R G B WH GR BL Grey Scale #13 C M Y K
DANES-PICTA.COM
A 1 2 3 4 5 6 M 8 9 10 11 12 13 14 15 B 17 18 19

AKADEMIA OBRONY NARODOWEJ

CENTRUM SYMULACJI
I KOMPUTEROWYCH GIER WOJENNYCH

ANALIZA MOŻLIWOŚCI INTEGRACJI SYSTEMU DOWODZENIA SZAFRAN Z SYSTEMEM SYMULACYJNYM JTLS

BIBLIOTEKA GŁÓWNA - ARCHIWUM
Nr ewid.
Akademii Obrony Narodowej
59581

WARSZAWA 2005

Colour Chart #13
Inches
Centimetres
Blue Cyan Green Yellow Red Magenta White 3/Color Black
DANES-PICTA.COM

AKADEMIA OBRONY NARODOWEJ

CENTRUM SYMULACJI
I KOMPUTEROWYCH GIER WOJENNYCH

BIBLIOTEKA GŁÓWNA - ARCHIWUM
Nr ewid. 59581
II
Akademii Obrony Narodowej

ANALIZA MOŻLIWOŚCI INTEGRACJI SYSTEMU DOWODZENIA *SZAFRAN* Z SYSTEMEM SYMULACYJNYM *JTLS*

płk dr inż. Ryszard WIELEBA
mjr mgr inż. Sławomir BIENIEK

Warszawa **2005**

Pracę naukowo-badawczą **ANALIZA MOŻLIWOŚCI INTEGRACJI SYSTEMU DOWODZENIA SZAFRAN Z SYSTEMEM SYMULACYJNYM JTLS** wykonał zespół pracowników naukowo-badawczych CENTRUM SYMULACJI I KOMPUTEROWYCH GIER WOJENNYCH pod kierownictwem płk. dr. inż. Ryszarda WIELEBA w składzie:

płk dr inż. Ryszard WIELEBA
mjr mgr inż. Sławomir BIENIEK

Poszczególni członkowie zespołu opracowali

- płk dr inż. Ryszard WIELEBA wprowadzenie, Rozdział 2, 4, 6 -współudział, zakończenie oraz nadzór merytoryczny pracy
- mjr mgr inż. Sławomir BIENIEK –Rozdział 1, 3, 5, 7, wprowadzenie, Rozdział 2, 4, 6 współudział

Spis treści

WPROWADZENIE **3**

1. SYSTEM JTLS **6**

1.1 GENEZA SYSTEMU 6
1.2 OGÓLNE WŁAŚCIWOŚCI SYSTEMU 9
1.2.1 Zakres konfliktu *9*
1.2.2 Skład sił *9*
1.2.3 Poziom szczegółowości *10*
1.2.4 Teren *11*
1.2.5 Symulacja sterowana zdarzeniami *11*
1.2.6 Upływ czasu *13*
1.2.7 Dane wejściowe *13*
1.2.8 Udział użytkownika *14*
1.2.9 Dane wyjściowe *14*
1.2.10 Ograniczenia *14*
1.3 DZIAŁANIA BOJOWE MODELOWANE W SYSTEMIE 15
1.4 STRUKTURA LOGICZNO-PROGRAMOWA SYSTEMU 19
1.4.1 Programy tworzenia scenariusza *23*
1.4.2 Programy inicjujące i konfiguracyjne *26*
1.4.3 Program zdarzeń walki (CEP – Combat Events Program) *32*
1.4.4 Podstawowe programy użytkowe *32*

2. SYSTEM SZAFRAN **41**

2.1 PRZEZNACZENIE 41
2.2 PODSTAWOWE ZADANIA PZSD ZT 42
2.3 STRUKTURA SYSTEMU 45
2.4 OPROGRAMOWANIE PZSD ZT 46
2.4.1 Podsystem Wymiany Dokumentów *47*

3. PORÓWNANIE DANYCH PRZETWARZANYCH W SYSTEMIE JTLS I SZAFRAN **50**

4. MOŻLIWOŚCI INTEGRACJI **53**

4.1 POCZTA ELEKTRONICZNA 53
4.1.1 Wiadomości niesformalizowane *54*
4.1.2 Wiadomości sformalizowane *54*
4.1.3 Pakiet IRIS *55*
4.2 OPEN ACCESS PROGRAMS 62
4.2.1 Scenario Data Client (SDC) *64*
4.2.2 Message Processing Client (MPC) *64*
4.2.3 Order Entry Client (OEC) *65*
4.3 INTEGRACJA – SPECJALIZOWANE INTERFEJSY 66
4.4 INTEGRACJA NA POZIOMIE BAZ DANYCH 67
4.5 STANDARD HLA 68
4.5.1 Infrastruktura programowa systemów w środowisku HLA *75*
4.5.2 Uwarunkowania integracji systemów w ramach standardu HLA *76*

5. PROBLEMY DO ROZWAŻENIA **80**

6. ANALIZA PORÓWNAWCZA ROZWIĄZAŃ **81**

7. KIERUNKI ROZWOJU SYSTEMÓW **83**

7.1 JTLS 83
7.2 SZAFRAN 86
7.3 WNIOSKI 91

ZAKOŃCZENIE **93**

LITERATURA **94**

Wprowadzenie

Nowoczesne pole walki wymaga umiejętnego koordynowania i synchronizowania działań wielu szczebli dowodzenia, rodzajów wojsk i sił zbrojnych. Zapewnienie tego wymaga posiadania możliwości szybkiego i terminowego przekazywania rozkazów i meldunków. Wdrażanie nowego sprzętu łączności znakomicie usprawnia proces dowodzenia lecz nie zmienia go w sposób jakościowy. Zmiany jakościowe mogą być uzyskane dopiero w wyniku automatyzacji określonych procesów i funkcji dowodzenia na stanowiskach dowodzenia i w pododdziałach bezpośrednio prowadzących działania bojowe. Zapewni ona integrację wszystkich elementów ugrupowania bojowego zaangażowanych w prowadzone działania. Osiągnięcie celu działań w nowych warunkach oraz użycie nowoczesnych systemów rozpoznania i rażenia jest praktycznie niemożliwe przy korzystaniu z tradycyjnych środków dowodzenia i łączności. Bez wdrożenia zautomatyzowanego systemu dowodzenia wojska lądowe nie będą w stanie prowadzić skutecznych działań w warunkach współczesnego i perspektywicznego pola walki.

Proces automatyzacji dowodzenia związany jest ściśle z rozwojem i zastosowaniem technologii komputerowych oraz systemów łączności cyfrowej. Zautomatyzowane systemy dowodzenia budowane są w oparciu o najnowocześniejsze technologie informatyczne, wykorzystując profesjonalne rozwiązania cywilne dostępne na rynku. Dotyczy to zarówno sprzętu jak i oprogramowania, w tym również wykorzystywania najnowszych technologii i rozwiązań internetowych.

Wojska Lądowe nie posiadają aktualnie kompleksowego zautomatyzowanego systemu dowodzenia ogólnowojskowymi związkami taktycznymi, oddziałami i pododdziałami. Realizowane dotychczas przedsięwzięcia w zakresie automatyzacji systemów dowodzenia dotyczyły wybranych obszarów zastosowania technik informatycznych na wyższych szczeblach dowodzenia. Dla niższych szczebli dowodzenia (szczebel taktyczny) prowadzone prace dotyczyły zautomatyzowanych systemów kierowania środkami walki poszczególnych rodzajów wojsk (OPL, artyleria). Podobna sytuacja ma miejsce w wielu krajach.

Prace w dziedzinie automatyzacji systemów dowodzenia i kierowania środkami walki w Wojskach Lądowych są od wielu lat prowadzone w różnych krajach. System niemiecki HEROS jest realizowany od ponad 20 lat. Obecnie trwają prace nad nową wersją systemu HEROS-2/1 oraz systemami niższych szczebli dowodzenia BMS (ang. *Battlefield Management Systems*) i CWCS (ang. *Command and Weapon Control Systems*). System

CWCS jest wdrażany począwszy od 1997 roku. System BMS, aktualnie nazywany FAUST jest przeznaczony dla stanowisk dowodzenia szczebla batalionu i poniżej. System Heros ma za zadania również integrację między innymi systemów obrony przeciwlotniczej, rozpoznania, artylerii oraz współdziałanie z podobnymi systemami innych krajów.

System amerykański ATCCS (ang. *Army Tactical Command and Control System*) jest opracowywany od początku lat 80-tych, a od 1992r. modyfikowany i testowany. Jednym z jego elementów jest MCS (ang. *Maneuver Control System*), który zapewnia dowodzenia oddziałami ogólnowojskowymi, integrację ze specjalizowanymi systemami dowodzenia oraz współdziałanie w układzie międzynarodowym. Dla automatyzacji dowodzenia szczebla brygady i poniżej opracowywany jest system FBCB2 (ang. *Force XXI Battle Command – Brigade and Below*). Dla automatyzacji dowodzenia w wojskach lądowych aktualnie trwają prace nad zintegrowanym systemem ABCS (ang. *Army Battle Command System*).

Nad podobnymi rozwiązaniami prowadzone są prace również w innych krajach, np. we Francji – SICF i SIR, w Norwegii – NORCCIS, we Włoszech - SIACCON, w Danii – DACCIS, w Hiszpanii – SIMACET i SGB, czy też w Holandii – ISIS.

Współczesne pole walki stawia bardzo wysokie wymagania. Wymusza utrzymywanie systemów walki w ciągłej gotowości do skutecznego działania oraz ciągłe ich doskonalenie. To z kolei rodzi potrzebę prowadzenia wszechstronnych badań tych systemów w celu poznania sposobów ich funkcjonowania oraz istotnych warunków i czynników mających wpływ na przebieg i skuteczność działań bojowych. Immanentną cechą środowiska działania systemów walki jest aktywne przeciwdziałanie przeciwnika. Jest to złożony problem, ponieważ każda z walczących stron dąży zawsze do ukrycia informacji przed przeciwnikiem lub wprost do jego dezinformowania.

Wysokie tempo prowadzonych działań, narastający strumień odbieranych informacji i krótki czas na podjęcie decyzji, to jedna z wielu cech działań militarnych. Ponadto procesy zachodzące na polu walki mają charakter procesów losowych, a na skutki podejmowanych decyzji mają wpływ czynniki niezależne od osoby podejmującej decyzję, w tym oczywiście również elementy kontrolowane i sterowane przez przeciwnika.

W tym kontekście przed ludźmi stawia się bardzo trudne zadania i wysokie wymagania, szczególnie w stosunku do członków elementów decyzyjnych, czyli dowództw i sztabów. Rodzi to potrzebę ciągłego szkolenia i doskonalenia ich umiejętności kierowania działaniami bojowymi. Tu jednak powstaje problem stworzenia w czasie pokoju rzeczywistych warunków, jakie występują podczas działań wojennych. Elementy takich działań można ćwiczyć podczas zajęć poligonowych, jednak z ekonomicznego punktu widzenia staje się to

dość problematyczne ze względu na ich wysoki koszt. Odpowiedzią na te potrzeby stały się systemy symulacyjne, wykorzystujące modele działań bojowych do stworzenia w środowisku komputerowym otoczenia dla elementów decyzyjnych podobnego, z jakim mogą się one spotkać na polu walki oraz systemy dowodzenia zapewniające informacyjne wspomaganie dowództw i sztabów.

W trakcie ćwiczeń CAX, realizowanych przez Centrum Symulacji i Komputerowych Gier Wojennych, przewidywane jest wykorzystanie dwóch systemów symulacji walki JTLS oraz ZŁOCIEŃ a także systemu wspomagania dowodzenia (o ile będzie to możliwe) SZAFRAN.

System symulacji pola walki JTLS jest szeroko stosowany w działalności szkoleniowej w armiach państw NATO w trakcie ćwiczeń CAX. W Wojsku Polskim jest to system nowy, obecnie rozpoznawany pod kątem możliwości i wdrażany do eksploatacji. Natomiast system symulacyjnego wspomagania szkolenia operacyjnego ZŁOCIEŃ, umożliwiający szkolenie dowództw przede wszystkim na szczeblu brygady i dywizji, jest systemem nowym, opracowanym w WAT i przewidywanym do wdrożenia w CSiKGW w 2006 roku.

W przyszłości planowane jest również wdrożenie systemu SZAFRAN, który jest wykorzystywany do wspomagania dowodzenia związku taktycznego, a będącego na etapie badań realizowanych przez PIT. System ten jest przeznaczony do wspomagania realizacji procesów dowodzenia na szczeblu korpusu, dywizji, brygad, batalionów Wojsk Lądowych. SZAFRAN jest produktem nowym, który jeszcze nie został przetestowany w żadnym z krajów członkowskich NATO, nieznane są więc możliwości jego współpracy z systemem JTLS.

W związku z powyższym powstaje problem, w jaki sposób można dokonać integracji obu systemów tak, aby stworzyć warunki dla możliwie najsprawniejszego przeprowadzenia ćwiczeń komputerowych organizowanych przez CSiKGW.

Praca niniejsza jest próbą podjęcia wyzwania stawianego zarówno przez konieczność wykorzystywania zautomatyzowanych systemów dowodzenia, jak i możliwości dostarczanych przez wykorzystanie systemów symulacyjnych w szkoleniu dowództw i sztabów. Zawiera ona analizę możliwości wykorzystania systemu JTLS z systemem wspomagania dowodzenia SZAFRAN w trakcie ćwiczeń CAX. W tym celu zespół zrealizował szereg badań laboratoryjnych w celu rozpoznania istoty, możliwości funkcjonalnych oraz architektury technologicznej obu systemów, co było podstawą do realizacji niniejszego opracowania.

1. System JTLS

1.1 Geneza systemu

Prace nad wytworzeniem systemu JTLS zainicjowane zostały w 1981 roku z inicjatywy dowódcy nowoutworzonego wówczas U.S. Readiness Command (USREDCOM), który poszukiwał narzędzia pozwalającego na wsparcie jego sztabu przy przeprowadzaniu analizy i oceny planów operacji połączonych. Do współpracy przy inicjacji przedsięwzięcia zaproszona została Naval Postgraduate School (NPS).

W wyniku uzgodnień między USREDCOM i Operations Research Department NPS powołany został Contingency Planning Institutew przy NPS. Pierwsze jego zadanie zostało określone jako REDCOM Contingency Planning Task. Realizacja tego zadania rozpoczęła się od analizy istniejących modeli symulujących działania na poziomie strategiczno-operacyjnym. Cykl spotkań, w których uczestniczyli przedstawiciele NPS, Vector Research Incorporated (VRI), Army War College (AWC) oraz Lawrence Livermore National Laboratory (LLNL), poświęcony został ocenie możliwości wykorzystania istniejących modeli do zaspokojenia potrzeb USREDCOM. Do bardziej szczegółowej analizy możliwości wybrano następujące cztery modele:

1. Vector – opracowany przez VRI nieinteraktywny deterministyczny model walki powietrznej i lądowej. Ćwiczące siły lądowe były reprezentowane przez podstawowe jednostki ćwiczące wielkości batalionu.

2. Janus – opracowany przez LLNL wykorzystywany przede wszystkim do oceny skutków użycia taktycznej broni jądrowej. Modelował również konwencjonalną walkę modelując działania jednostek od poziomu drużyny do poziomu brygady. Był modelem interaktywnym i mógł odzwierciedlać działania dwóch stron konfliktu. Posiadał graficzny interfejs użytkownika pozwalający na zobrazowanie efektów symulacji oraz wprowadzanie informacji wpływających na jej przebieg.

3. Warfare Environment Simulator (WES) – opracowany przez Naval Ocean Systems Center (NOSC) w San Diego;

4. McClintic Theater Model (MTM) – opracowany przez Freda McClintica z AWC. MTM był modelem interaktywnym, ukierunkowanym na odzwierciedlenie walki

lądowej. Posiadał szczątkową reprezentację walki powietrznej, ale nie uwzględniał elementów działań morskich.

Wniosek z analizy ww. modeli był taki, że nie istniało wówczas żadne narzędzie umożliwiające przeprowadzenie gry wojennej lub analizy, które byłoby opracowane z uwzględnieniem odpowiedniego poziomu szczegółowości pozwalającego na modelowania całego spektrum działań połączonych w sposób skoordynowany.
Jako model, na bazie którego można byłoby rozwijać narzędzie potrzebne USREDCOM, wybrano MTM.

W roku 1982 USREDCOM podpisało pierwszy kontrakt z Jet Propulsion Laboratory (JPL). Podwykonawcą w tym projekcie została firma ROLANDS & ASSOCIATES Corporation (R&A). Celem projektu było rozszerzenie możliwości modelu działań powietrznych wykorzystywanego w MTM. Po demonstracji rozwiązania oraz wniosków z przeprowadzonych prac w lecie 1982 podpisano kolejny kontrakt. Dotyczył on opracowania nowego interaktywnego modelu działań połączonych. Fundusze na finansowanie projektu pochodziły od USREDCOM, AWC oraz Army Concepts Analysis Agency (ACAA).
Każda z tych instytucji miała własne założenia i wymagania odnośnie modelu. USREDCOM było zainteresowane przede wszystkim narzędziem przydatnym przy tworzeniu, analizie i ocenie połączonych i koalicyjnych planów operacyjnych (ang. *joint and combined (coalition) operation plans*), AWC był zainteresowany głównie narzędziem wsparcia szkolenia słuchaczy, natomiast ACAA interesowało analizowanie różnych alternatywnych struktur sił. Od stycznia do kwietnia 1983 roku przeprowadzono kompletną analizę wymagań oraz opracowano projekt koncepcyjny. Szczegółowa specyfikacja projektowa została przedstawiona w kwietniu i zatwierdzona w maju. Trzy podstawowe założenia, na których oparto projekt systemu brzmiały następująco:

- model miał być sterowany danymi – oznaczało to, że żadne parametry symulacji nie miały być zakodowane na stałe w systemie. Cała logika systemu miała opierać się na danych parametrycznych definiowalnych w bazie danych.
- system miał zostać zakodowany przy użyciu tylko jednego języka programowania. Miało to zmniejszyć koszty wytworzenia i utrzymania systemu.
- konstrukcja systemu miała być zoptymalizowana dla jednej wybranej platformy systemowej. Pominięcie elementów pozwalających na możliwość uruchamiania modelu na różnych platformach systemowych pozwalała na efektywne wykorzystanie możliwości

wybranej platformy docelowej. Przy tak złożonym zagadnieniu, jakim było modelowanie działań połączonych, pozwalało to na wykorzystanie mocy przetwarzania do uzyskania większej wierności modelu.

W późniejszym czasie okazało się, że założenie dotyczące użycia jednego języka programowania jest niecelowe. JPL udało się przekonać sponsorów do przyjęcia dwóch języków programowania jako narzędzi tworzenia JTLS. Jako język symulacji wybrano SIMSCRIPT, do tworzenia narzędzi wspomagających wykorzystano język C.

Pierwsze ćwiczenia z wykorzystaniem JTLS pod nazwą „Yellow Breeches" przeprowadzono w październiku 1984 roku.

W roku 1985 powstał program nowoczesnego wsparcia dla planowania (ang. *MAPP – Modern Aids to Planning Program*) i nadzór na pracami przy rozwijaniu JTLS przejął Joint Analysis Directorate (JAD), który został później przemianowany na JCS/J-8 (ang. *JCS – Joint Chiefs of Staff*). W tym samym roku rozpoczęto przygotowania do wsparcia przez JTLS ćwiczeń „Determined Warrior 85" przeprowadzanych przez Warrior Preparation Center (WPC). Na podstawie wniosków z dwutygodniowych ćwiczeń system rozszerzono o możliwość restartu systemu w oparciu mechanizm punktu kontrolnego (ang. *checkpoint*), którego dane zapisywane były w plikach ASCII.

W przeprowadzonych w pierwszej połowie 1986 roku przez WPC ćwiczeniach „Determined Warrior 86" wykorzystywany był JTLS w wersji z mechanizmem punktu kontrolnego zapisywaneo w plikach ASCII oraz prototypem Post Procesor Program.

W 1987 roku system rozszerzono o modelowanie działań morskich i desantu morskiego. Ponadto narzędzia wspomagające rozszerzono o SDS (ang. Scenario Development System) oraz IMT.

W roku 1988 pierwsze ćwiczenia oparte na JTLS przeprowadziło EUCOM.

W roku 1989 zakończono prace nad modułem ATO-G (ang. *Air Tasking Order – Generator*). Model rozbudowano o reprezentację transportu barkami, sieci drogowej, rurociągów na potrzeby modelowania działań logistycznych.

W kolejnych latach model rozszerzano o reprezentację przydzielania i wydzielania jednostek (1990), działań podwodnych (1991), sił specjalnych (1991), koalicje więcej niż dwóch stron konfliktu (1992). W roku 1994 do zobrazowania sytuacji wykorzystano GIAC opracowany przez Los Alamos National Laboratory zamiast poprzednio wykorzystywanego systemu GraphOver. W roku 1997 SDS został zastąpiony przez nowy moduł DDS (ang.

Database Development System). Prace nad interfejsem HLA w JTLS rozpoczęto w roku 1998.

1.2 Ogólne właściwości systemu

JTLS (ang. *Joint Theater Level Simulation*) jest interaktywnym systemem umożliwiającym prowadzenie wspartej komputerowo symulacji wielostronnych działań bojowych, w ramach której modelowane są działania powietrzne, lądowe i morskie wraz ze wsparciem logistycznym, działaniami sił specjalnych oraz wywiadu i rozpoznania. Jest często wykorzystywany jako model wspomagający szkolenie. JTLS jest niezależny od teatru działań i nie wymaga umiejętności programowania.

1.2.1 Zakres konfliktu

System koncentruje się głównie na konwencjonalnych operacjach połączonych i koalicyjnych na szczeblu strategiczno-operacyjnym określonych w dokumencie „Universal Joint Task List" wytworzonym przez Połączone Sztaby (ang. *Joint Staff*). JTLS modeluje bezpośrednio operacje powietrzne, lądowe, morskie, desantu morskiego i sił specjalnych. Model wspiera w ograniczonym zakresie symulację efektów użycia broni jądrowej i chemicznej, konflikty o niskiej intensywności (ang. *low intensity conflict*) oraz działania poprzedzające konflikt (ang. *pre-conflict operations*).

1.2.2 Skład sił

W modelu mogą być reprezentowane wielostronne koalicje sił powietrznych, lądowych, morskich oraz sił specjalnych. Dostarcza również mechanizmy umożliwiające reprezentację sił cywilnych (ang. *civilian forces*) oraz nie biorących udziału w walce (ang. *non-combatant forces*) wewnątrz sektorów działania (ang. *sectors of interest*).

Model pozwala na reprezentację maksymalnie dziesięciu stron konfliktu. Każda strona może być podzielona na dowolną liczbę frakcji. Lojalność frakcji stron może być dynamicznie zmieniana w trakcie gry. Relacje pomiędzy stronami są asymetryczne i również mogą być zmieniane w trakcie gry.

Nazwy stron, frakcji i kolory używane do wyświetlania sił przynależnych do stron są konfigurowalne poprzez modyfikację bazy danych.

1.2.3 Poziom szczegółowości

Wszystkie procesy zostały zaprojektowane tak, aby umożliwić niezależną od doktryny elastyczną implementację. Podstawowymi bytami reprezentowanymi w modelu są jednostki (ang. *units*) i obiekty (ang. *targets*). Baza konfigurowalna przez użytkownika definiuje wielkość jednostek, kategorie zaopatrzenia oraz obiekty ważne z militarnego punktu widzenia, które mają być reprezentowane. Małe jednostki reprezentowane na dużym poziomie szczegółowości, które mogą być niszczone jako cele (ang. *high resolution target systems*) uzupełniają bardziej zagregowane struktury jednostek. Jednostki są reprezentowane zarówno na wysokim poziomie szczegółowości, jak i na niskim – z użyciem agregacji. Baza danych może zostać przygotowana tak, aby w ramach zagregowanej struktury jednostki, opisywała wybrane systemy na koniecznym poziomie szczegółowości. Dlatego różnorodność tych poziomów w modelu jest rzeczą normalną.

