



**AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE**

Rozbudowa i uaktualnienie oprogramowania systemu GGSS detektora ATLAS TRT

Arkadiusz Kasprzak

Jarosław Cierpich

Opiekun: dr hab. inż. Bartosz Mindur, prof. AGH

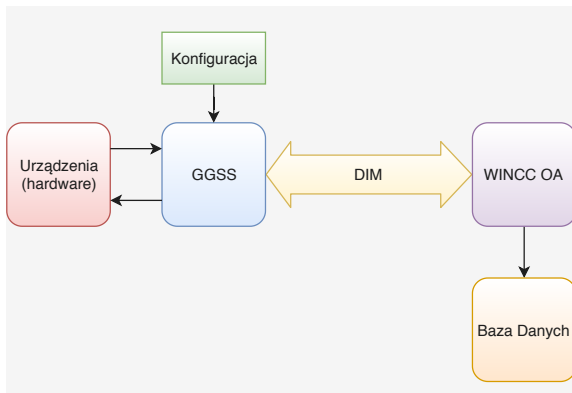
Kraków, styczeń 2020



- ➊ Wprowadzenie
- ➋ Cel pracy
- ➌ Założenia
- ➍ Wykonane zadania
- ➎ Podsumowanie



- System Stabilizacji Wzmocnienia Gazowego (GGSS) - zintegrowany z systemem kontroli detektora ATLAS (DCS - Detector Control System)
- umożliwia poprawne działanie detektora TRT (będącego częścią detektora ATLAS)
- składa się z warstwy sprzętu (np. multiplekser) oraz oprogramowania
- warstwa oprogramowania napisana w języku C++
- system budowania projektu oparty o narzędzie CMake oraz pomocnicze skrypty (Bash, Python)
- jasny podział na moduły (biblioteki), ale początkowa architektura o płaskiej strukturze
- kontrola wersji - przed rozpoczęciem prac SVN



Rys. 1: Wysokopoziomowa architektura systemu GGSS ¹

¹Wszystkie wykorzystane w prezentacji rysunki pochodzą z omawianej pracy (Arkadiusz Kasprzak, Jarosław Cierpich „Rozbudowa i uaktualnienie oprogramowania systemu GGSS detektora ATLAS TRT”)



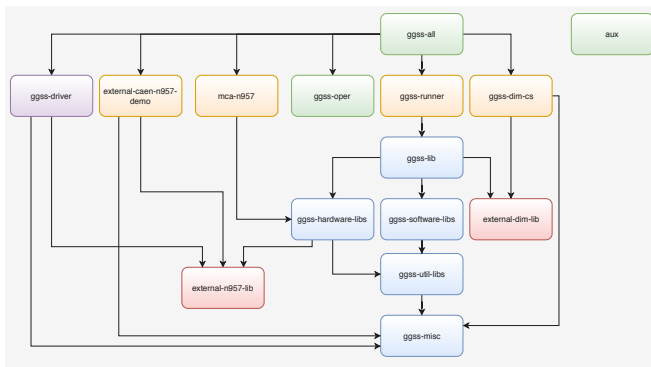
- zdobycie wiedzy na temat infrastruktury informatycznej w CERN
- utworzenie wygodnego, nowoczesnego środowiska do zarządzania projektem informatycznym
- utworzenie prostego w rozwoju, intuicyjnego systemu budowania oprogramowania opartego o narzędzie CMake
- modularyzacja projektu
- migracja do systemu kontroli wersji Git
- automatyzacja procesu budowania i dystrybucji projektu
- przygotowanie dokumentacji



- praca wykonywana przez dwie osoby - konieczne odpowiednie zaplanowanie i podział prac
- powinna istnieć możliwość zbudowania projektu za pomocą infrastruktury dostępnej w CERN (ograniczenia nałożone na wersje bibliotek i narzędzi)
- minimalne zmiany w kodzie źródłowym aplikacji - zmiany w ramach pracy dotyczyć powinny narzędzi ułatwiających pracę z projektem, nie samego projektu

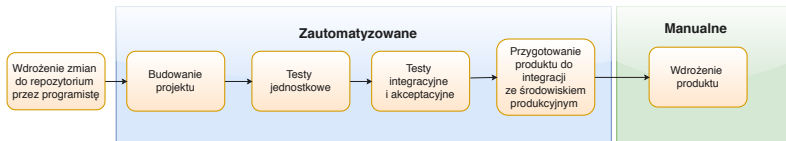


- migracja projektu do systemu kontroli wersji Git i zmiana architektury na hierarchiczną z wykorzystaniem mechanizmu tzw. *submodułów*
- przygotowanie infrastruktury wspomagającej zarządzanie projektem opartej na funkcjonalnościach portalu Gitlab
- zmiana systemu budującego aplikację (zagwarantowanie niezależności poszczególnych modułów)
- umożliwienie automatycznego pobierania nowej wersji pakietu DIM
- zastosowanie podejścia *Continuous Integration and Delivery*
- przygotowanie maszyny wirtualnej wspomagającej proces ciągłej integracji
- przygotowanie dokumentacji projektu w języku angielskim



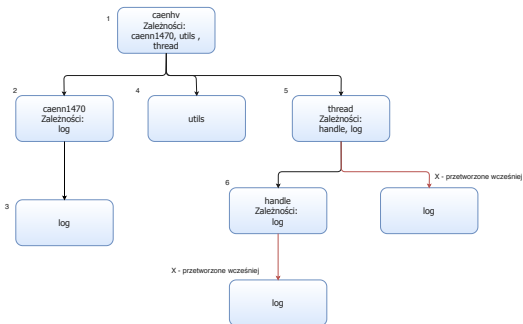
Rys. 2: Nowa architektura projektu GGSS

- ciągła integracja i ciągłe dostarczanie
- szeroko stosowana praktyka programistyczna
- wykorzystanie usługi Gitlab CI



Rys. 3: Przykładowy schemat działania podejścia CI/CD

- przystosowany do modularnej, hierarchicznej architektury
- każda biblioteka przetwarzana raz
- oparty na narzędziu CMake



Rys. 4: Przykład działania systemu budującego



- migracja projektu GGSS do systemu kontroli wersji Git zakończyła się powodzeniem
- podczas migracji wprowadzone zostały zmiany ułatwiające pracę z projektem
- wszystkie postawione cele zostały zrealizowane
- projekt w wersji finalnej został przetestowany w środowisku produkcyjnym
- wyniki testów są pozytywne

Dziękuję za uwagę.