250 HJD
<b>Camera Id</b>	3
<b>Star Id on the other camera</b>	<a href="#">2077647</a>

**Advanced**

To continue searching come back to: [Pi Browser](#)

Rysunek 16: Strona WWW poświęcona pojedynczej gwiazdzie.

## Podsumowanie

Celem niniejszej pracy była migracja interfejsu WWW służącego do przeglądania katalogu gwiazd w eksperymencie „ $\pi$  of the Sky”. Swoją pracę rozpoczęłam od wprowadzenia zmian dotyczących połączenia z bazą PostgreSQL oraz DB2. Połączenie w obydwu przypadkach zakończyło się powodzeniem. W kolejnym etapie przekształciłam skrypty PHP uzupełniając je o funkcje pozwalające na współpracę z bazą IBM DB2. Aktualnie strona WWW służąca do przeglądania katalogu gwiazd działa zarówno na bazie DB2 jak i na PostgreSQL, a użytkownik interfejsu nie musi dysponować wiedzą na temat rodzaju używanej bazy. Nie jest również w stanie odróżnić na jakiej bazie pracuje. Kod napisany jest w sposób prosty oraz zrozumiały. Niestety na serwerze WWW, na którym pracowałam biblioteka JpGraph odpowiedzialna za generowanie wykresów nie była odpowiednio skompilowana do pracy ze środowiskiem Zend Core2, co w rezultacie spowodowało błędy w ładowaniu map i krzywych blasku. Problemem tym zespół „ $\pi$  of the Sky” zajmie się w późniejszym terminie.

Dodatkowo, przy okazji nauki języka SQL i PHP, stworzyłam osobny interfejs umożliwiający kontrolowanie temperatury, w jakich wykonywane są ekspozycje w eksperymencie „ $\pi$  of the Sky”. Zarówno temperatura chipu CCD ustawiona przez komputer i temperatura odczytana bezpośrednio z nagłówka zdjęcia zapisane są w bazie danych. Stworzona przeze mnie strona internetowa za pomocą skryptów PHP łączy się z bazą danych i odpytuje ją między innymi o zapisane tam temperatury. Strona ta okazała się być na tyle funkcjonalna, że została włączona w system monitoringu detektora.

## **PODZIĘKOWANIE**

Chciałabym serdecznie podziękować wszystkim członkom projektu „ $\pi$  of the Sky” za umożliwienie mi pracy w ciekawym projekcie, a w szczególności dr hab. Lechowi Mankiewiczowi dzięki któremu było to możliwe. Szczególny wyraz wdzięczności chciałabym skierować do mgr Katarzyny Małek, która zaproponowała mój temat pracy licencjackiej, jak również przez cały okres przygotowywania zarówno części praktycznej jak i merytorycznej służyła swoją pomocą i cennymi radami. Wskazówki i pomoc mgr Marka Biskupa i mgr Marcina Sokołowskiego, pozwoliły napisanie kodu w sposób przejrzysty i prosty.

## BIBLIOGRAFIA

- [1] M. Biskup, *Poszukiwanie gwiazd zmiennych w eksperymencie „Pi of the Sky”*, Uniwersytet Warszawski, Wydział Fizyki, praca magisterska, Warszawa 2007
- [2] Katarzyna Ewa Malek, *System wyszukiwania gwiazd nowych i zmiennych w danych pochodzących z przeglądów nieba w eksperymencie „Pi of the Sky”*, Uniwersytet Stefana Kardynała Wyszyńskiego, Szkoła Nauk Ścisłych, praca magisterska, Warszawa 2006
- [3] <http://grb.fuw.edu.pl/>
- [4] L. W. Piotrowski, *Poszukiwania błysków światła widzialnego towarzyszących pozagalaktycznym błyskom gamma*, Uniwersytet Warszawski, Wydział Fizyki, praca magisterska, Warszawa 2005
- [5] A. Burd, M. Cwiok, H. Czyrkowski, R. Dabrowski, W. Dominik, M. Grajda, M. Husejko, M. Jegier, A. Kalicki, G. Kasprowicz, K. Kierzkowski, K. Krupska, K. Kwiecinska, L. Mankiewicz, K. Nawrocki, B. Pilecki, L.W. Piotrowski, K. Pozniak, R. Romaniuk, R. Salanski, M. Sokolowski, D. Szczygiel, G. Wrochna, W. Zabolotny, „*Pi of the Sky*” - *all-sky, real-time search for fast optical transients*, *New Astronomy*, Volume 10, Issue 5, April 2005
- [6] R. Stones, N. Matthew, *Od podstaw. Bazy danych i PostgreSQL*, Helion, Warszawa 2002
- [7] R. F. Chong, C. Liu, S. F. Qi, D. R.. Snow, tłumaczenie: A. Chodkowska-Gyurics, L. Borkowski, *Zrozumieć DB2. Nauka na przykładach*, Wydawnictwo Naukowe PWN, Warszawa 2006
- [8] <http://pl2.php.net/>
- [9] <http://wapedia.mobi/pl>