Działania walki powietrznej mogą być dowodzone jako zagregowane wielostronne zespołowe zadania lotnicze (ang. *multi-sided mission packages*), ale są rozstrzygane na poziomie elementarnym dla każdego obiektu wchodzącego w skład misji przy wykorzystaniu współczynników prawdopodobieństwa trafienia (ang. *probability of hit*) pH i prawdopodobieństwa zniszczenia (ang. *probability of kill*) pK. Wyniki walki są rozstrzygane stochastycznie.

Działania artylerii przeciwlotniczej są modelowane na poziomie elementarnym dla każdego obiektu przy wykorzystaniu współczynników prawdopodobieństwa podjęcia walki (ang. *probability of engage*) pE i prawdopodobieństwa zniszczenia (ang. *probability of kill*) pK dla każdej misji. Wyniki walki są rozstrzygane stochastycznie.

Użycie rakiet powietrze-ziemia (ang. *air-to-ground*), woda-woda, ziemia-ziemia, ziemia-woda i woda-ziemia (ang. *surface-to-surface*) oraz artylerii (w tym wsparcia artylerii okrętowej) jest modelowane na poziomie elementarnym dla każdego obiektu. Wykorzystywane są w tym przypadku współczynniki pH i pK dla pocisków naprowadzanych (ang. *precision-guded munitions*) oraz funkcje gęstości celów dla broni o oddziaływaniu obszarowym (ang. *area weapon coverage*). Wyniki są rozstrzygane stochastycznie dla pocisków naprowadzanych oraz deterministycznie dla broni o rażeniu obszarowym.

Model walki lądowej wykorzystuje modele walki Lanchestera do zsumowania efektów oddziaływania ognia bezpośredniego oraz środków wsparcia bezpośredniego, które nie zostały użyte w wyraźnie postawionych zadaniach wspierania ogniem bezpośrednim. Wyniki są rozstrzygane deterministycznie.

1.2.4 Teren

Wykorzystanie map cyfrowych w formacie ADRG (ang. *ARC (Arc second Raster Chart/map) Digitized Raster Graphics*) oraz danych o terenie w formacie WVBS pozwala na symulowanie działań na dowolnym obszarze i tym samym czyni JTLS możliwym do użycia na całym świecie. Do wytworzenia plików opisu terenu opartych na podziale obszaru na siatkę sześciokątów może zostać wykorzystany system przygotowania terenu (ang. *TPS – Terrain Preparation System*). Algorytmy modelu zakładają, że siatka sześciokątów jest nałożona na mapę z odwzorowaniem stożkowym wiernokątnym Lamberta (ang. *Lambert conformal conic projection*). Odwzorowanie to może doprowadzić do niepożądanych zniekształceń, jeżeli mapa będzie obejmowała prostokątny obszar o boku przekraczającym 2000 mil morskich z którejkolwiek strony.

Odwzorowanie terenu oparte na siatce sześciokątów agreguje lokalne charakterystyki terenowe i środowiskowe: przejezdność, wzniesienie terenu oraz skażenie chemiczne i radioaktywne. Drogi odwzorowane są poprzez łączenie ze sobą środków sześciokątów. Rurociągi i linie kolejowe są odwzorowywane jako niezależne od sześciokątów sieci połączonych ze sobą węzłów. Rzeki i linie brzegowe odwzorowywane są na krawędziach sześciokątów.

Obiekty punktowe modyfikują przejezdność poprzez dostarczanie celowego polepszenia podstawowych (bazowych) warunków terenowych. Na przykład takie obiekty, jak most, tunel czy punkt izolacji (ang. *interdiction point*), mogą być wprost reprezentowane i obierane jako cele. Niszczenie tych obiektów wpływa na zależną od nich reprezentację terenu, na których się znajdują. Podobnie stacje pomp oraz stacje kolejowe mogą być wprost reprezentowane i obierane jako cele. Ich niszczenie wpływa na możliwości związanych z nimi rurociągów i sieci kolejowych.

1.2.5 Symulacja sterowana zdarzeniami

Model JTLS symuluje działania sił połączonych na szczeblu strategiczno-operacyjnym i w związku z tym odzwierciedla kluczowe aspekty walki powietrzno-lądowej oraz operacje wspierające sił morskich.

Z punktu widzenia klasyfikacji związanej wpływem czasu na wynik symulacji, która wyróżnia następujące modele:

- dynamiczne - są to takie modele, w których czynnik czasu ma kluczowe znaczenie. Stan systemu zmienia się dzięki upływowi czasu symulacyjnego, właściwości i atrybuty

systemu są zależne od wartości czasu symulacyjnego; wynik jest zależny od czasu trwania symulacji;

- statyczne - w tego rodzaju modelach czas nie wpływa na wynik, zegar symulacji nie jest potrzebny; stan systemu nie jest zależny od czasu i atrybuty systemu nie zmieniają się wraz z czasem symulacji;

w JTLS zastosowano model dynamiczny.

Podstawowymi składnikami systemu dynamicznego są operacje (ang. *activities*). Istnieją dwie ważne cechy operacji:

- zajmują czas,
- potencjalnie mogą zmienić stan systemu.

Z punktu widzenia klasyfikacji związanej z podstawą czasu, na której występują zdarzenia modelu, która wyróżnia następujące modele:

- model ciągły w czasie – czas płynie w sposób ciągły, zegar modelu zwiększa swoją wartość o stałe przyrosty, w zakresie liczb rzeczywistych;
- model dyskretny w czasie – czas płynie skokowo, zegar modelu zwiększa się okresowo poprzez skoki z jednej wartości całkowitej na następną;
- model statyczny – jeżeli podstawa czasu jest nie istotna dla modelu;

JTLS implementuje model dyskretny w czasie.

Modele dyskretne w czasie mogą być taktowane czasem (ang. *time-stepped*) lub sterowane zdarzeniami (ang. *event-driven*). W symulacji sterowanej zdarzeniami zmiany stanu następują w określonych momentach, w których dochodzi do oddziaływania pomiędzy elementami składowymi systemu.

W trakcie konstruowania modelu systemu należy zidentyfikować i opisać operacje w sposób, który umożliwi odtworzenie w trakcie symulacji zależne od czasu zachowanie się rzeczywistego systemu. Operacje muszą zostać zamodelowane w taki sposób, że stan systemu będzie się zmieniał prawidłowo gdy wystąpią. To wymaganie narzuca dodatkowe rygory wymagające poprawnego odzwierciedlenia charakterystyk operacji oraz sekwencji ich występowania, które spowoduje, że kolejność ich wykonywania będzie odzwierciedlała ich rzeczywistą sekwencję wystąpień.

Każda operacja w systemie jest ograniczana przez dwa zdarzenia o pomijalnie krótkim czasie trwania: uruchomienie i zatrzymanie operacji. Zdarzenie (ang. *event*) jest najprostszym elementem opisu operacji. Istnieją dwie główne własności zdarzenia:

Każda operacja, na którą składa się ciąg zdarzeń, jest w systemie ograniczona startem i zatrzymaniem operacji. Zdarzenie stanowi najprostszy element opisu operacji. Istnieją dwie główne cechy zdarzenia:

- występuje ono w pewnej określonej chwili;
- pojawia się natychmiastowo i przez bardzo krótki (pomijalnie mały) czas.

Zmiany w systemie, które pojawiają się kiedy operacje są uruchamiane lub zatrzymywane, mają związek raczej ze zdarzeniami niż wykonywanymi operacjami. Ponieważ to właśnie te zdarzenia powodują wszystkie znaczące zmiany stanów systemu, upływ czasu pomiędzy zdarzeniami nie musi być dokładnie odwzorowywany. Zamiast tego upływ czasu jest sterowany sekwencjami zdarzeń, przesuwając się zawsze do chwili wystąpienia następnego znaczącego zdarzenia.

1.2.6 Upływ czasu

Użytkownik określa pożądany stosunek czasu symulacyjnego do czasu rzeczywistego. Maksymalne osiągalne przyspieszenie zależne jest od możliwości sprzętu komputerowego oraz złożoności scenariusza. Projekt systemu zakładał zdolność utrzymywania przyspieszenia przynajmniej w stosunku 4 do 1.

1.2.7 Dane wejściowe

Jako dane wejściowe system JTLS uznaje informacje dotyczące terenu (ang. *terrain*), uzbrojenia (ang. *weapons*), przemieszczenia (ang. *movement*), tabel współczynników zadawanych strat (ang. *attrition tables*) wykorzystywanych w równaniach Lanchestera, charakterystyki jednostek oraz czasy pojawiania się jednostek w teatrze działań (ang. *TPFDD – Time-Phased Force Deployment Data*). Żadne dane nie są na sztywno zakodowane w systemie, dlatego tworzenie bazy danych jest złożonym i czasochłonnym przedsięwzięciem przy rozpoczynaniu od zera. Model jest dystrybuowany wraz przykładową bazą danych zawierającą wiele elementów nadających się do wykorzystania przy opracowywaniu własnych baz. Opracowanie nowej bazy danych zajmuje od 6 do 8 miesięcy.

1.2.8 Udział użytkownika

Zaimplementowany w JTLS model jest interaktywny, w związku z czym wymaga decyzji człowieka aby zarządzać procesami i obiektami. Jako interfejs służący do komunikacji z modelem, w celu wprowadzania rozkazów i graficznego zobrazowania sytuacji w grze symulacyjnej, wykorzystywany jest GIAC (ang. *Graphical Input Aggregate Control*). Automatyczne wprowadzanie rozkazów zadań lotniczych (ang. *ATO – Air Tasking Order*) realizowane jest przez translator rozkazów zadań lotniczych (ang. *ATOT – Air Tasking Order Translator*).

1.2.9 Dane wyjściowe

Każda stacja robocza GIAC umożliwia wyświetlanie graficzne zagregowanych informacji o jednostkach lądowych, misjach lotniczych, okrętach nawodnych i podwodnych oraz wydzielonych małych jednostkach (ang. *HRU – High Resolution Units*). Użytkownik ma także udostępnione interaktywne narzędzie zarządzania informacją IMT (ang. *IMT – Information Management Tool*) otrzymywaną na bieżąco. IMT dostarcza, konfigurowane przez użytkownika, arkusze danych pozwalające na uzyskanie aktualnych informacji o działaniach sił, ich postawie i zdolnościach bojowych.

Użytkownicy JTLS otrzymują wiadomości i raporty dotyczące ruchów, strat oraz stanu logistyki sił własnych, jak również o zdolności bojowej przeciwnika i raporty zbiorcze rozpoznania. Użytkownik na każdej stacji roboczej może wybrać podgląd wiadomości w postaci niesformalizowanej lub w formacie USMTF (ang. *U.S. Message Text Format*). Wiadomości otrzymywane z systemu mogą być wysłane pocztą elektroniczną z wykorzystaniem protokołu SMTP (ang. *SMTP – Simple Mail Transfer Protocol*). Istnieje również możliwość elektronicznego zasilenia danymi systemów C4I (np.: GCCS – Global Command Control System, JOTS – Joint Operational Tracking System, JMCIS – Joint Military Command Information System) oraz systemów zobrazowania (np.: COP – Common Operational Picture).

Dodatkowo istnieje możliwość odtwarzania wyświetlanej sytuacji graficznej zapisywanej co ustalony interwał (funkcjonalność GAWS).

1.2.10 Ograniczenia

Założenia modelowe JTLS ograniczają jego użyteczność przy reprezentacji zagregowanych informacji na szczeblu poniżej batalionu, aczkolwiek wspierane jest modelowanie takich małych jednostek wydzielonych (ang. *HRU - High Resolution Unit*), jak

punkty regulacji ruchu (ang. *Traffic Control Points*) oraz patrole rozpoznawcze (ang. *reconnaissance patrols*).

1.3 Działania bojowe modelowane w systemie

CEP symuluje wykonywanie działań powietrznych, lądowych, logistycznych, morskich oraz wywiadu i rozpoznania. Szczegóły sposobu modelowania poszczególnych działań opisane są w odpowiadających im rozdziałach „JTLS Analyst's Guide".

Działania powietrzne:

- samoloty wczesnego wykrywania i naprowadzania (ang. *AWACS – Airborne Warning and Control System*);
- walka elektroniczna (ang. *EC – Electronic Combat*);
- tankowanie w powietrzu (ang. *air refueling*);
- eskorta (ang. *escort*);
- samoloty zwalczania środków obrony przeciwlotniczej (ang. *WW – Wild Weasel*);
- działania defensywne w ramach walki o przewagę w powietrzu (ang. *defensive counter-air*) w tym również alarm naziemny (ang. *ground alert*) oraz dyżurowanie w powietrzu (ang. *airborne alert*);
- atak na cele naziemne lub nawodne (ang. *surface attack*);
- bliskie wsparcie z powietrza (ang. *CAS – Close Air Support*) w tym również alarm naziemny (ang. *ground alert*) oraz dyżurowanie w powietrzu (ang. *airborne alert*);
- rozpoznanie, z wykorzystaniem jednostek uzbrojonych i nieuzbrojonych (ang. *reconnaissance, both armed and unarmed*), w tym również rozpoznanie rejonów wskazanych (ang. *DSAs – Directed Search Areas*);
- zrzut z powietrza jednostek lub zaopatrzenia (ang. *airdrop of units or supplies*);
- transport lotniczy jednostek lub zaopatrzenia (ang. *airlift of units or supplies*);
- zespołowe misje lotnicze (ang. *Air Mission Packages*);
- minowanie narzutowe z powietrza minami lądowymi lub morskimi (ang. *air emplacement of land or sea mines*);
- zwalczanie okrętów podwodnych (ang. *Anti-Submarine Warfare*);
- kierowanie z powietrza zadaniami na lądzie (ang. *OAG – Orbiting Air Ground Missions*);

- angażowanie w czasie rzeczywistym powietrznych systemów kierowania i zmienianie zadań ofensywnych na lądzie (ang. *Real Time Commitment of OAG and diversion of surface attack missions*);
- angażowanie się w zadania lotnicze w celu uniemożliwienia wykonania zadań lotniczych przez przeciwnika (ang. *Manual commitment of air missions to intercept enemy air missions*);
- identyfikacja swój-obcy (ang. *IFF – Identification, Friend or Foe*) w tym również identyfikacja jako nieznany;
- szczegółowe zasady użycia siły (ang. *ROE – Rules of Engagement*) w misjach powietrznych (ang. *Air-to-Air*) i działaniach artylerii przeciwlotniczej (ang. *Surface-to-Air*) w tym specyficzne zasady użycia siły dla obszaru operacyjnego (ang. *OPAREA*);
- ustanawianie łączy z innymi frakcjami w celu współdzielenia informacji o sytuacji powietrznej;
- działania poszukiwawcze i ratownicze (ang. *SAR – Search and Rescue*) w stosunku do załóg zestrzelonych statków powietrznych;
- działania zintegrowanego systemu obrony powietrznej (ang. *IADS – Integrated Air Defense System*).

Działania lądowe:

- straty powodowane przez ogień bezpośredni (ang. *attrition caused by direct fire*);
- straty powodowane przez ogień pośredni (ang. *attrition caused by indirect fire*);
- ogień przeciwartyleryjski (ang. *counter-battery fire*);
- przemieszczanie (ang. *movement*);
- atak (ang. *attack*);
- obrona doraźna (ang. *hasty defence*);
- obrona przygotowana (ang. *prepared defence*);
- działania opóźniające (ang. *delay*);
- wycofanie (ang. *withdrawal*);
- efekty opóźniania i straty powodowane przez miny lądowe (ang. *delaying effects of, and attrition caused by, land mines*);
- bliskie wsparcie z powietrza (ang. *CAS – Close Air Support*);
- zwalczanie artylerii przeciwnika (ang. *counter-battery*);

- systemy dowodzenia i łączności (ang. *C3 – Command, Control, and Communications*);
- straty i opóźnienia powodowane przez skażenie radioaktywne i chemiczne (ang. *attrition and delays caused by nuclear and chemical contamination*);
- inżynieria wojskowa, w tym budowanie mostów;
- przydzielanie, wydzielanie i odtwarzanie jednostek;
- zakładanie zapór minowych przez artylerię, statki powietrzne i działania jednostek (ang. *emplacement of mines by artillery, air drop, or unit action*);
- niszczenie i naprawa obiektów na rozkaz;
- siły specjalne;
- koalicje wielu frakcji;
- szczegółowe zasady użycia siły w walce lądowej (wraz z lotnictwem wojsk lądowych), w tym również specyficzne dla obszaru operacyjnego i granic państw;
- koordynacja współpracy cywilno-wojskowej (ang. *Civil Affairs*) i operacje psychologiczne (ang. *PSYOPS – Psychological Operations*);
- wydzielanie małych jednostek (ang. *HRU – High Resolution Units*) do przeprowadzania różnorodnych działań taktycznych.

Rozpoznanie:

- dane wywiadowcze dostarczane poprzez okresowy raport zbiorczy;
- bezpośrednie rozpoznanie przez wydzielone grupy (ang. HRU), zwiad lotniczy (ang. *airborne reconnaissance*) i zasoby organiczne jednostki (ang. *unit organic resources*);
- raporty z obszarów działań sił lądowych oraz z misji lotniczych (ang. *spot/mission reports from ground forces and airborne missions*);
- oddzielne postrzeganie pola bitwy przez każdą ze stron konfliktu;
- selektywne współdzielenie danych rozpoznawczych między stronami konfliktu;
- częściowe rozpoznanie obiektów i jednostek (ang. *partial intelligence of targets and units*);
- zewnętrzne i narodowe źródła informacji (ang. *external/national asset resources*);
- sieci łączności.

Logistyka:

- skutki działań logistycznych na zdolność bojową, w tym również obsługi (ang. *maintenance*) oraz przywracania sprawności (ang. *return to service*) systemom walki;
- zaautomatyzowana obsługa uzupełniania zaopatrzenia (ang. *automated resupply*), w tym: wyliczanie wielkości potrzeb zaopatrzeniowych (ang. *stockage objective*), kontrola minimalnych wielkości zapasów (ang. *reorder level*), generowanie zapotrzebowania (ang. *backorder*);
- przypisywanie jednostkom wielorakich klas zaopatrzenia (ang. *multiple support unit assignment*) w zależności od kategorii zaopatrzenia (ang. *supply category*);
- bezpośrednie decydowanie przez gracza o zaopatrzeniu (ang. *supply*), uzupełnianiu zaopatrzenia (ang. *resupply*) i wzajemnym wsparciu w utrzymywaniu minimalnych wielkości zapasów (ang. *cross leveling*) jednostek;
- przemieszczanie zaopatrzenia koleją, barkami lub ciężarówkami;
- transport jednostek koleją, barkami lub konwojami ciężarówek;
- zaopatrzenie rurociągami (ang. *pipeline operation*);
- ukrywanie i odzyskiwanie (ang. *caching and recovery*) własnego zaopatrzenia oraz przechwytywanie (ang. *capture*) zaopatrzenia przeciwnika;
- działania przeładunkowe (ang. *port operations*) przy transporcie powietrznym i morskim, zarówno wewnątrz teatru działań jak i poza nim;
- sieci transportowe (wraz z mostami i tunelami);
- losowe awarie i uszkodzenia sprzętu.

Działania morskie:

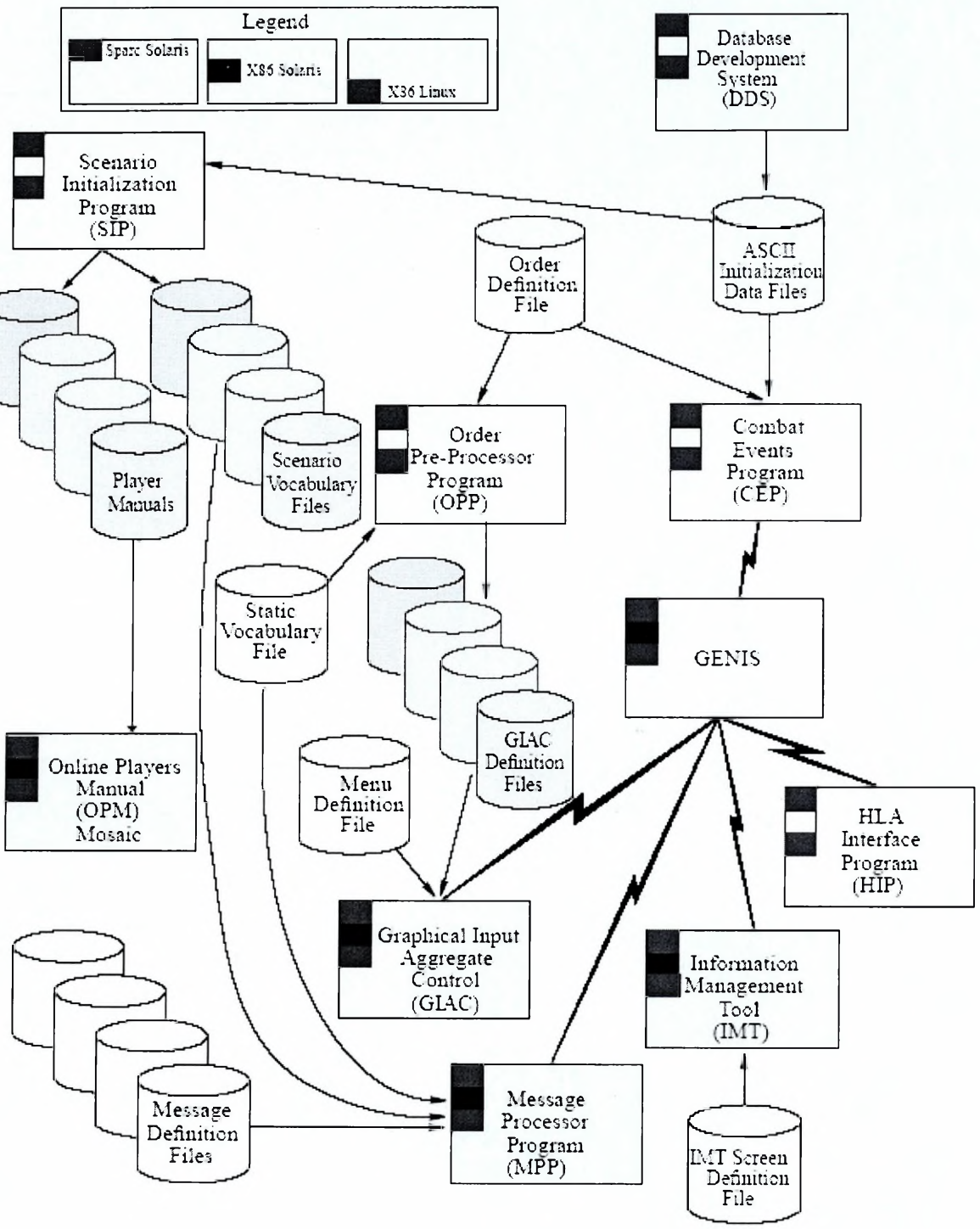
- działania powietrzne (wymienione w sekcji modelowanych działań powietrznych) w oparciu o wykorzystanie lotniskowców (ang. *carrier-based air activities*);
- transport morski (ang. *sealift*);
- pokrycie radarowe (ang. *serach radar coverage*);
- użycie rakiet klasy woda-woda i woda-ziemia (ang. s*urface-to-surface*) i artylerii pokładowa (ang. *naval gunfire*);
- wsparcie ogniowe artylerii okrętowej (ang. *naval reinforcing fire*), w tym ogień przeciwartyleryjski;
- siły zadaniowe/formacje (ang. *task organization/formations*), w skład których wchodzi wiele frakcji i stron;

- desant morski (ang. *amphibious assault*) oraz operacje załadunku i wyładunku z wykorzystaniem środków desantu morskiego (ang. *amphibious pickup/extraction*);
- działania minowe (ang. *mine warfare operations*), w tym stawianie zapór minowych, trałowanie, zadawane straty;
- przemieszczanie sił zadaniowych/formacji (ang. *task organization/formations movement*), w tym łączenie formacji (ang. *formations merging*);
- działania okrętów podwodnych (ang. *submarine operations*);
- patrolowanie akwenów (ang. *area patrol*);
- głębokość mórz i oceanów, strefy głębokości oraz wpływ głębokości na operacje.

Dodatkowo w systemie JTLS modelowane jest oddziaływanie chorób na personel. Może nabawić się choroby i umrzeć lub odzyskać zdrowie. Status oddziaływania chorób na jednostkę jest przedstawiany w ramach procesu raportowania okresowego.

1.4 Struktura logiczno-programowa systemu

System JTLS jest konglomeratem powiązanych ze sobą programów. Cześć z nich jest widoczna bezpośrednio, jako samodzielne komponenty posiadające własny interfejs użytkownika, inne zaś mogą być wykorzystywane jako zestaw udostępnionych bibliotek funkcji.



Rysunek 1. Struktura programowa JTLS
/źródło: JTLS Executive Overview v.2.7/

Ze względu na funkcje oraz czas wykorzystania programów w całym cyklu wykorzystania systemu JTLS, można dokonać podziału programów na niżej wymienione grupy:

1. Programy tworzenia scenariusza i narzędzia wspomagające:
   - system opracowywania bazy danych *(*ang. *DDS – Database Development System);*
   - program weryfikacji scenariusza *(*ang. *SVP – Scenario Verification Program);*
   - program modyfikacji terenu (ang. *TMU – Terrain Modification Utility);*

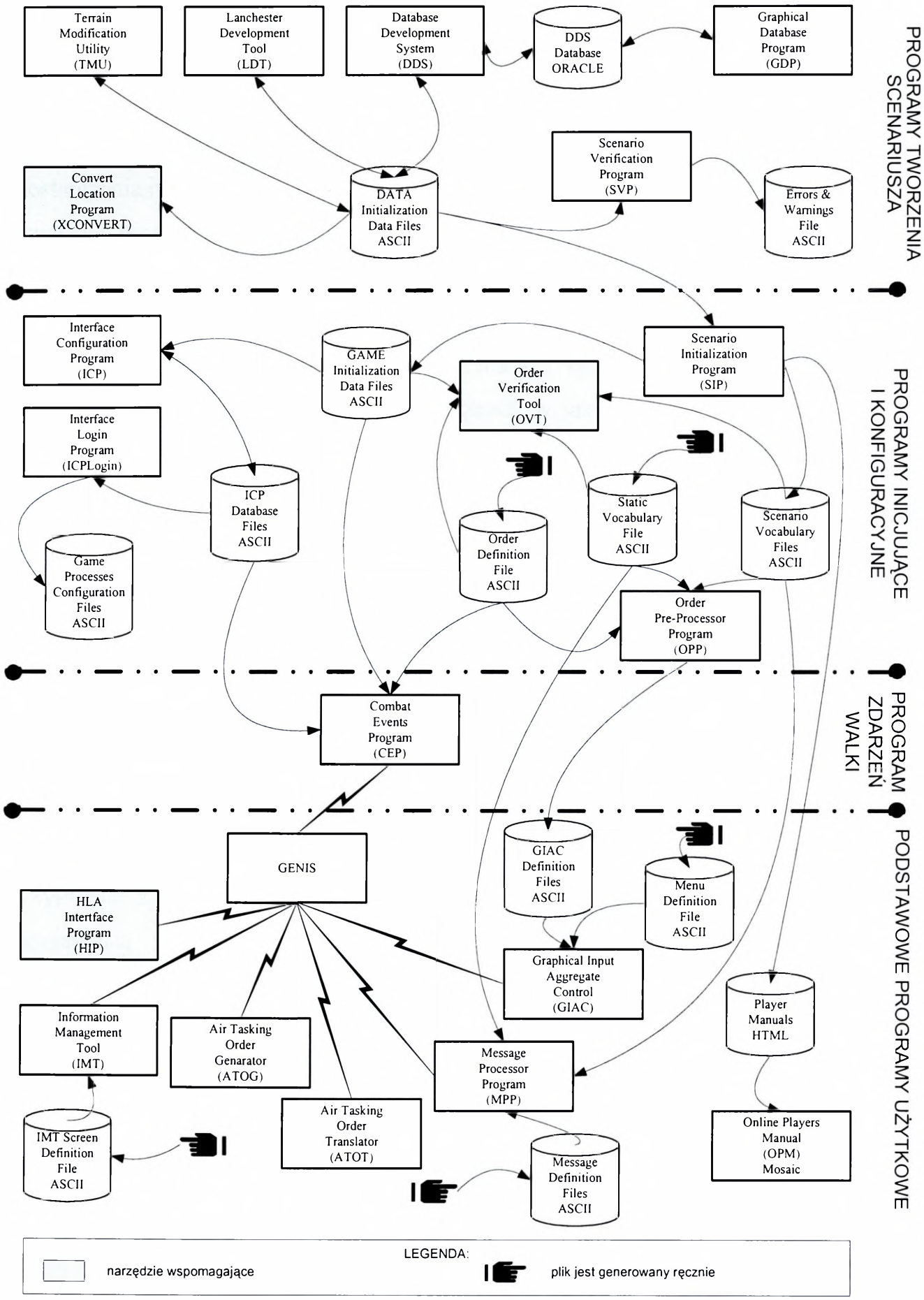
- program modyfikacji współczynników równań Lanchester'a *(ang. LDT – Lanchester Development Tool);*
- program obsługi graficznej bazy danych *(ang. GDP – Graphical Database Program);*
- program konwersji położenia *(ang. XCONVERT – Convert Location Program);*
- program interfejsu HLA *(ang. HIP – High Level Architecture Interface Program);*
- program weryfikacji rozkazu *(ang. OVT – Order Verification Tool).*

2. Programy inicjujące i konfiguracyjne:
   - program inicjujący scenariusz *(ang. SIP – Scenario Initialization Program);*
   - program konfiguracji interfejsu *(ang. ICP – Interface Configuration Program);*
   - program logowania *(ang. ICP Login – Interface Login Program)*
   - program wstępnego przetwarzania rozkazów *(ang. OPP – Order Pre-Processor).*

3. Program zdarzeń walki *(ang. CEP – Combat Events Program).*

4. Podstawowe programy użytkowe:
   - program sterujący graficznymi danymi wejściowymi *(ang. GIAC – Graphical Input Aggregate Control);*
   - generator rozkazów zadań lotniczych *(ang. ATOG – Air Tasking Order Generator);*
   - translator rozkazów zadań lotniczych *(ang. ATOT – Air Tasking Order Translator);*
   - program przetwarzania wiadomości *(ang. MPP – Message Processor Program);*
   - narzędzia zarządzania informacją *(ang. IMT – Information Management Tool);*
   - podręcznik gracza *(ang. OPM – Online Player's Manual).*

Schemat struktury programowej JTLS uwzględniający wszystkie ww. programy, ich powiązania oraz podział przedstawia Rysunek 1 Struktura programowa JTLS. Nie uwzględniono na nim, dla zachowania czytelności, powiązań między ICP, ICPLogin i innymi programami JTLS (patrz rysunek 7 – Powiązania między ICP, ICPLogin i innymi programami).



Rysunek 2 Struktura programowa JTLS
/źródło: opracowanie własne/

1.4.1 Programy tworzenia scenariusza

Narzędzia przygotowania scenariusza i wspomagające są używane do:

- dostępu i czytania plików danych inicjujących ASCII;
- dostarczania informacji o spójności i poprawności;
- działania na danych i zapisywanie nowych, uaktualnionych wersji danych w formacie plików ASCII.

Ponieważ interfejsem wymiany danych pomiędzy narzędziami fazy przygotowania danych a narzędziami inicjalizacji i uruchomienia gry są pliki ASCII, dobrze wyszkolony personel może mieć dostęp do plików inicjujących w jakimkolwiek edytorze. Korzystanie z dostępu do plików danych w taki sposób zalecany jest tylko dla doświadczonego personelu zapoznanego z formatem plików danych.

1.4.1.1 System opracowywania bazy danych (DDS – Database Development System)

DDS jest podstawowym narzędziem tworzenia, rozwijania i modyfikacji baz danych. Jest używany do stworzenia nowej bazy danych, modyfikacji istniejącej bazy danych lub wyszukiwania w bazie informacji spełniających określone kryteria - np. DDS może być użyty do uzyskania z bazy danych listy jednostek określonego typu. DDS jest aplikacją wykorzystującą funkcjonalność serwera ORACLE zbudowaną przy użyciu narzędzi ORACLE Forms/Reports Developer 6i.

Pliki danych ASCII określające inicjującą bazę danych dla wybranego scenariusza są wczytywane przez DDS i wypełniają zbiór tabel ORACLE. Pliki te są tak zaprojektowane, aby były zgodne ze strukturą bazy danych JTLS. Użytkownik DDS uzyskuje dostęp do tych tabel aby modyfikować lub przeszukiwać zawarte w niej dane.

Kiedy modyfikacja jest zakończona, dane są składowane i tworzony jest nowy zbiór plików inicjujących dla opracowywanego scenariusza.

DDS uruchamia się poprzez wybór pierwszej opcji: „Tworzenie lub zmiana bazy danych scenariusza" (ang. *Prepare or Alter a Scenario Database*) z głównego menu JTLS. Z podmenu wybiera się: „Menu opracowywania bazy danych" (ang. *Access the Database Development System Menu*).

1.4.1.2 Program weryfikacji scenariusza (SVP – Scenario Verification Program)

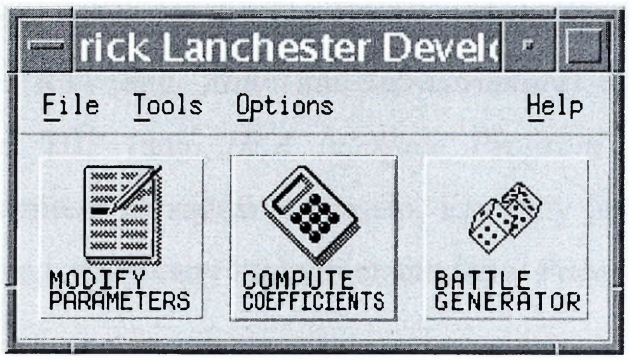
SVP jest częścią rodziny narzędzi przedstawionych jako SIP (ang. *Scenario Initialization Program*). Zadaniem SVP jest weryfikacja, czy dane wprowadzone dla zadanego scenariusza są zgodne, poprzez sprawdzenie spójności między zmiennymi. SIP wczytuje pliki bazy danych scenariusza, które są wykorzystywane przez CEP (ang. *Combat Events Program*) i przechowuje dane w takich samych strukturach danych, jakie są używane przez model w czasie wykonywania symulacji. To powoduje, że cała potrzebne kontrola spójności jest wydajna i dokładna. To zapewnia, że np.: jednostki morskie są umieszczone w heksagonach morskich, a jednostki lądowe w lądowych. Wynikami działania programu jest lista błędów i możliwych niespójności, które zostały wykryte w danych. SVP jest uruchamiany poprzez wybór drugiej opcji: „Weryfikacja istniejącej bazy danych scenariusza" (ang. *Verify an Existing Scenario Database*) z głównego menu JTLS.

1.4.1.3 Program modyfikacji terenu (TMU – Terrain Modification Utility)

Ten program narzędziowy służy do modyfikacji plików opisujących teren w oparciu o siatkę heksagonalną. Program wyświetla wybraną przez użytkownika bazę danych terenu nałożoną na wybraną mapę wektorową. Użytkownik może zmienić skalę i powiększyć określony obszar na mapie oraz wyświetlić siatkę sześciokątów. Można wówczas przeglądać i modyfikować poszczególne atrybuty heksagonów (typ terenu, typ przeszkody, wzniesienie).

1.4.1.4 Program modyfikacji współczynników równań Lanchester'a *(LDT – Lanchester Development Tool)*

Ten program wspomaga opracowywanie współczynników równań Lanchester'a, które są używane do obliczania rezultatów walki lądowej w JTLS. Użytkownik wprowadza spodziewany wynik bitwy o określonym czasie trwania, a LDT generuje proponowana tabelę współczynników Lanchester'a. LDT posiada także funkcję weryfikacji, która pozwala użytkownikowi zobaczyć w postaci tabeli rezultaty długoterminowej bitwy pomiędzy dwoma lub więcej jednostkami.



Rysunek 3 Okno interfejsu graficznego LDT

1.4.1.5 Program obsługi graficznej bazy danych *(GDP – Graphical Database Program)*

Ten program pozwala na graficzne rozmieszczenie jednostek i obiektów oraz na tworzenie lub modyfikację granic państw i obszarów operacji. Program wczytuje bieżące informacje scenariusza z bazy danych DDS i wyświetla je w GIAC (ang. *Graphical Input Aggregate Control*). GIAC działa w tym trybie inaczej niż zwykle i zmiany są zapisywane bezpośrednio w bazie DDS.

1.4.1.6 Program konwersji położenia (XCONVERT – Convert Location Program)

Program konwersji położenia jest narzędziem przyjmującym koordynaty położenia w następujących formatach:

- szerokość i długość geograficzna wyrażona w postaci tekstowej i dziesiętnej;
- współrzędne heksagonu;
- Siatka Merkatora (ang. *UTM – Universal Transverse Mercator*);
- wojskowa siatka współrzędnych (ang. *military grid*).

Kiedy zostanie wprowadzona poprawna lokalizacja program przekształca ją w pozostałe trzy systemy lokalizacji. Wyświetlane są koordynaty we wszystkich czterech formatach.

1.4.1.7 Program interfejsu HLA (HIP – High Level Architecture Interface Program)

Kiedy JTLS pracuje w trybie HLA (ang. *High Level Architecture)*, do rozprowadzania danych symulacji poprzez RTI (ang. *Run Time Infrastructure)* do innych federatów HLA w federacji, używany jest HIP (ang. *HLA Interface Program*) . Jeżeli tryb HLA jest uruchomiony z wykorzystaniem zarządzania czasem, używany jest dodatkowy program do regulacji szybkości symulacji, nazywany krokomierzem (ang. *Pacer*).

1.4.1.8 Program weryfikacji rozkazu (OVT – Order Verification Tool)

OVT jest tak zaprojektowany, żeby przyjmował informacje dystrybuowane przez GENIS (ang. *Generic Information Server*) z programu CEP (ang. *Combat Events Program*) i wykorzystywał je do weryfikacji formatu i zawartości rozkazów graczy i kontrolera, które zostały utworzone przy pomocy innych programów niż GIAC przed wysłaniem ich do modelu. Wszystkie sprawdzenia rozkazu, które byłyby realizowane przez GIAC, są także realizowane przez OVT.

OVT dostarcza interfejs graficzny, który pozwala operatorowi zweryfikować istniejące pliki rozkazów GIAC oraz pozwala innym programom połączyć się i zweryfikować rozkazy poprzez wykorzystanie odpowiedniego protokołu. Funkcjonalność sprawdzania rozkazów dostarczana jest przez bibliotekę funkcjonalną OVT (ang. *OVT-FL – OVT Functional Library*), która składa się z minimalnego podzbioru kodu niezbędnego do przeprowadzenia weryfikacji rozkazu. Ten zbiór kodu biblioteki obiektów jest również dostępny dla innych programów, które chcą przeprowadzić własną weryfikację rozkazu zamiast wykorzystywania OVT.

1.4.2 Programy inicjujące i konfiguracyjne

Programy te są używane do przygotowania uruchomienia JTLS z określonym scenariuszem.

1.4.2.1 Pogram inicjujący scenariusz (SIP – Scenario Initialization Program)

SIP jest jednym z programów, który musi być uruchomiony w celu przygotowania scenariusza, który nie był poprzednio uruchamiany w grze. Jedna z jego głównych funkcji, weryfikacja poprawności scenariusza przez moduł SVP, została omówiona wcześniej. Pozostałe funkcje programu SIP:

1. Ustalenie, czy istnieją wszystkie pliki danych definiujących scenariusz;
2. Utworzenie poprawnej struktury katalogów do przechowywania plików danych podczas działania gry;
3. Kopiowanie wszystkich plików danych inicjujących scenariusz do katalogu gry;
4. Utworzenie plików danych niezbędnych dla innych procesów JTLS – np. plików definicji rozkazów (z wykorzystaniem OPP) i definicji menu dla GIAC;
5. Weryfikacja wyboru pliku mapy obszaru dla gry.

SIP musi zostać powtórnie uruchomiony dla scenariusza, kiedy:

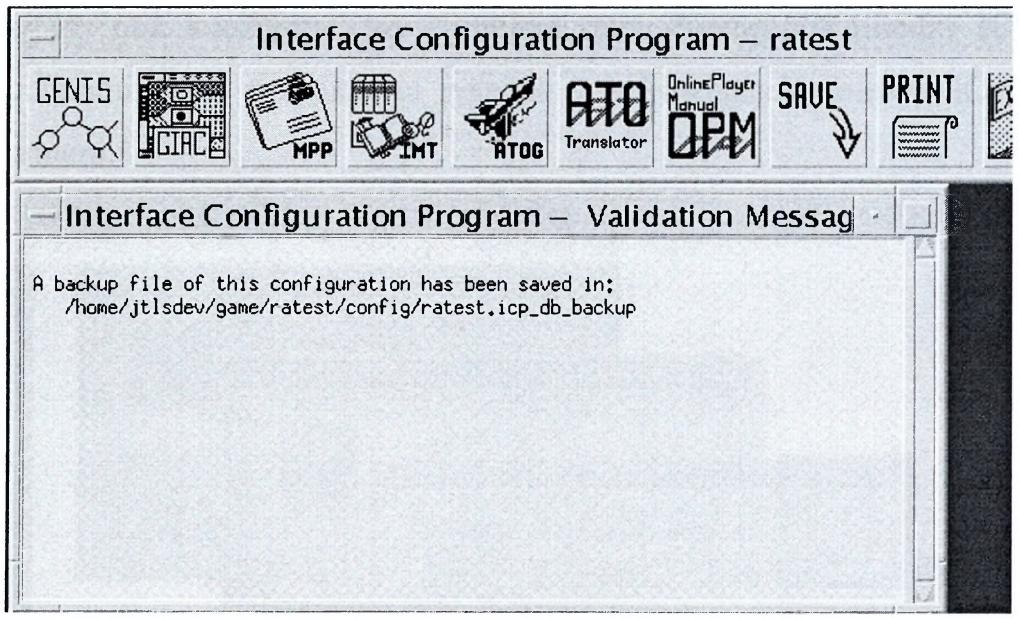
1. zmieniły się dane inicjujące;
2. zmieniły się definicje w słowniku statycznym;
3. zainstalowano nową wersję GIAC;
4. zmienił się plik definicji rozkazu.

1.4.2.2 Program konfiguracji interfejsu (ICP – Interface Configuration Program)

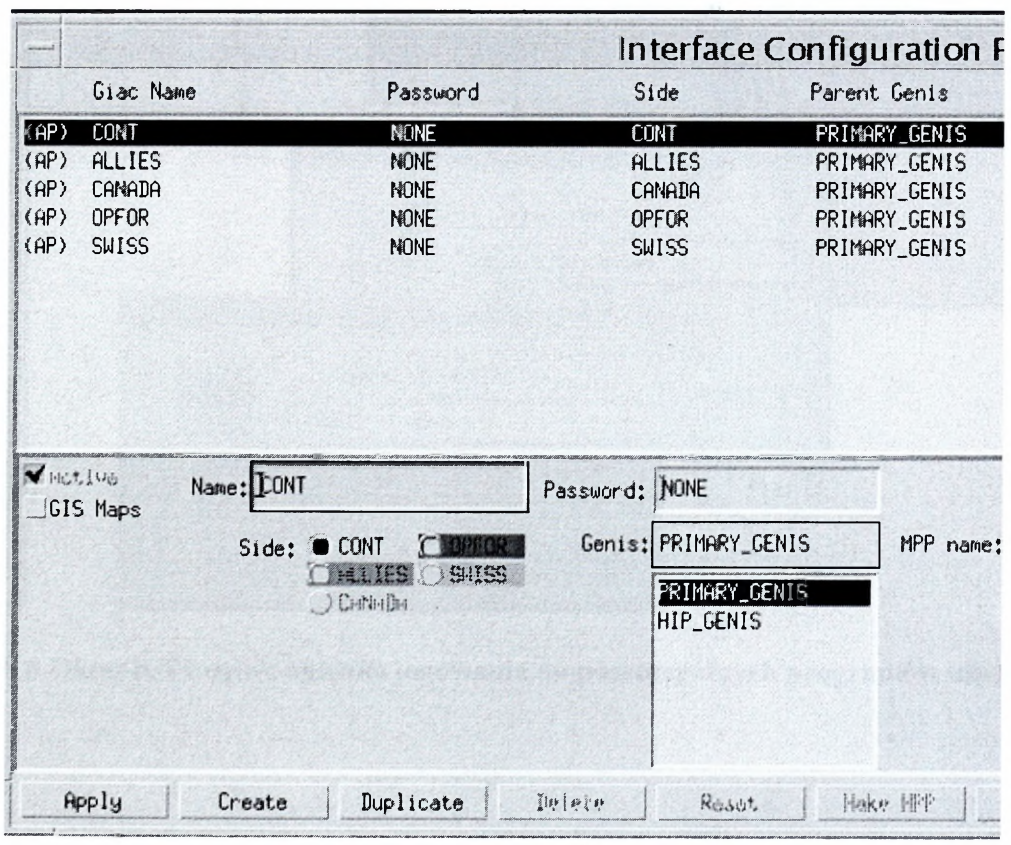
ICP jest interaktywnym programem, który umożliwia użytkownikowi zdefiniowanie wyspecyfikowanie parametrów konfiguracyjnych dla każdego procesu, jaki może być uruchomiony dla danego scenariusza: OPMs (ang. *Online Player Manuals*), ATOG/Ts (ang. *Air Tasking Order Generator and Translator Programs*), IMTs (ang. *Information Management Tools*), MPPs (ang. *Message Processor Programs*), GIACs, GENISes i CEP. Program wykorzystuje interfejs graficzny by pozwolić użytkownikowi edytować domyślne (określone, gdy uruchomiony został SIP) konfiguracje procesów. Dane wymagane do zdefiniowania procesów w konfiguracji gry są różne dla różnych typów procesów. Definicja procesu zawiera nazwę, hasło, proces nadrzędny, stronę, do której jest przypisany, hosta, na którym proces ma się wykonywać, itd.

ICP zwykle jest uruchamiany przed startem symulacji, aby dostosować konfigurację do potrzeb ćwiczenia, jakkolwiek może być również uruchomiany w trakcie działania JTLS. Na

przykład ICP może być uruchomiony w celu wyspecyfikowania parametrów dla nowej wersji GIAC (oraz związanych z nim MPP i IMT), która została dodana konfiguracji początkowej systemu i uruchomiona.



Rysunek 4 Okno główne ICP

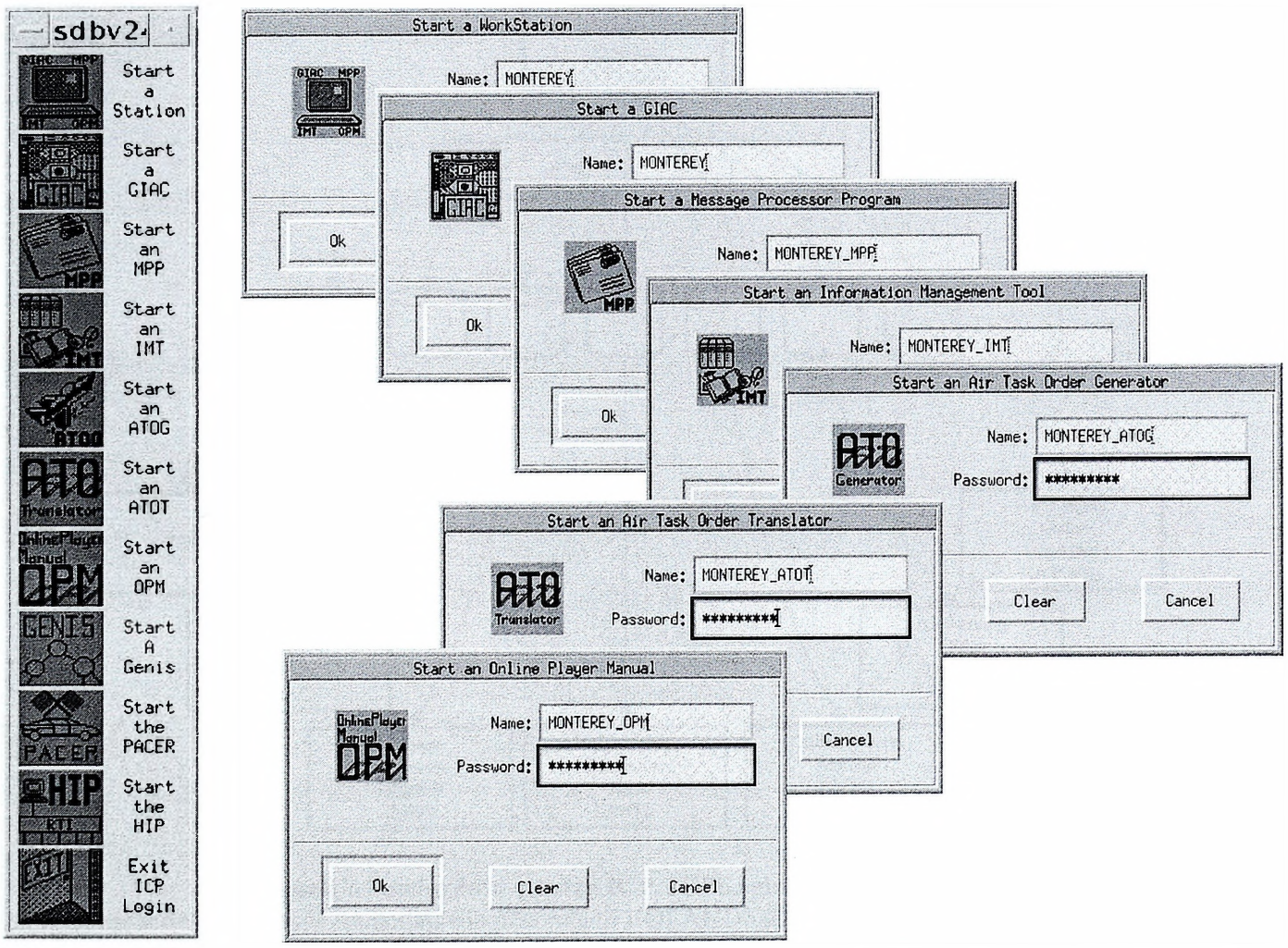


Rysunek 5 Okno ICP konfiguracji GIAC

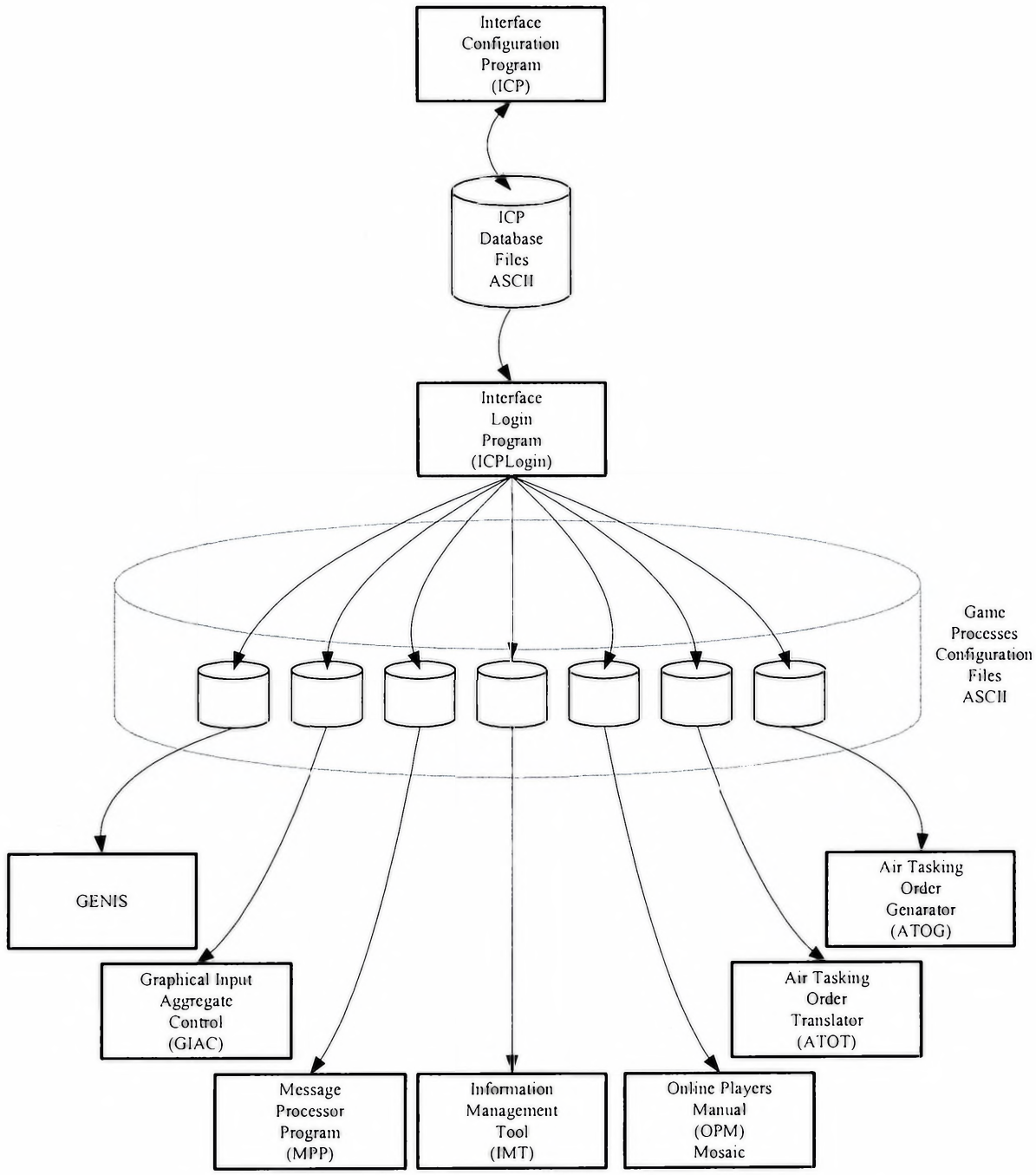
1.4.2.3 Program logowania (ICPLogin – Interface Login Program)

Program zaprojektowany w celu ułatwienia uruchamiania procesów symulacji (np. GIAC). Wszystkie procesy symulacji są zwykle uruchamiane przy użyciu tego programu.

Program ICPLogin jest uruchamiany przez wybór opcji w menu głównym JTLS. Dlatego też, aby uruchomić proces użytkownik po prostu klika na właściwym klawiszu. Po wyborze procesu program ICPLogin czyta plik danych, który został utworzony przez ICP, tworzy wymagany plik konfiguracyjny i inicjuje start. Powiązania między ICP, ICPLogin i innymi programami JTLS przedstawia rysunek 7 – Powiązania między ICP, ICPLogin i innymi programami.



Rysunek 6 Okno ICPLogin i okienka logowania do poszczególnych programów użytkowych

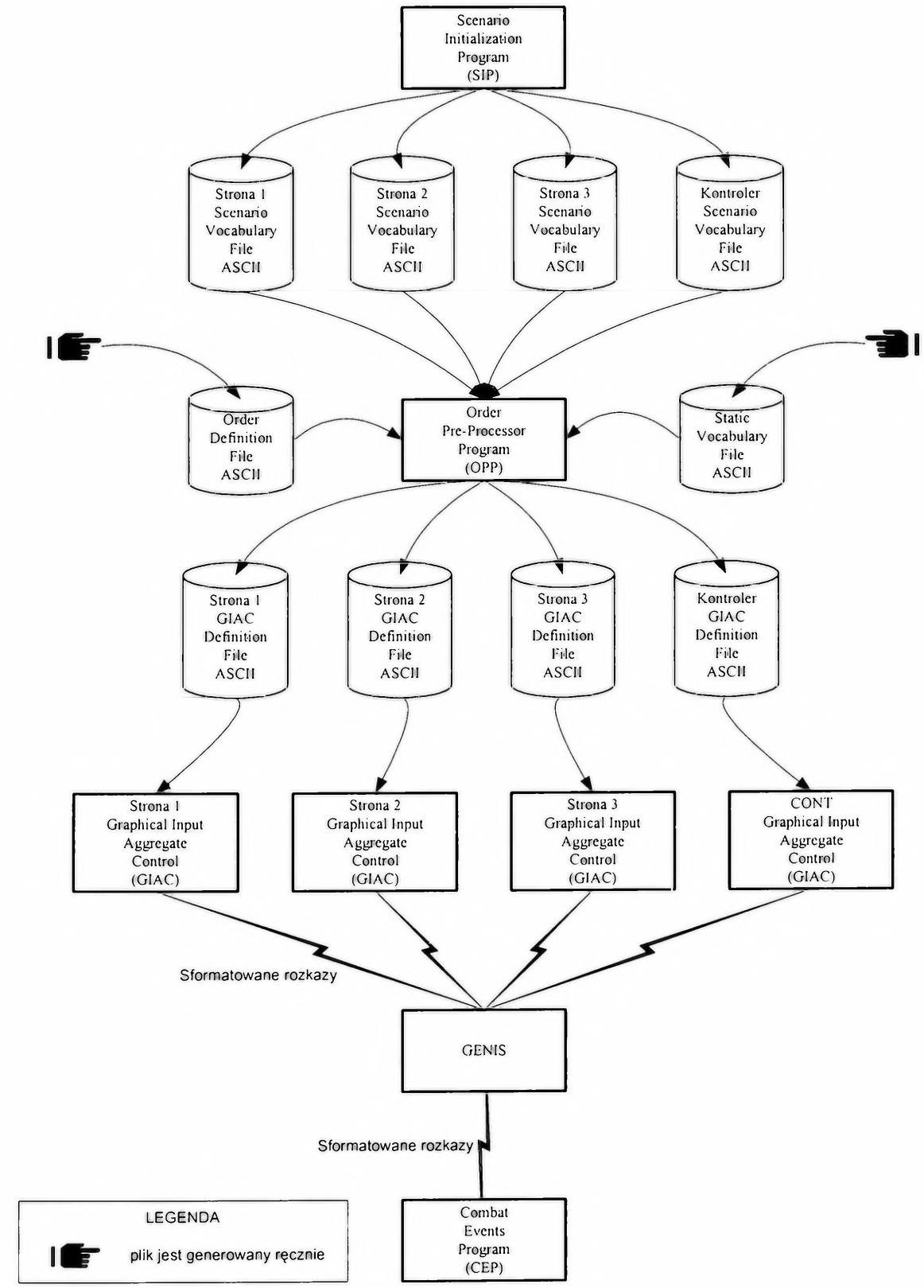


Rysunek 7 Powiązania między ICP, ICPLogin i innymi programami
/źródło: opracowanie własne na podstawie prezentacji ROLANDS&ASSOCIATES Corporation – styczeń 2004/

1.4.2.4 Program wstępnego przetwarzania rozkazów *(OPP – Order Pre-Processor)*

Program wstępnego przetwarzania rozkazów (ang.*OPP – Order Pre-Processor*) jest odpowiedzialny za odczyt pliku definicji rozkazów (ang. *order definition file*), pliku stałego słownictwa (ang. *static vocabulary file*) oraz pliku słownictwa scenariusza (ang. *scenario vocabulary files*) w celu wygenerowania plików wejściowych i kodów dla innych programów. W tym plikami definicji *.gdf dla programu sterującego graficznymi danymi wejściowymi GIAC (ang. *Graphical Input Aggregate Control*) wykorzystywanymi przez niego do definiowania panelu rozkazów JTLS, które są wypełniane przez użytkowników JTLS. Pliki te

zawierają również formaty tabel Oracle i związane z nimi kody używane do czytania tabel przez program kliencki wprowadzania rozkazów (ang. *OEC – Order Entry Client*). OPP jest uruchamiany jako część procesu inicjalizacji scenariusza w celu wygenerowania plików *.gdf dla GIAC i może być uruchomione z linii komend dla wygenerowania tabel i kodów dla OEC.



Rysunek 8 Powiązania między OPP i innymi programami
/źródło: opracowanie własne na podstawie JTLS Software Maintenance Manual v.2.6/

1.4.3 Program zdarzeń walki *(CEP – Combat Events Program)*

Program zdarzeń walki *(CEP – Combat Events Program)* jest głównym programem systemu JTLS – jest implementacją modelu walki. Określa wszystkie akcje i interakcje pomiędzy siłami lotniczymi, lądowymi i morskimi zdefiniowanymi i modelowanymi dla określonego, uruchamianego scenariusza. CEP tworzy, obsługuje i raportuje bieżący status modelowanego środowiska działań wojennych.

CEP może modelować do 10 stron lub koalicji w dowolnym scenariuszu. Każda strona może określić relacje (przyjacielski, wrogi, podejrzany, neutralny) z każdą inną przedstawioną stroną. Tylko jeden program może działać na każdym pojedynczym komputerze lub w sieci w czasie działania JTLS.

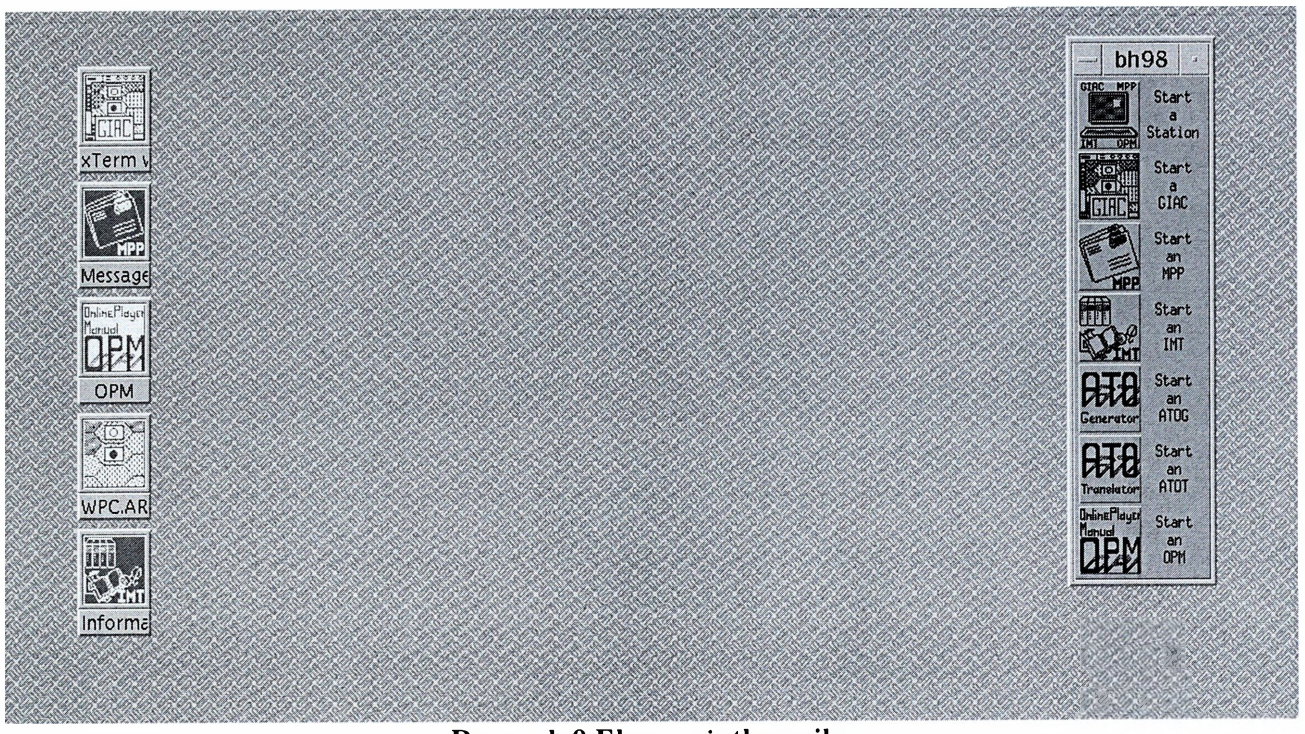
CEP komunikuje się poprzez protokół TCP/IP z wykorzystaniem mechanizmu gniazdek (ang. *socket*) jedynie z pojedynczym procesem Primary GENIS. Primary GENIS otrzymuje dane inicjujące i okresowe uaktualnienia z CEP. Ten z kolei, komunikuje się z GIACs, MPPs, IMTs i Secondary GENISes, które są do niego przydzielone. Każdy GENIS obsługuje swoja własna bazę danych symulacji i komunikuje się z procesami bezpośrednio podległymi w strukturze drzewa dystrybucji informacji.

Dane wejściowe do symulacji dostarczane przez gracza są wprowadzane w formie rozkazów wpisywanych w GIAC i przesyłane do CEP w celu przetworzenia przez jeden lub więcej procesów GENIS w drzewie dystrybucji informacji. Gracze otrzymują informacje o grze z CEP w formie: uaktualnień zobrazowania graficznego w GIAC, wiadomości MPP i uaktualnień danych tabelarycznych wyświetlanych w IMT.

1.4.4 Podstawowe programy użytkowe

Aplikacje GIAC, IMT, OPM, MPP używane są przez wszystkich użytkowników biorących udział w ćwiczeniu, natomiast aplikacje ATOG i ATOT tylko przez użytkowników związanych z kierowaniem siłami lotniczymi.

Po włączeniu stacji roboczej, na której zainstalowany został system JTLS, w odpowiednim oknie użytkownik uzyskuje dostęp do wszystkich aplikacji użytkowych (patrz: Rysunek 9 Ekran użytkownika).



Rysunek 9 Ekran użytkownika

Po kliknięciu na odpowiedniej ikonie uruchamiane są przypisane aplikacje.

1.4.4.1 Program sterujący graficznymi danymi wejściowymi *(GIAC – Graphical Input Aggregate Control)*

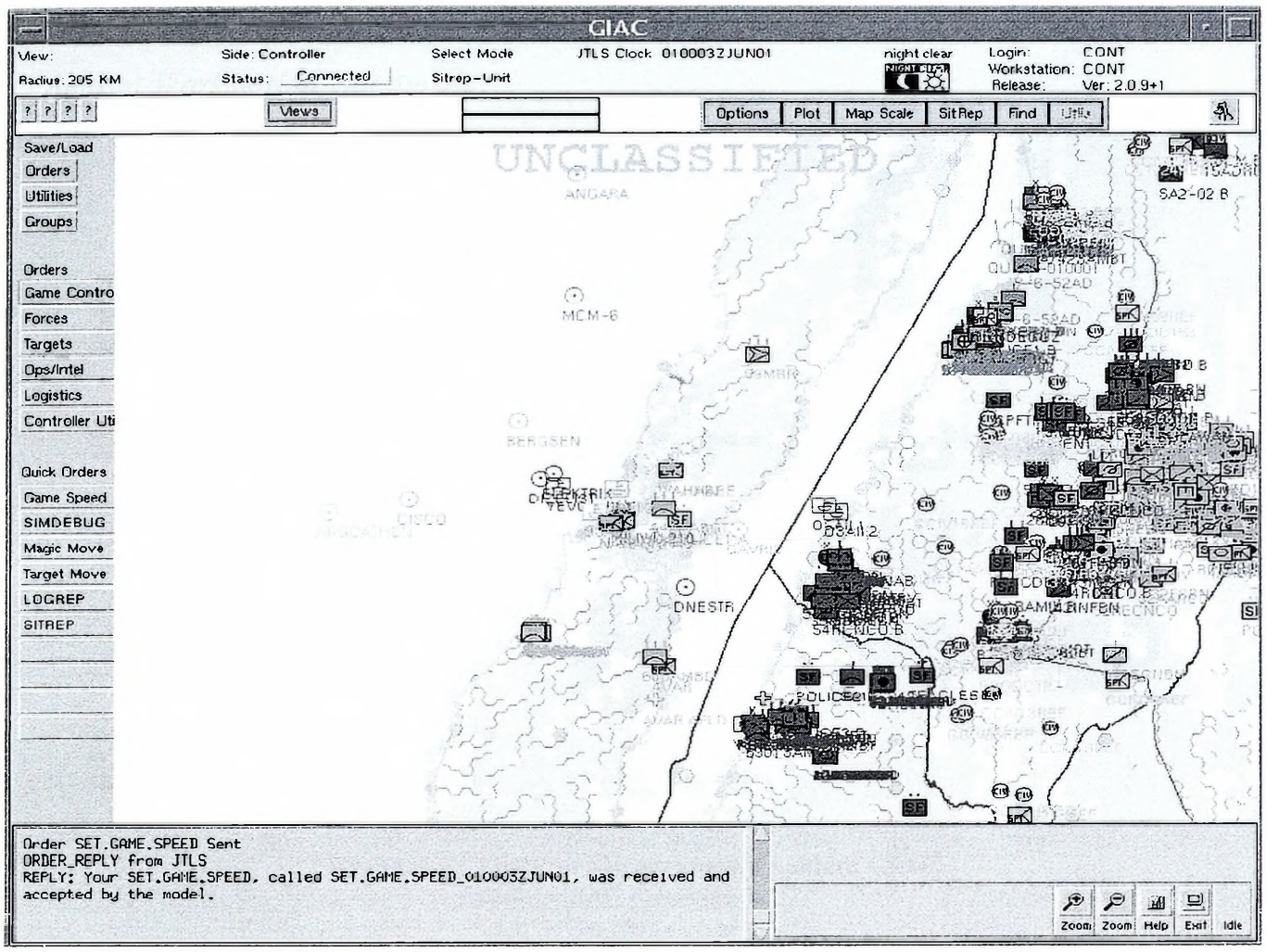
Z punktu widzenia użytkownika, GIAC jest interfejsem pomiędzy ćwiczącym, a CEP (ang. *Combat Events Program*) pozwalającym na dwustronną wymianę danych.

GIAC dostarcza szeregu szablonów rozkazów. Użytkownik (gracz, operator stacji) wprowadza dane, żeby wypełnić każde wymagane pole, natomiast GIAC sprawdza, czy można je zaakceptować, zanim prześle rozkaz do CEP.

Ten sam GIAC jest wykorzystywany do zobrazowania graficznego informacji przestrzennej o stanie i położeniu wojsk własnych oraz przeciwnika, na podstawie dostępnych dla gracza informacji o stronie przeciwnej.

Skrótowo podstawowa funkcjonalność GIAC wyraża się następująco:

- graficzne zobrazowanie terenu (mapy);
- dostarczanie niezbędnych informacji o terenie działań (współrzędne, odległości);
- wyświetlanie zagregowanych informacji o stanie i położeniu wojsk;
- wyświetlanie pory doby (noc, dzień);
- wyświetlanie informacji o warunkach pogodowych;
- wydawanie rozkazów dla podległych wojsk.



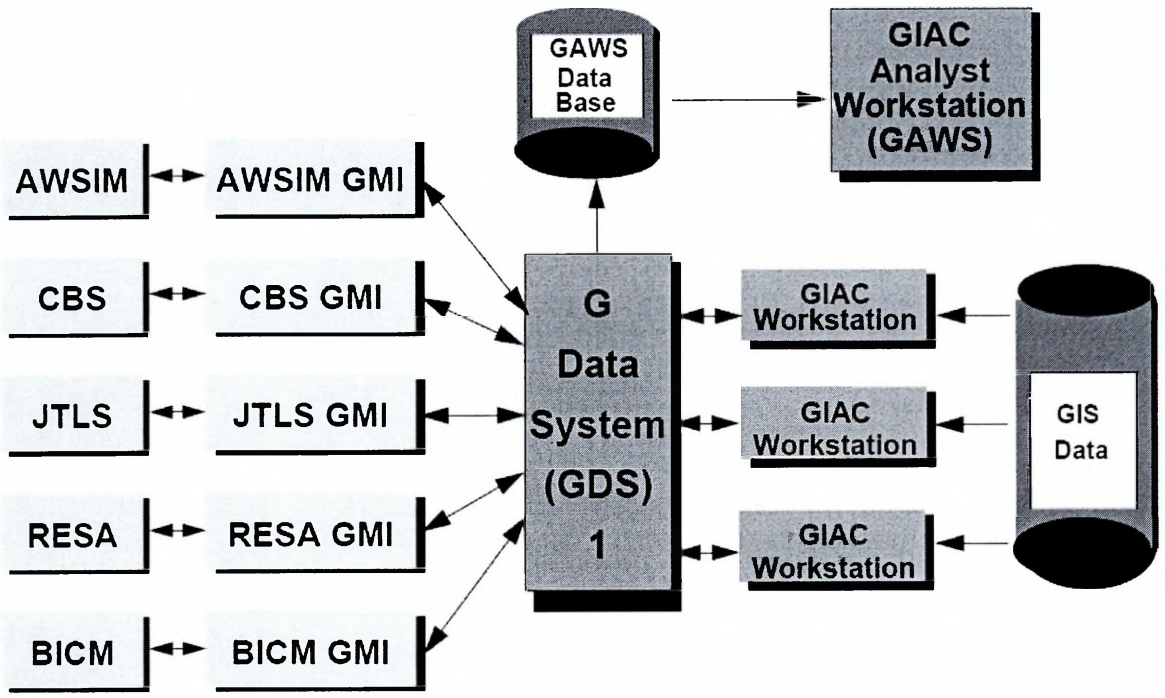
Rysunek 10. Okno aplikacji GIAC – zobrazowanie sytuacji

Z technologicznego punktu widzenia GIAC jest systemem zobrazowania graficznego opartym o X-Windows. Widoczny dla użytkownika element pakietu GIAC pobiera informacje do zobrazowania z własnej sieciowej bazy danych zewnętrznej w stosunku do JTLS. CEP aktualizuje informacje wysyłane do GIAC z wykorzystaniem protokołu opartego na sformatowanych komunikatach. Te komunikaty są wysyłane do GENIS, który przesyła je do GDS (ang. *G Data System*), który udostępnia usługi dostępu do sieciowej bazy danych GIAC. Warto nadmienić, że GIAC może współpracować z różnymi modelami symulacyjnymi.

Na całość pakietu GIAC składają się następujące elementy:

- interfejs modelu GIAC (ang. *GMI – GIAC Model Interface*);
- podsystem bazy danych (ang. *GDS – G Data System*);
- stacja robocza dowodzenia (ang. *GIAC Command and Control Workstation*);
- system informacji geograficznej (ang. *GIAC GIS – Geographical Information System*);
- graficzna stacja robocza analityka (ang. *GAWS – Graphical Analyst Workstation*);

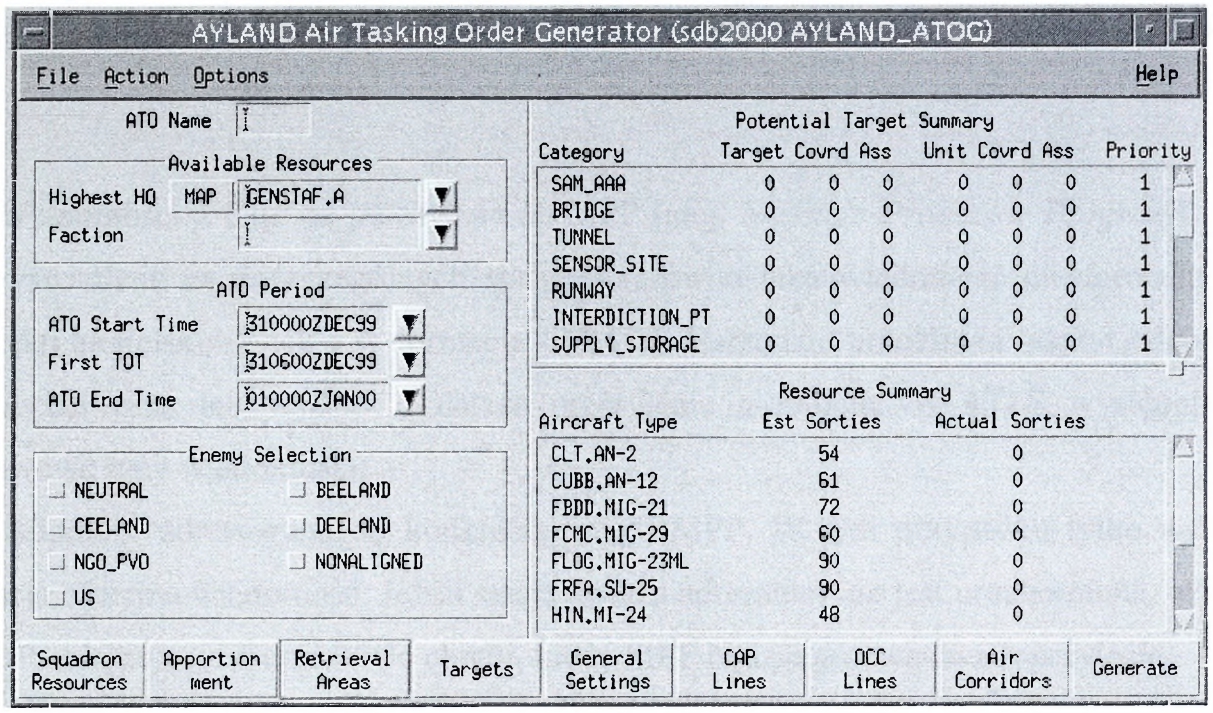
– baza danych graficznej stacji roboczej analityka (ang. *GAWS Database – Graphical Analyst Workstation Database*).



Rysunek 11 Struktura programowa pakietu GIAC
/źródło: GIAC Overview - 1.7/

1.4.4.2 Generator rozkazów zadań lotniczych (*ATOG – Air Tasking Order Generator*)

Generator rozkazów zadań lotniczych (ang. *ATOG – Air Tasking Order Generator*) został zaprojektowany w celu dostarczenia wsparcia graczowi sił powietrznych w tworzeniu rozkazów zadań lotniczych – zarówno ofensywnych, jak i defensywnych. Gracz specyfikuje obszary celów, priorytety celów, dostępne lotnicze środki bojowe oraz zamierzony przez dowództwo podział zadań dla wybranego okresu ATO. ATOG, wykorzystując te wskazówki, posiadane przez gracza informacje o bieżącej sytuacji bojowej oraz bieżący stan ukompletowania i zaopatrzenia jednostki, tworzy zestaw skoordynowanych rozkazów dla lotnictwa. Gracz ma możliwość przeglądania i zmian poszczególnych parametrów rozkazów przed wysłaniem wygenerowanych rozkazów do CEP.



Rysunek 12 Okno główne ATOG

1.4.4.3 Translator rozkazów zadań lotniczych (*ATOT – Air Tasking Order Translator*)

Translator rozkazów zadań lotniczych (ang. *ATOT – Air Tasking Order Translator*) dokonuje translacji rozkazów zadań lotniczych w formacie USMTF (ang. *USMTF ATO – US Message Text Format Air Tasking Order*), które są tworzone przez program generatora rozkazów zadań lotniczych systemu CTAPS (ang. *Contingency Tactical Air Planning System*) wykorzystywanego przez USAF, na format rozkazów zadań lotniczych JTLS. Konieczne jest uzyskanie, dla potrzeb symulacji, rozkazów zadań lotniczych w wyspecyfikowanym formacie.

Personel techniczny będzie umieszczał plik w odpowiednim katalogu. Następnie program translatora umożliwi wgląd w wyniki translacji i umożliwia w ograniczonym zakresie na modyfikację niektórych zadań.

To narzędzie pozwala na masowe przyjmowanie rozkazów wydanych przez działające centrum operacji powietrznych (ang. *AOC – Air Operations Center*). Jednak niektóre rozkazy zadań lotniczych nie są translowane automatycznie. Dlatego te rozkazy gracz lotniczy będzie musiał wprowadzić na swojej stacji roboczej GIAC.

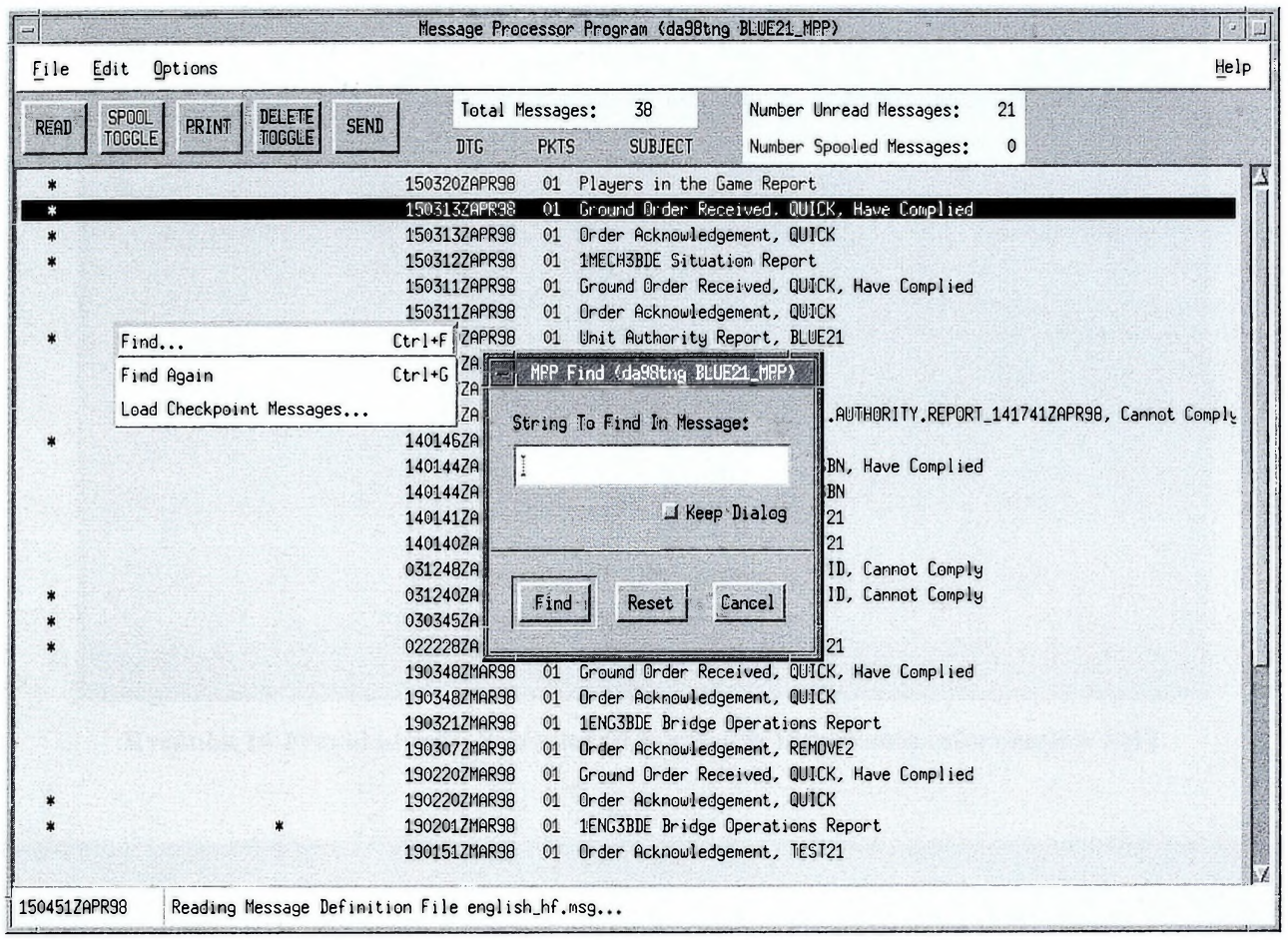
1.4.4.4 Program przetwarzania wiadomości (*MPP – Message Processor Program*)

Wiadomości z CEP są przesyłane do MPP (ang. *Message Processor Program*) i mogą być wyświetlane na poszczególnych stacjach zarówno jako wiadomość niesformalizowana (w języku angielskim), jak i w formacie USMTF. Narzędzie umożliwia odczyt odebranych wiadomości oraz ich wydruk i dalsze przesyłanie e-mailem. W JTLS występują dwa podstawowe typy wiadomości:

1. Wiadomość adresowana do konkretnej stacji MPP. W tym przypadku tylko wskazana stacja otrzyma wiadomość. Jeżeli stacja będąca adresatem nie jest uruchomiona, wówczas CEP kolejkuje wiadomości do chwili, kiedy MPP będzie gotowe do ich przyjęcia.
2. Wiadomość adresowana ogólnie do strony i funkcji graczy. Taka wiadomość jest określana jako wiadomość rozgłoszeniowa (ang. *broadcast*) i jest przesyłana do każdego MPP należącego do wskazanej strony. Gracz posiada możliwość kontrolowania, jakiego rodzaju wiadomości powinny być wyświetlane na jego stacji MPP, a jakie nie.

Zazwyczaj JTLS przesyła informacje o charakteryzujące bieżący stan gry do GIAC i IMT. Oprócz tego, w kilku przypadkach generowane są wiadomości MPP. Ma to miejsce, gdy:

1. gracz przesłał żądanie szczegółowej informacji o wojskach (np. SITREP lub LOGREP);
2. wystąpiła potrzeba poinformowania gracza przez CEP o powstałych problemach (np. nieprawidłowościach dotyczących wydanych rozkazów – niewłaściwie współrzędne położenia, nazwy jednostek);
3. jedna z jednostek podległych stwierdziła, że powinna poinformować przełożonego (dowódcę) o zaistniałej sytuacji – np. że znalazła się pod ostrzałem artyleryjskim;
4. generowany jest zbiorczy okresowy raport o podjętych działaniach.

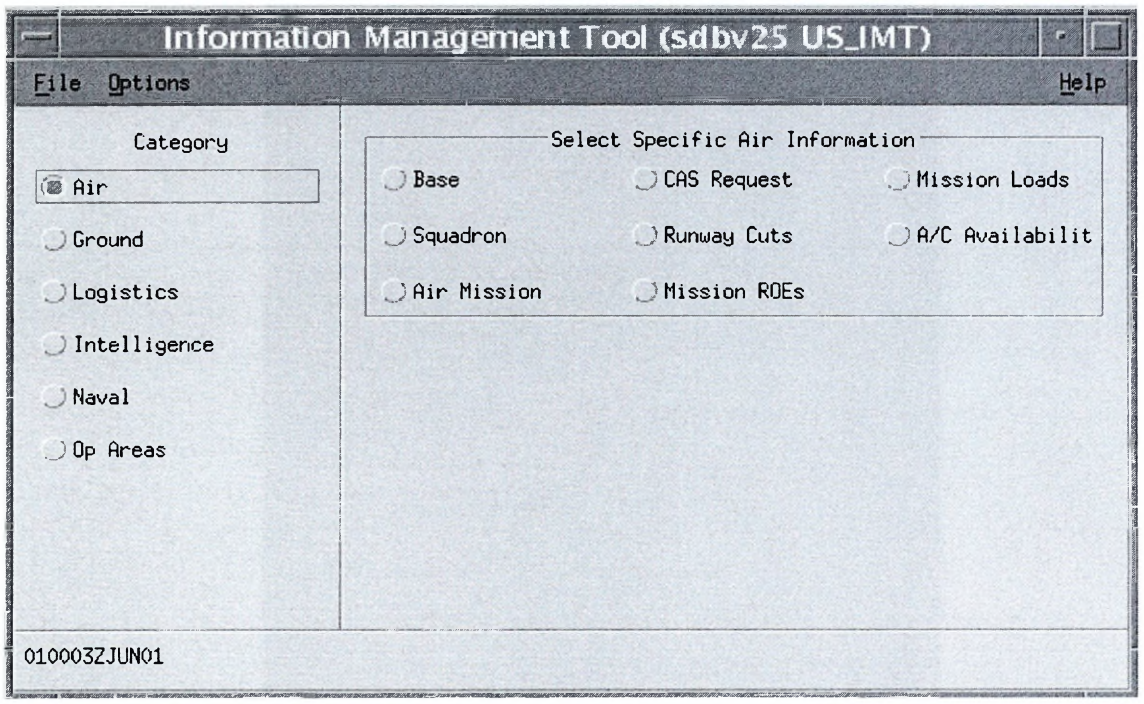


Rysunek 13. Przykładowe okno aplikacji MPP

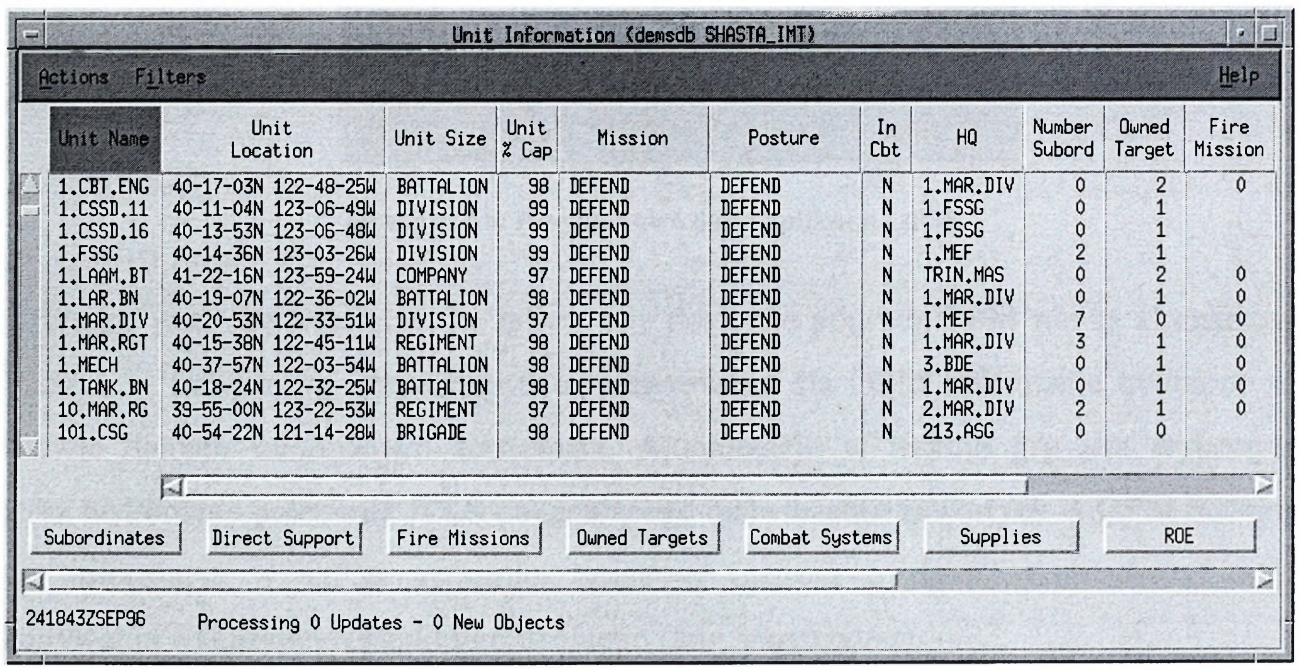
1.4.4.5 Narzędzie zarządzania informacją (IMT – Information Management Tool)

Aplikacja IMT pozwala na wyświetlenie w postaci tabelarycznej szczegółowych aktualnych informacji na temat wybranych oddziałów, grup oddziałów lub innych elementów (np. logistycznych) wojsk własnych oraz informacji zgromadzonych o wojskach przeciwnika. W aplikacji można wyświetlić wyselekcjonowane informacje poprzez wykorzystanie mechanizmu ich filtrowania.

Program posiada około 40 różnych tabel i menu.



Rysunek 14 Przykładowe okno selekcji kryteriów filtrowania informacji w IMT

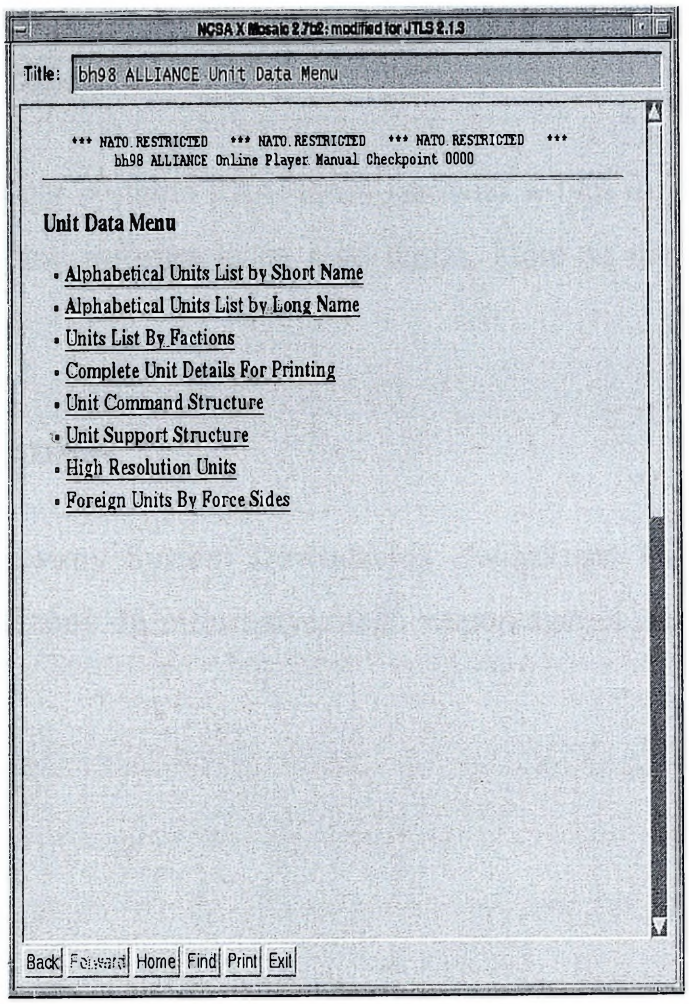


Rysunek 15. Przykładowa tabela w aplikacji IMT

1.4.4.6 Podręcznik gracza (OPM – Online Player's Manual)

Aplikacja OPM (ang. *Online Player's Manual*) spełnia następujące funkcje:

- umożliwia dostęp do plików tekstowych zawierających sformatowane dane opisujące scenariusz;
- pozwala przeglądać informacje o zdolnościach bojowych i potencjale jednostek biorących udział w grze;
- informacje są wyświetlane poprzez przeglądarkę HTML (np. Mozilla).



Rysunek 16. Przykładowe okno aplikacji OPM

Ważne jest, aby pamiętać, że informacje dostępne poprzez OPM nie są aktualizowane na bieżąco w trakcie gry. Początkowo wygenerowane dla OPM informacje tworzone są na podstawie danych inicjujących scenariusz. Aktualizacja w trakcie gry jest wykonywana poprzez wykonanie polecenia JTLS nakazującego odświeżenie zawartych w OPM informacji. Nowe informacje w OPM są umieszczane w oparciu o dane zgromadzone podczas wykonywania wskazanego punktu kontrolnego (ang. *checkpoint*).

2. System SZAFRAN

Ze względu na charakter systemu SZAFRAN zarówno w tym rozdziale jak i następnych zostaną zawarte tylko ogólne informacje na jego temat, które są niezbędne do omówienia analizowanych zagadnień.

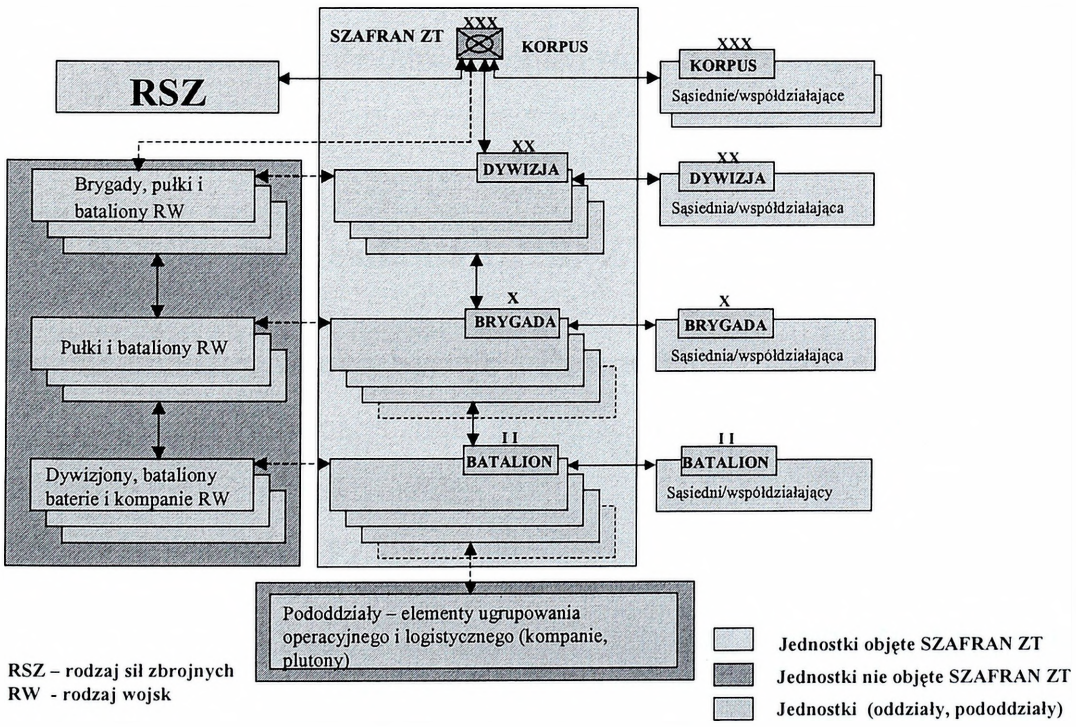
2.1 Przeznaczenie

Polowy Zautomatyzowany System Dowodzenia Związkiem Taktycznym (PZSD ZT) „SZAFRAN" jest przeznaczony do informatycznego wspomagania procesów dowodzenia na szczeblach:

- korpusu;
- dywizji;
- brygady;
- batalionu

wojsk lądowych.

PZSD ZT ma zapewnić integrację zautomatyzowanych systemów dowodzenia i kierowania środkami walki rodzajów wojsk i służb, współpracę z zautomatyzowanymi systemami dowodzenia rodzajów sił zbrojnych oraz interoperacyjność z podobnymi systemami innych krajów NATO na z wyżej wymienionych szczebli dowodzenia.



Rysunek 17 Zasięg i otoczenie PZSD ZT

/źródło: Piela J.: Polowy Zautomatyzowany System Dowodzenia Związkiem Taktycznym. Stan obecny i perspektywy rozwoju/

Potrzeba spełnienia wymagań interoperacyjności wiąże się z koniecznością stosowania standardów i wymagań NATO, w szczególności dotyczącymi wymiany informacji sformalizowanej tzn. AdatP-3 i model danych ATCCIS (ang. *Army Tactical Command and Control Information System*), oraz zalecanych przez Agencję NC3A (ang. *NATO C3 Agency*) standardów międzynarodowych, między innymi w zakresie stosowanych znaków taktycznych APP6A oraz systemów otwartych NATO OSE (ang. *NATO Open Systems Environment*).

Powyższe założenia wymuszają wymagania stawiane systemowi dowodzenia dotyczące między innymi:

- czasów rozwijania i zwijania stanowisk dowodzenia,
- współdziałania z innymi systemami (w tym, wymiana informacji z systemami rodzajów wojsk oraz z systemami innych krajów NATO),
- czasu i jakości planowania działań bojowych,
- podatności na zmiany sytuacji bojowej i nieoczekiwane zdarzenia,
- żywotności stanowisk i systemu dowodzenia,
- działania w różnych warunkach klimatycznych
- podatności transportowej.

Wymagania te mogą być spełnione jedynie poprzez automatyzację poszczególnych procesów cyklu dowodzenia oraz stosowanie odpowiednich rozwiązań technicznych zabezpieczających ich realizację.

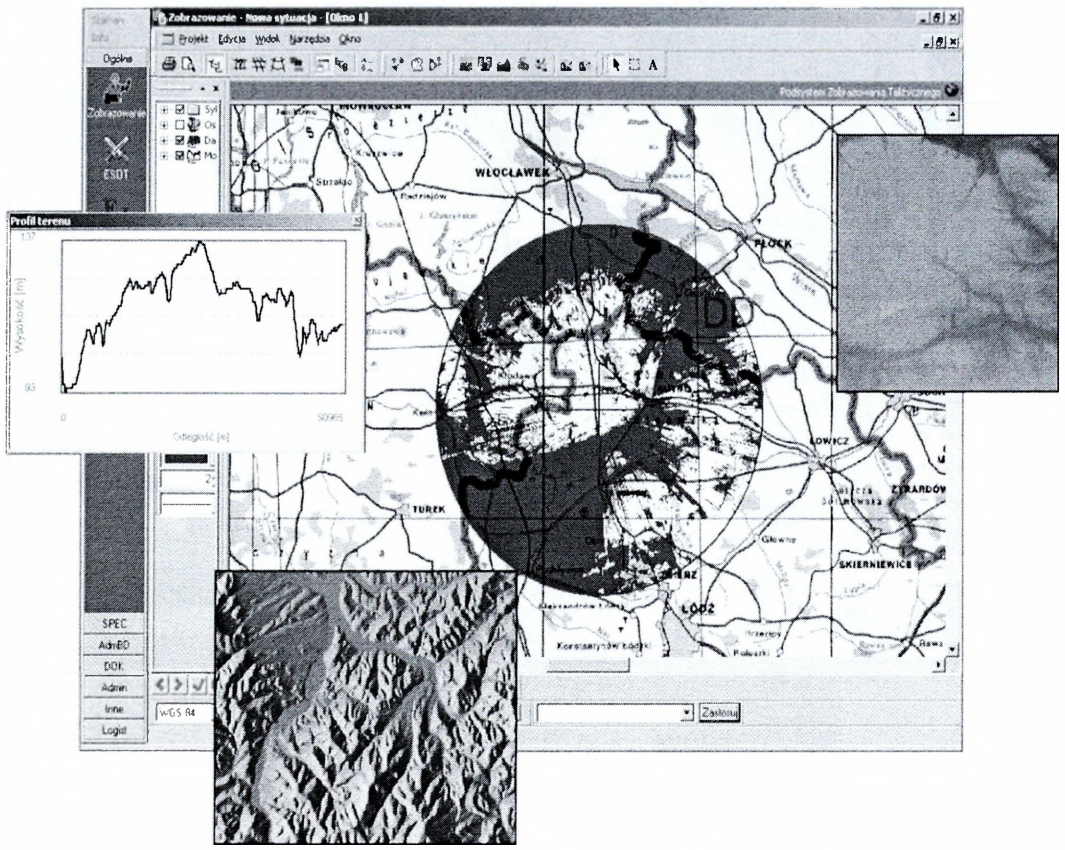
2.2 Podstawowe zadania PZSD ZT

Polowy Zautomatyzowany System Dowodzenia Związkiem Taktycznym jest rozproszonym systemem informatycznym przeznaczonym do wspomagania wszystkich faz cyklu dowodzenia na szczeblu taktycznym.

Podstawowymi zadaniami PZSD ZT są:

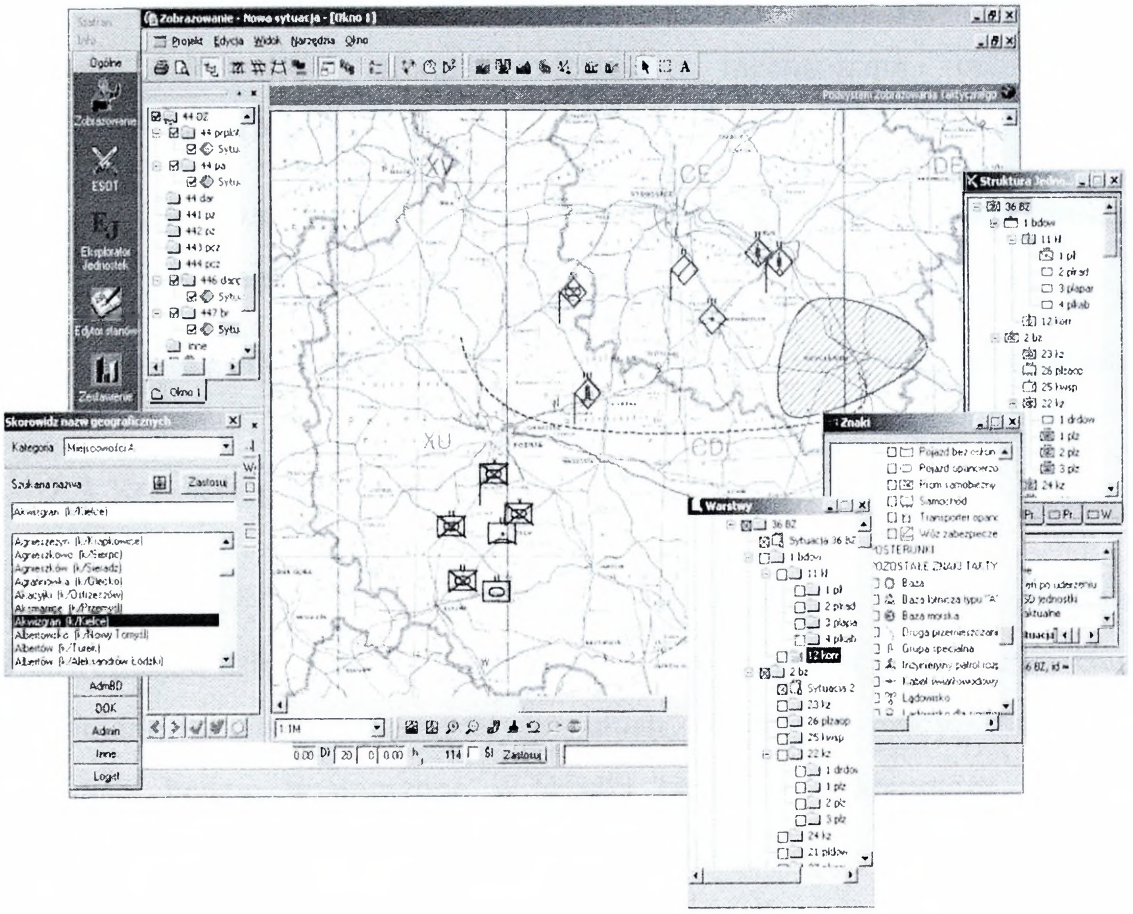
- utrzymywanie baz danych zawierających aktualne i jednolite dane (model danych zgodny ze standardem LC2IEDM v 2) o:
  - sytuacji taktycznej;
  - wojskach własnych i sąsiadach;

- przeciwniku;
- warunkach prowadzenia działań (informacje o terenie, pogodzie, itp.);



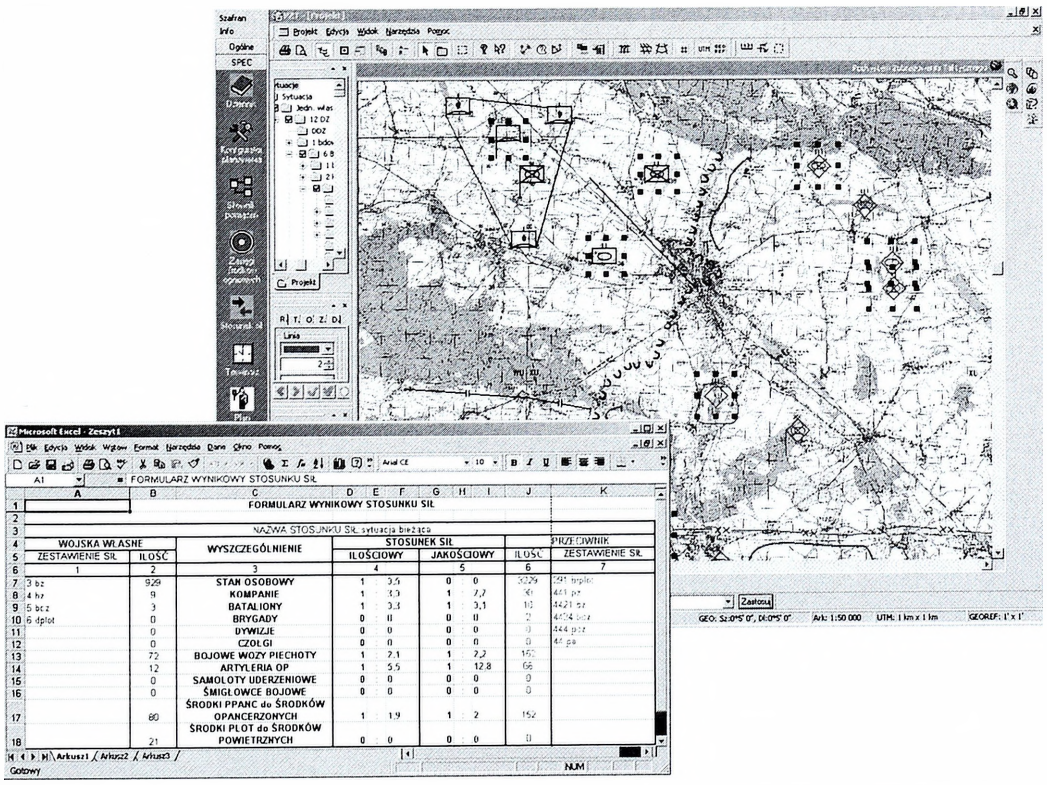
Rysunek 18 Możliwości analizy o terenu w systemie SZAFRAN

- sporządzanie dokumentów dowodzenia (sformalizowanych meldunków, rozkazów, zarządzeń, sprawozdań, planów) z uwzględnieniem warunków pracy grupowej;
- wymiana dokumentów dowodzenia między stanowiskami dowodzenia i ich archiwizacja, w tym wymiana dokumentów sformalizowanych, zapewniających minimalizację ilości przesyłanych informacji i współdziałanie z innymi systemami, w tym z systemami innych krajów NATO (AdatP-3);
- zobrazowanie, na bazie jednolitej informacji utrzymywanej w bazie danych, sytuacji taktycznej na tle „mapy elektronicznej", dostosowanej do danego szczebla dowodzenia;



Rysunek 19 Zobrazowanie sytuacji w systemie SZAFRAN

- realizacja ustalonych zadań typu „kalkulacji sztabowych" i udostępnianie ich wyników osobom funkcyjnym;



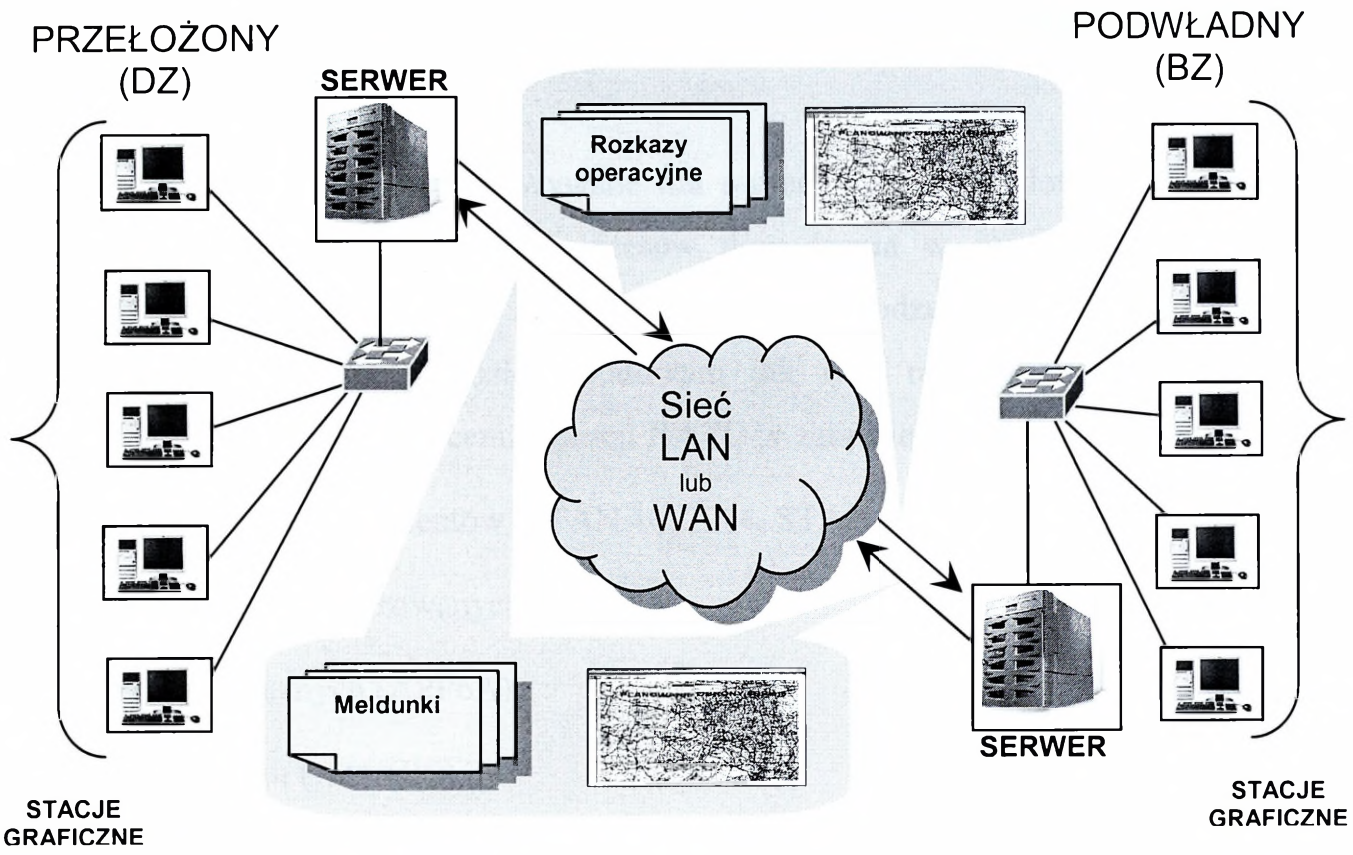
Rysunek 20 Wyznaczanie stosunków sił w systemie SZAFRAN

- synchronizacja zawartości baz danych między innymi poprzez wykorzystanie wymiany dokumentów sformalizowanych i mechanizmu replikacji zgodnego z wymaganiami NATO;
- integracja zautomatyzowanych systemów dowodzenia i kierowania środkami walki rodzajów wojsk i służb;
- współdziałanie z zautomatyzowanymi systemami dowodzenia rodzajów sił zbrojnych.

2.3 Struktura systemu

PZSD ZT dostarczy osobom funkcyjnym poszczególnych komórek organizacyjnych stanowisk dowodzenia usług wspomagających realizację procesów dowodzenia. Narzędzia te będą udostępniane w ramach zautomatyzowanych stanowisk pracy. Zautomatyzowane stanowisko pracy będzie wyposażone w środki łączności i komputer z odpowiednim oprogramowaniem użytkowym wspomagającym wykonywanie zadań określonej osoby funkcyjnej. Będzie ono zlokalizowane w zautomatyzowanych wozach dowódczo-sztabowych, zautomatyzowanych wozach sztabowych lub też innych pomieszczeniach wykorzystywanych do rozwijania stanowisk dowodzenia.

Rozmieszczenie zautomatyzowanych stanowisk pracy w zautomatyzowanych wozach dowódczo-sztabowych i zautomatyzowanych wozach sztabowych w istotny sposób wpłynie na mobilność systemu. Zautomatyzowane stanowiska pracy w zautomatyzowanych wozach dowódczo-sztabowych zapewnią możliwość pracy zarówno na postoju jak i ruchu, natomiast w zautomatyzowanych wozach sztabowych zapewnią pracę na postoju.



Rysunek 21 Architektura sprzętowa PZSD SZAFRAN-ZT

PZSD ZT składa się z oprogramowania i środków technicznych. Środkami technicznymi jest sprzęt łączności i informatyki dostosowany do pracy w warunkach polowych.

2.4 Oprogramowanie PZSD ZT

Oprogramowanie PZSD ZT można podzielić na trzy zasadnicze warstwy:

- oprogramowanie systemowe;
- oprogramowanie usługowe i narzędziowe;
- oprogramowanie użytkowe.

Oprogramowanie systemowe stanowią sieciowe systemy operacyjne (SOLARIS i Windows NT/Windows 2000) stosowane zarówno na serwerach jak i stacjach roboczych.

W skład oprogramowania usługowego wchodzą komercyjne pakiety oprogramowania zapewniające między innymi funkcjonowanie baz danych, poczty elektronicznej, pracy grupowej czy też zarządzanie siecią komputerową. Oprogramowanie narzędziowe stanowią

biblioteki procedur oprogramowania systemowego i narzędziowego wykorzystywane przez oprogramowanie użytkowe.

Oprogramowanie użytkowe opracowywane dla potrzeb PZSD ZT dotyczy realizacji funkcji specyficznych dla wspomagania procesów dowodzenia w wojskach lądowych. Uwzględnia obowiązujące dokumenty regulujące proces dowodzenia przy prowadzeniu działań bojowych zarówno w układzie narodowym jak i w ramach struktur NATO. Uwzględnia również standardy i zalecenia agencji NATO w zakresie:

- obowiązujących dokumentów (STANAG 2014, STANAG 5500),
- wiadomości sformalizowanych (AdatP-3),
- znaków taktycznych (APP6A),
- modelu danych (ATCCIS),
- map numerycznych (VPF, CADRG)
- modelu terenu (DTED).

Oprogramowanie użytkowe zapewnia realizację zarówno podstawowych zadań systemu wymienionych powyżej jak i szeregu zadań specjalizowanych. Zapewnia również organizację funkcjonowania systemu w różnych warunkach.

Oprogramowanie użytkowe zapewni osobom funkcyjnym poszczególnych komórek organizacyjnych świadczenie następujących podstawowych usług:

- administrowania i zarządzania systemem;
- utrzymywania i zarządzania bazą danych;
- sporządzania i wymiany dokumentów i elementów danych;
- zobrazowania graficznego.

2.4.1 Podsystem Wymiany Dokumentów

W ramach realizowanych prac w ZSyD SZAFRAN ZT został opracowany Podsystem Wymiany Dokumentów (PWD). Wspomaga on pracę osób funkcyjnych na stanowiskach dowodzenia automatyzując procesy:

- tworzenia;
- przesyłania;

- archiwizowania;

dokumentów dowodzenia, zarówno wewnętrznych (opracowywanych dla potrzeb stanowiska dowodzenia danego szczebla) jak i zewnętrznych (opracowywanych dla potrzeb stanowisk dowodzenia szczebli nadrzędnych, podrzędnych i współpracujących).

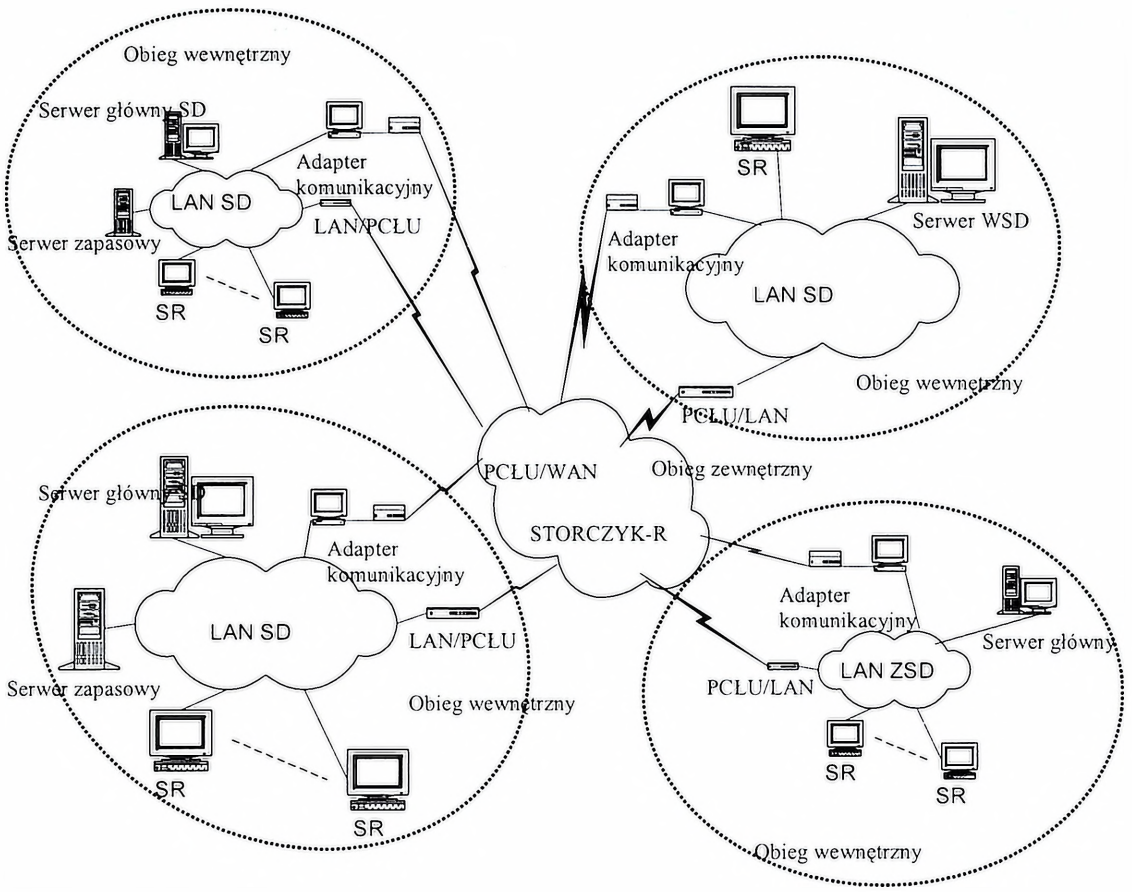
Opracowane oprogramowanie umożliwia:

- automatyzację procesu opracowywania, przechowywania i wymiany dokumentów;
- archiwizowania oraz wyszukiwania dokumentów;
- sformalizowanie postaci dokumentów wykorzystywanych w procesie dowodzenia;
- ujednolicenie sposobu wymiany dokumentów;
- kontrole dostępu do sytemu, dokumentów i realizowanych na nich funkcji;
- rejestrację działań użytkownika w systemie.

PWD pozwala także wymieniać standardowe wiadomości pocztowe i stanowi platformę komunikacyjną do wymiany wiadomości sformatowanych ADatP-3.

Oprogramowaniem narzędziowym do tworzenia PWD jest pakiet pracy grupowej Lotus Notes/Domino. Lotus Notes/Domino jest systemem bazodanowym, posiadającym własny format bazy. Jest to również system pocztowy, a raczej infrastruktura komunikacyjna, z pełną możliwością projektowania i programowania przepływu dokumentów i wiadomości. Co ważne, jest dostępny na różnych platformach systemowo – sprzętowych.

Podstawowym zadaniem serwera podsystemu jest współpraca z innymi serwerami dla zapewnienia wymiany dokumentów, wiadomości w ramach ZSyD i między różnymi ZSyD. To serwer stwierdza czy adresat informacji jest osiągalny w ramach systemu, czyli czy można do niego przesłać dokumenty lub wiadomości.



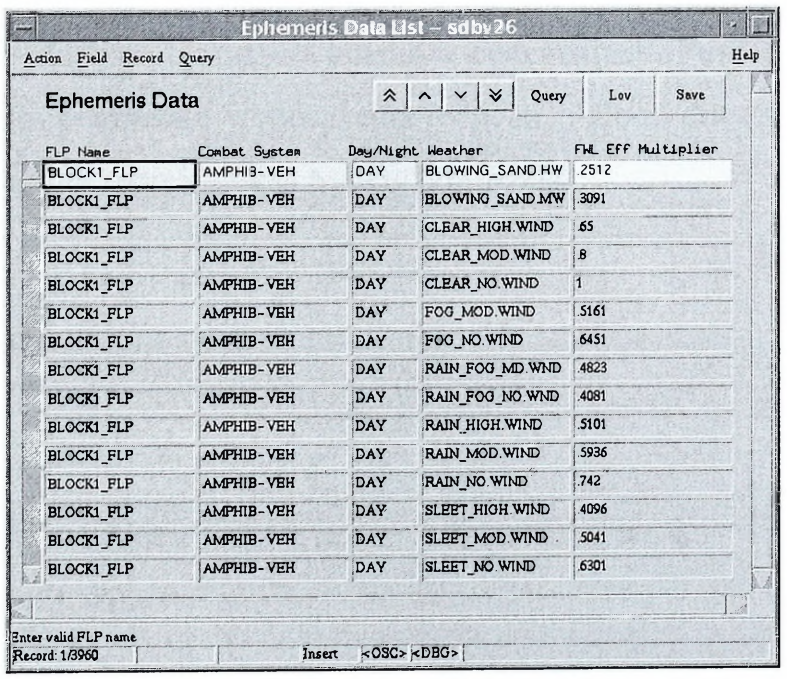
Rysunek 22 Schemat ideowy zewnętrznego i wewnętrznego obiegu dokumentów i wiadomości

/źródło: Bryła D., Sass S.: Integracja ZSyD Kolorado i SZAFRAN ZT na poziomie dokumentów i wiadomości/

3. Porównanie danych przetwarzanych w systemie JTLS i SZAFRAN

Systemy JTLS i SZAFRAN są systemami dwóch różnych klas. Pierwszy jest system symulacji walki, drugi natomiast systemem dowodzenia (C2I). Z tego powodu wynikają znaczne różnice w danych przetwarzanych w obu tych systemach.

System symulacyjny JTLS posiada w swoim składzie moduły odpowiedzialne za symulację prowadzonych działań bojowych. Stąd też musi on posiadać w swojej bazie danych informacje pozwalające na parametryzację zachowania się modelu. Parametry takie pozwalają na opis takich właściwości systemów walki jak np.: odporność na rażenie środkami walki, zdolność do zadawania strat. W wielu przypadkach wartościami, na których operuje model są współczynniki prawdopodobieństwa. Są to np.: prawdopodobieństwo trafienia, prawdopodobieństwo zniszczenia, prawdopodobieństwo podjęcia walki, prawdopodobieństwo wykrycia itd. Dodatkowo są one modyfikowane danymi mówiącymi o warunkach ich użycia i ich wpływie na modyfikowane właściwości symulowanych obiektów. Tutaj przykładem mogą być wartości modyfikujące skuteczność systemów walki w zależności od warunków pogodowych, które to z kolei również są modelowane w systemie.
Ponadto operuje on na założonym poziomie agregacji upraszczającym odwzorowanie rzeczywistości.

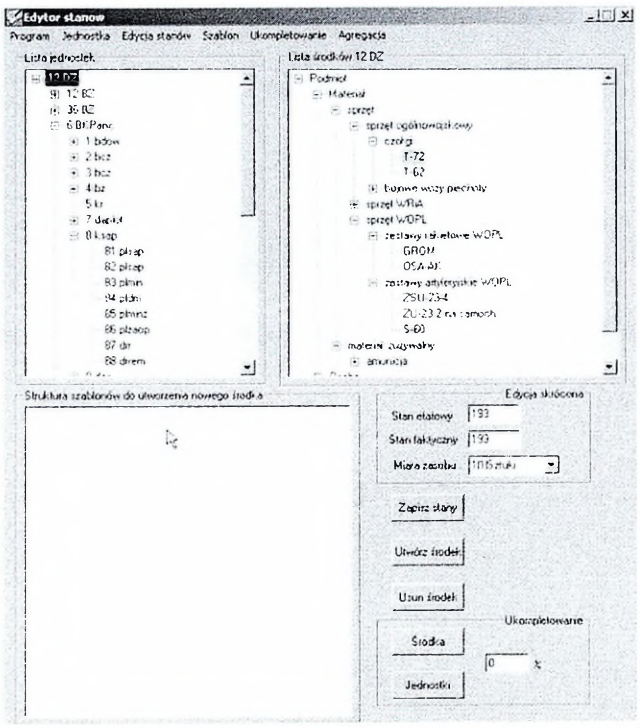


Rysunek 23 Parametry systemu JTLS modyfikujące skuteczność systemu walki w zależności od warunków pogodowych

Baza danych DDS zawiera 329 tabel z danymi opisującymi scenariusz gry oraz tabel pomocniczych wykorzystywanych w trakcie przetwarzania danych. Baza danych systemu

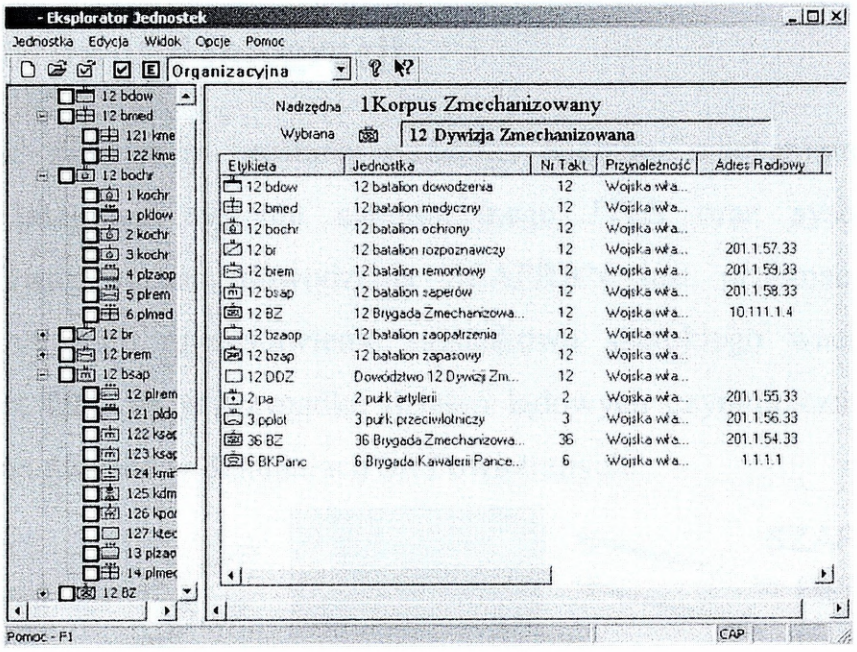
SZAFRAN oparta jest na koncepcji metamodelu zawierającego zarówno opis struktury bazy, jak i zawartości opisującej odwzorowywane rzeczywiste obiekty z wykorzystaniem mechanizmów wybranego RDBMS.

Zadaniem systemu SZAFRAN jest wsparcie procesów dowodzenia. Nie musi on zatem gromadzić żadnych parametrów modelowych, ponieważ jego zadaniem jest zbieranie i przetwarzanie informacji o rzeczywistej sytuacji na polu walki.



Rysunek 24 Dane o wyposażeniu jednostek w systemie SZAFRAN

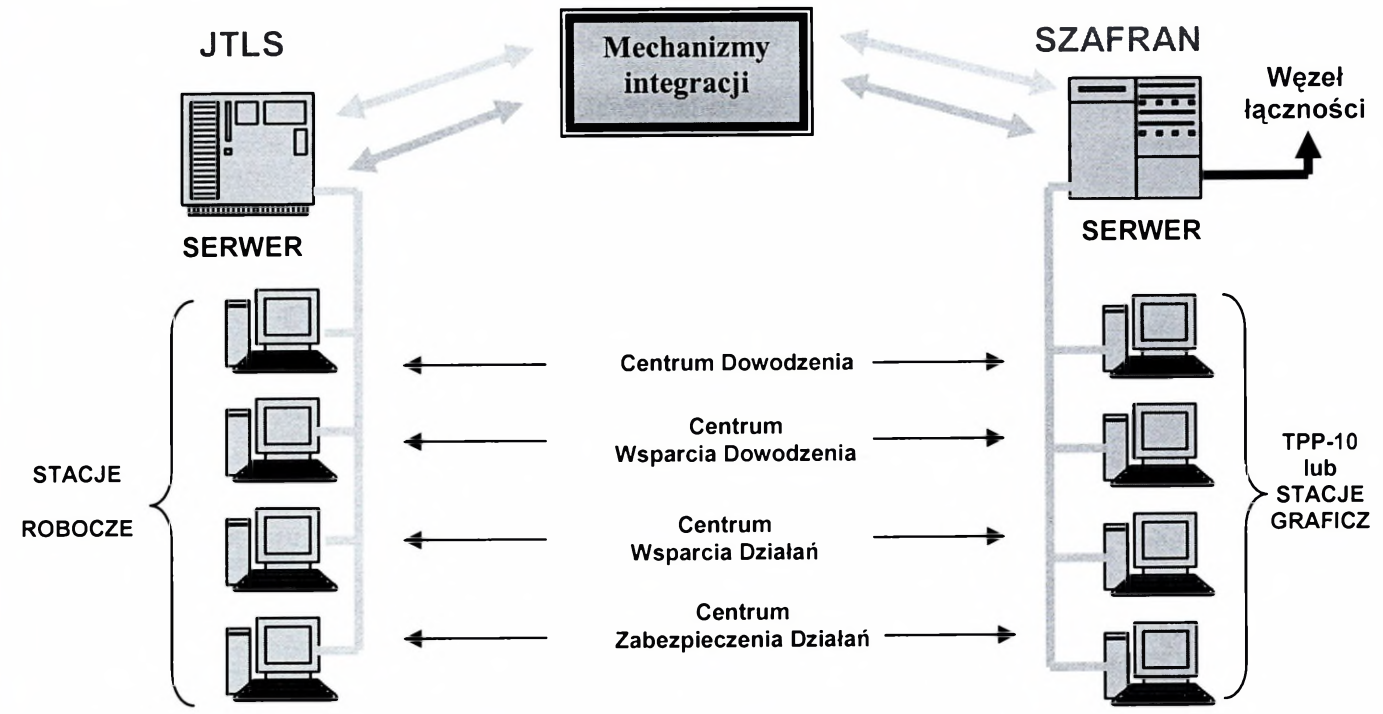
Wspólną cechą obu systemów pod kątem przechowywanych danych może być ilość, wielkość i struktura jednostek biorących udział w ćwiczeniach. Pewne podobieństwa mogą występować również w przypadku wyposażenia jednostek, ale to dotyczyć mogłoby z reguły tylko ich nazewnictwa. Ich reprezentacja byłaby odmienna.



Rysunek 25 Struktura jednostek w systemie SZAFRAN

4. Możliwości integracji

W rozdziale niniejszym zostaną poddane analizie mechanizmy, jakie można wykorzystać do integracji systemu symulacyjnego JTLS oraz systemu dowodzenia SZAFRAN. Ponieważ system dowodzenia SZAFRAN jest przeznaczony dla Wojsk Lądowych, możliwe było przedstawienie stosunkowo szerokiego wachlarza rozwiązań. Powodem tego jest fakt, że w przypadku działań lądowych czynnik czasu nie jest aż tak krytyczny, jak to ma miejsce w działaniach Sił Powietrznych.



Rysunek 26 Idea integracji systemu JTLS oraz SZAFRAN.

4.1 Poczta elektroniczna

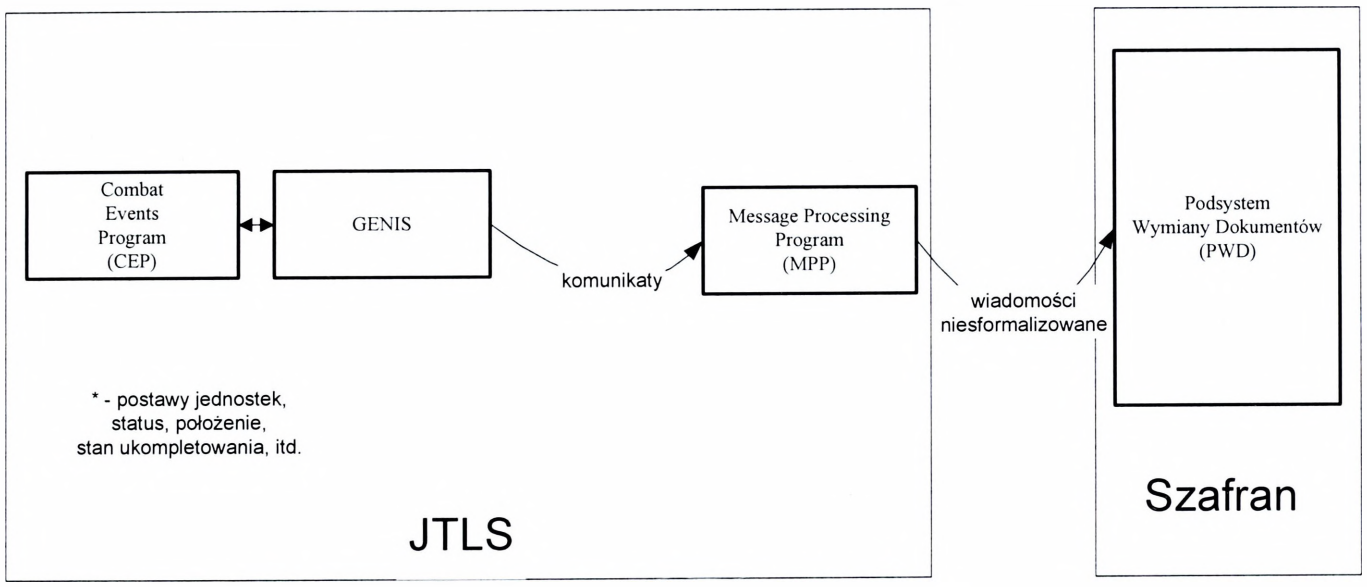
Oprogramowanie MPP, wchodzące w skład pakietu programów użytkowych systemu JTLS, posiada dwie funkcje warte uwagi przy rozpatrywaniu możliwości wymiany informacji z innymi systemami. Te funkcje, dotyczące otrzymywanych z CEP wiadomości, to:

- przesyłanie wiadomości poprzez e-mail z wykorzystaniem standardowego protokołu SMTP;
- formatowanie wiadomości pozwalające na nadanie im postaci niesformalizowanej lub zgodnej z formatem USMTF (ang. *US Message Text Format*).

4.1.1 Wiadomości niesformalizowane

Wykorzystanie wiadomości niesformalizowanych polega na prostym mechanizmie przesyłania wybranej wiadomości (w języku angielskim) z wykorzystaniem funkcjonalności analogicznej, jak posiadana przez każde oprogramowanie klienckie poczty elektronicznej – np. „Prześlij dalej" w MS Outlook Express.

Rozwiązanie takie nie wymaga dodatkowych komponentów programowych, ale z drugiej strony oferuje niewielką funkcjonalność i dostarcza informacji w formie mało przydatnej do dalszego przetwarzania przy wykorzystaniu narzędzi programowych.



Rysunek 27 Idea integracji systemów JTLS i SZAFRAN z wykorzystaniem mechanizmu przesyłania wiadomości niesformalizowanych
/źródło: opracowanie własne/

4.1.2 Wiadomości sformalizowane

Wykorzystanie funkcji formatowania komunikatów do postaci zgodnej z formatem USMTF stwarza możliwość automatycznego przetwarzanie informacji zawartych w przesyłanych przez CEP do MPP wiadomościach. W tym przypadku jednak powstaje konieczność dokonania konwersji z formatu USMTF używanego przez US Army, do formatu ADatP-3 przyjętego jako standard w krajach członkowskich NATO.

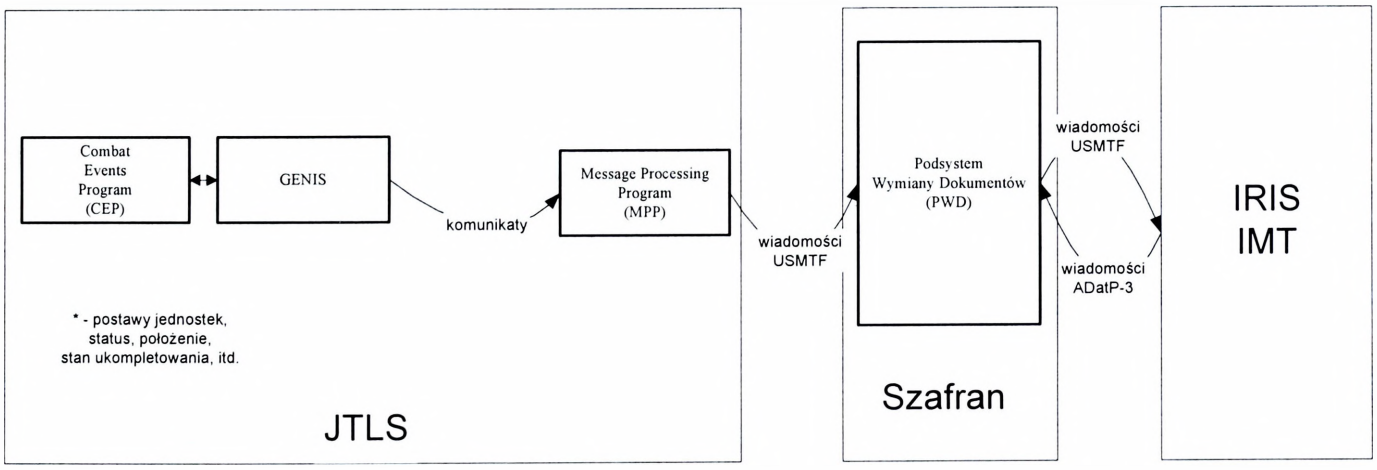
Format ADatP-3 został opracowany przez pakt NATO na podstawie formatu USMTF. Stąd też postać wielu wiadomości w formacie ADatP-3 jest w pełni zgodna z formatem USMTF. Jednak pomimo tego, że dwa formaty mogą mieć te same identyfikatory

wiadomości i w związku z tym być wiadomościami tego samego typu, istnieje kilka powodów, dla których nie będą one dokładnie zgodne:

- Część wiadomości została zmodyfikowana tak, aby spełnić specyficzne dla NATO wymagania. Modyfikacje te dotyczyły nie tylko przyjęcia dla poszczególnych pól innych wartości dopuszczalnych czy ich formatów, ale również pewnych szczegółów dotyczących podstawowych zasad formatowania. Doprowadziło to do niezgodności zarówno semantycznych, jak i syntaktycznych pomiędzy wiadomościami tego samego typu.

- Do formatów mogą być wprowadzane zmiany dla zapewnienia zgodności ze specyficznymi wymaganiami narodowymi. NATO definiuje zbiór wspólnych wiadomości dla wszystkich służb, które mogą być wykorzystywane przez wszystkie kraje członkowskie NATO do wymiany informacji z ich własnymi systemami oraz systemami innych krajów członkowskich. Jednak każdy kraj członkowski paktu może rozszerzyć format o nowe zbiory i pola zgodnie z własnymi wymaganiami.

- Formaty wiadomości ewoluują z czasem i w związku z tym we wdrożonych systemach mogą jednocześnie występować różne wersje tych samych formatów.



Rysunek 28 Idea integracji systemów JTLS i SZAFRAN z wykorzystaniem mechanizmu wiadomości sformalizowanych
/źródło: opracowanie własne/

4.1.3 Pakiet IRIS

Funkcjonalność zapewniającą konwertowanie wiadomości z formatu USMTF do formatu ADatP-3 dostarcza jedno z narzędzi pakietu IRIS firmy Systematic Software Engineering.

Trzy podstawowe elementy tego pakietu to:

- IRIS MFS - Message Formatting System,

- IRIS DEF - Message Definition System,
- IRIS IMT - Information Mapping Tool.

4.1.3.1 IRIS MFS

IRIS MFS zapewnia automatyczne przetwarzanie informacji dla systemów dowodzenia oraz wymianę informacji pomiędzy różnorodnymi systemami dowodzenia używanymi w poszczególnych krajach NATO.

IRIS MFS wspiera różne przyjęte w zastosowaniach wojskowych formaty transmisji wiadomości, między innymi ACP127, JANAP128, DD173, ACP126m, na równi z formatem e-mail opisanym w RFC 822. MFS Message Manager ułatwia zarówno tworzenie jak i odczytywanie wiadomości w tych formatach, w sposób czytelny dla ich użytkownika niezależnie od formatów, w jakich zostały stworzone, tym samym nie jest niezbędne posiadania wiedzy o użytych wcześniej formatach przez jego użytkownika. IRIS MFS zapewnia łatwy dostęp do wiadomości. Profilowanie i późniejsza automatyczna dystrybucja wiadomości do odpowiednich operatorów lub systemów komputerowych, celem szybkiego przetworzenia, jest podstawą działania systemu przekazywania wiadomości z ośrodków dowodzenia.

IRIS MFS umożliwia stworzenie konfigurowalnego obiegu dokumentów, w którym możliwe jest wysyłanie wiadomości w sposób sekwencyjny (wiadomość przechodzi łańcuch kolejnych opiniujących) lub bezpośrednio do końcowego adresata. Umożliwia odczyt wiadomości oraz dołączanie komentarza przez upoważnione do tego osoby. Wiadomości sformalizowane (przykładowo USMTF lub AdatP-3) mogą być tworzone przez operatora, używając modułu IRIS/MFS Body Editor. Używając modułu IRIS MFS MDE, wiadomość może być częściowo lub całkowicie automatycznie wygenerowana z różnych źródeł takich jak baza sytuacji. W czasie procesu obiegu dokumentów w momencie otrzymania sformatowanych wiadomości, może być ona automatycznie użyta do aktualizacji bazy danych (przy użyciu IRIS IMT) lub odczytana przez operatora w celu jej sprawdzenia i dokonania niezbędnych korekt.

4.1.3.2 IRIS DEF

IRIS DEF wspomaga definiowanie jednoznacznej wymiany formatów informacji.

IRIS DEF jest przeznaczony dla analityków danych i twórców MTF do definiowania i dystrybucji reguł MTF i kojarzenia tzw. tekstów pomocy, które pozwolą użytkownikom odczyt i tworzenie prawidłowych sformalizowanych wiadomości. Tworzenie MTF musi być zgodne z odpowiednimi specyfikacjami standardów używanych w danych krajach. Standardy te zawierają zbiór reguł, jakie musi spełniać każda wiadomość tekstowa. IRIS/DEF dostarcza metod zapewniających, że każda wiadomość tekstowa jest zgodna z przyjętymi standardami. IRIS/DEF obsługuje wiele różnych standardów w tym: ADatP-3 NATO; USMTF - siły zbrojne USA; VMF US Bit Oriented; ADFORM - Australia; OTH-T Gold - Marynarka USA.

4.1.3.3 IRIS IMT

Narzędzia IMT zostały zaprojektowane do modelowania i translacji danych. Zapewniają one automatyczne uzyskanie modeli danych z różnych źródeł i definiowanie łącz między dwoma modelami danych. Umożliwiają w prosty sposób tworzenie powiązań pomiędzy wiadomościami sformatowanymi oraz np. bazami danych będącymi składnikiem systemów dowodzenia (C4I) IMT składa się z dwóch części:

- modułu programistycznego – zawiera narzędzia do prac z modelami danych;
- modułu wykonawczego (ang. *runtime*) – odpowiada za przetwarzanie wiadomości w oparciu o wygenerowane i skompilowane wcześniej komponenty wytworzone w module programistycznym.

Najważniejszą cechą IMT jest system integracji danych z różnych źródeł niezależnie od ich formatów. IMT obsługuje modele ADatP3, USMTF, VMF oraz ADFORM oraz bazy danych np. Oracle, Sybase lub ODBC.

Idea IMT opiera się na wprowadzeniu koncepcji reprezentacji formatu danych za pomocą modelu danych. Wykorzystanie modelu danych pozwala na ujednolicenie przetwarzania danych niezależnie od ich rzeczywistego formatu i tym samym ukrycie szczegółów technicznych związanych z formatowaniem. Abstrakcyjne podejście do przetwarzania i przechowywania danych upraszcza integratorom zunifikowanie różnych formatów bez specjalistycznej wiedzy i olbrzymiego nakładu pracy. Dzięki czemu możliwa

jest znaczna redukcja kosztów wdrożenia i użytkowania, a także czasu potrzebnego do wdrożenia i uaktualniania systemu. IRIS/IMT umożliwia ponadto automatyczne pobieranie informacji ze źródeł o różnych modelach danych i miejscach pochodzenia.

IRIS IMT jest niezależny od stosowanych formatów i producentów baz danych. Dowolne dwa źródła (struktury) informacji reprezentowane jako modele IMT mogą zostać użyte jako źródło lub cel mapowania informacji. Mapowanie informacji wykonywane jest z wykorzystaniem interfejsu graficznego poprzez tworzenie połączenia pól z poszczególnych formatów i ewentualne specyfikowanie odpowiednich przekształceń. Jeżeli jest to konieczne można użyć języków programowania Java lub C#. Po zakończeniu tworzenia specyfikacji mapowania generowany jest kod źródłowy w języku Java lub C#, który jest następnie kompilowany. Tak wytworzone moduły są umieszczane w systemie, gotowe do późniejszego użycia.

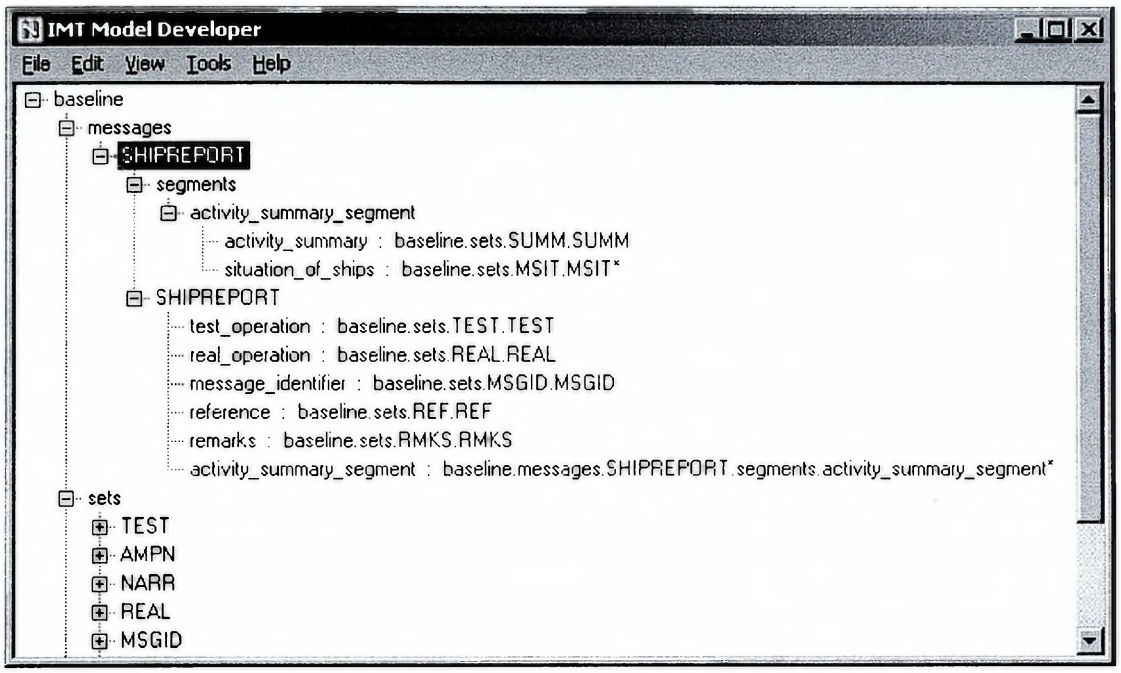
System oparty jest na architekturze modułowej, umożliwiając tym samym używanie tylko niezbędnych komponentów systemu IMT.

4.1.3.3.1 Tworzenie modelu w IMT

To, że IMT wspiera liczne formaty baz danych oraz wiadomości tekstowych oznacza, że ich modele mogą być generowane automatycznie, jak również możliwe jest wykonywanie mapowania informacji pomiędzy nimi. IMT jest dostarczany ze wsparciem dla baz danych, które są zgodne ze standardem OLE DB oraz dla formatów wiadomości tekstowych obsługiwanych przez IRIS/MFS i IRIS/DEF, jak również dla formatów XML-MTF oraz XML. Ten zestaw może być później łatwo rozszerzony o własne formaty.

Dla każdego formatu istnieją dwie tzw. wtyczki (ang. *plug-in*). Jedna do generowania modelu danych IMT oraz druga dostarczająca interfejsu dostępu do danych w środowisku wykonawczym (ang. *runtime*). Nowe formaty mogą być obsługiwane po dodaniu nowych wtyczek.

Modele danych są generowane i wyświetlane przy użyciu aplikacji graficznej IMT Model Developer. Jeżeli w formacie danych zostały wyrażone poprawne i dopuszczalne wartości, mogą być one później pobierane i wykorzystywane w procesie mapowania informacji. Modele danych są zapisywane w oddzielnych plikach w formacie XML, które są później wykorzystywane przez narzędzie mapowania.

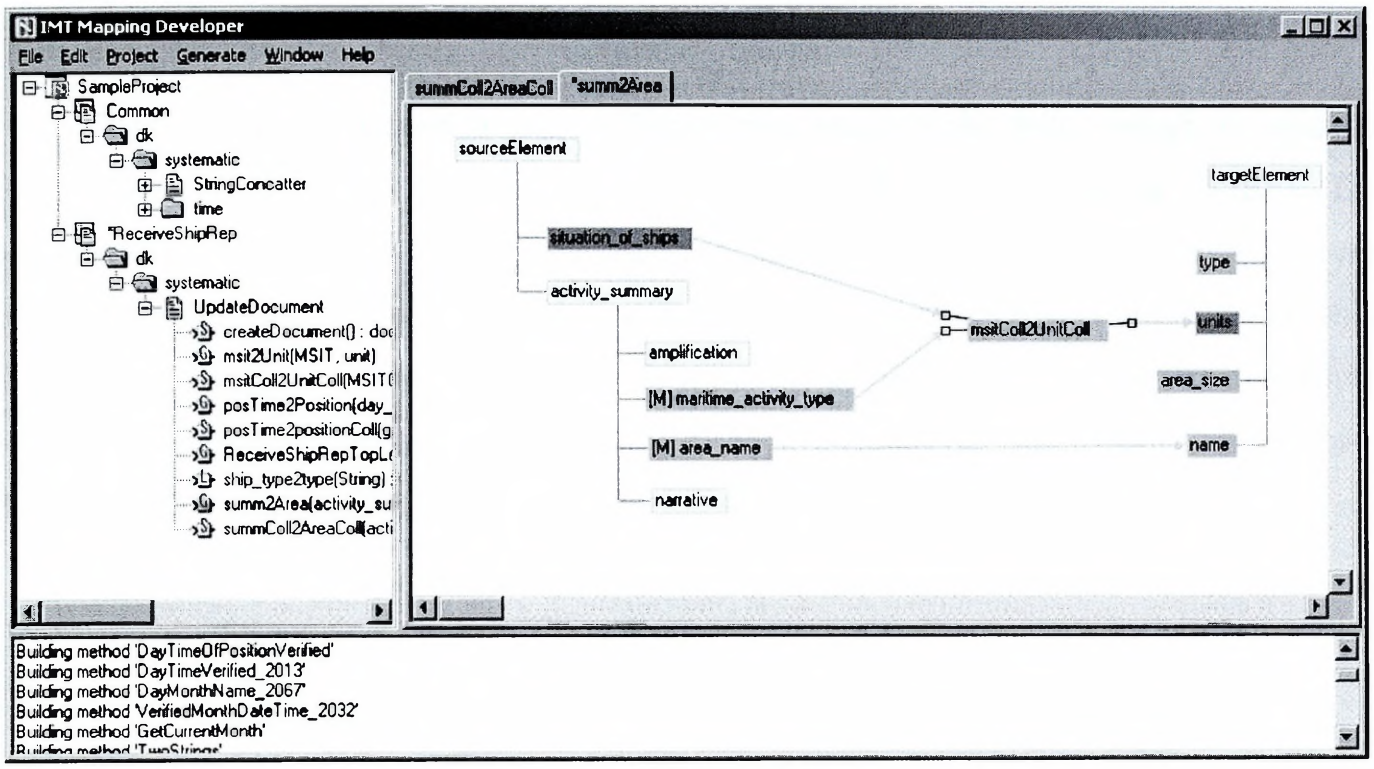


Rysunek 29 Model danych wiadomości ADatP-3

4.1.3.3.2 Tworzenie mapowania

Mapowania tworzy się w narzędziu o nazwie IMT Mapping Developer i, podobnie jak modele danych, są one zapisywane w oddzielnych plikach w formacie XML. Wykonywanie znacznej liczby specyficznych czynności niezbędnych do przeprowadzenia procesu tworzenia mapowania ułatwiają kreatorzy mapowania.

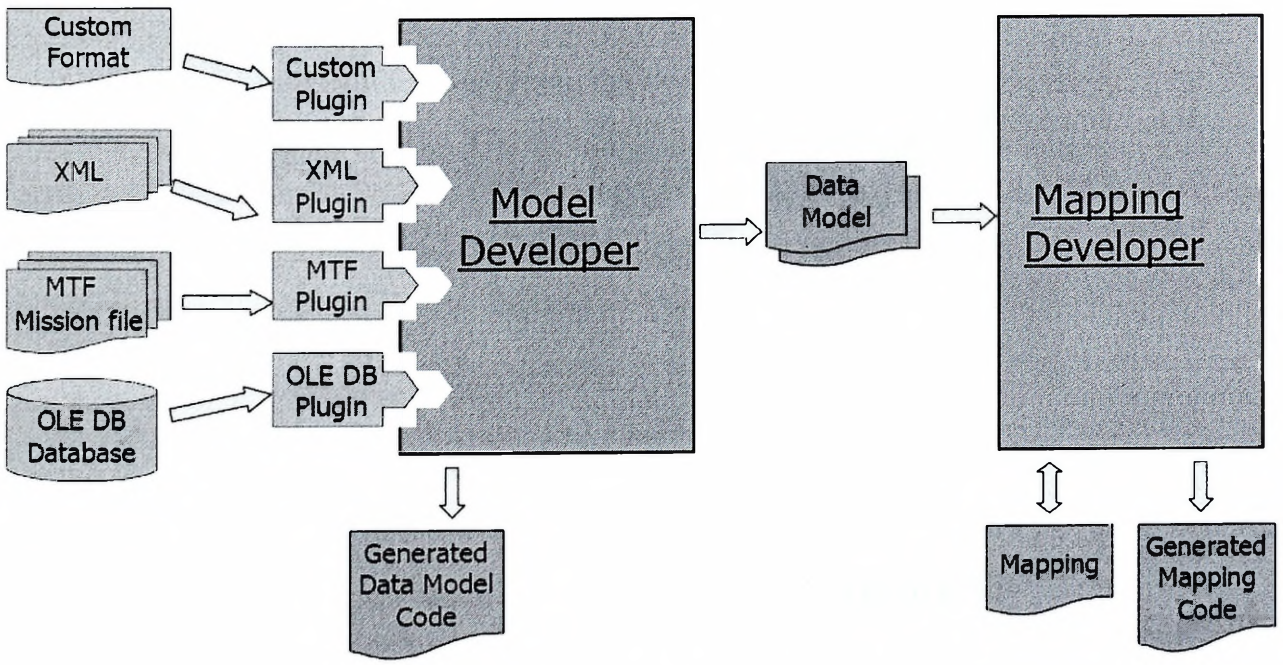
Większość typowych zadań mapowania informacji zawartych w poszczególnych formatach może być wykonana przy użyciu narzędzi graficznego mapowania. Jednak w przypadku bardziej skomplikowanych operacji można je wyspecyfikować w dostępnych językach programowania (Java i C#). Do przeprowadzania bardzo skomplikowanych operacji mapowania można wykorzystać funkcjonalność pozwalającą na wykorzystanie już istniejących funkcji (zaimplementowanych w Java lub C#) lub implementację nowych.



Rysunek 30 Edytor tworzenia mapowania

Zarówno podczas tworzenia mapowania w trybie graficznym, jak i kompilacji wygenerowanego kodu, przeprowadzana jest bardzo silna kontrola zgodności typów danych.

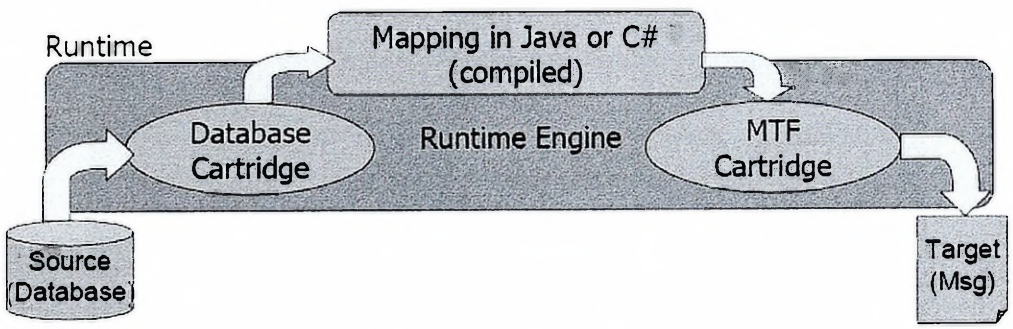
W przypadku zmian w formatów konieczne jest wygenerowanie nowych modeli oraz aktualizacja dotyczących ich transformacji informacji.



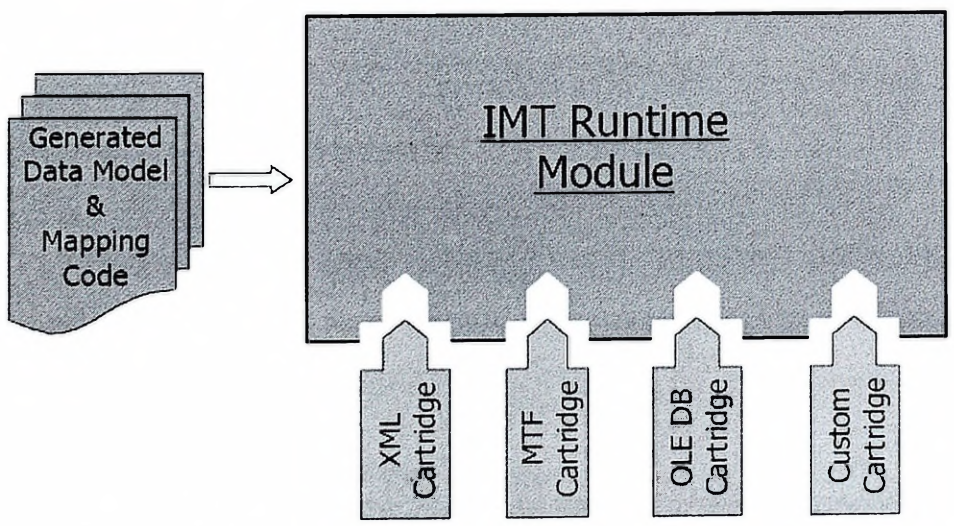
Rysunek 31 Architektura modułu programistycznego IMT
/źródło: IRIS Information Mapping Tool/

4.1.3.3.3 Wykonywanie mapowania

IMT dostarcza wsparcia dla wykonywania mapowania jako usługi w oparciu o platformę Microsoft .NET platform oraz Java 2 Application Server (J2EE). Do wykonywania mapowania wykorzystywany jest specjalny rodzaj wtyczek (ang. *plug-in*) określanych jako kasety (ang. *cartridges*). W środowisku wykonawczym silnik mapowania IMT wykorzystuje mechanizm kaset do dostępu do danych zapisanych w specyficznym formacie. Mechanizm ten jest odpowiedzialny za odczyt, zapis oraz zapewnienie integralności danych poprzez zastosowanie mechanizmów transakcji, jeżeli jest możliwe.



Rysunek 32 Mechanizm kaset
/źródło: IRIS Information Mapping Tool/



Rysunek 33 Architektura modułu wykonawczego IMT
/źródło: IRIS Information Mapping Tool/

4.1.3.3.4 Interfejs klientów modułu wykonawczego IMT

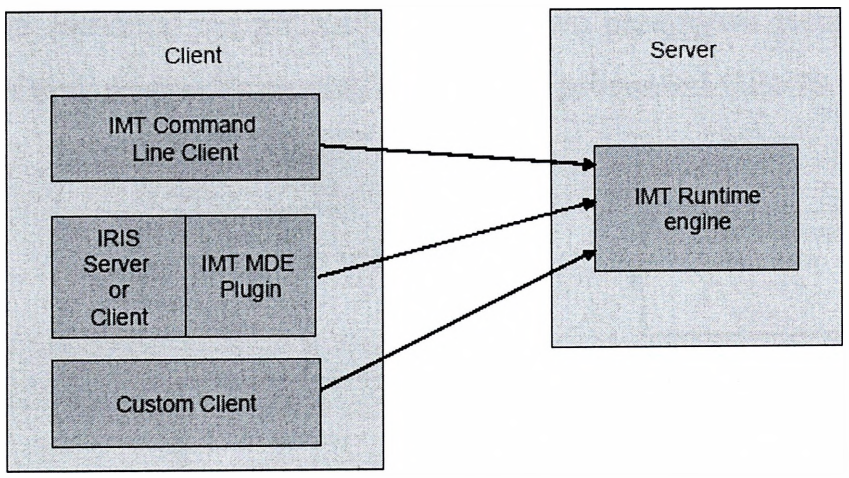
IMT jest dostarczane wraz z implementacją protokołu SOAP (ang. *Simple Object Access Protocol*) pozwalającego na zdalne wywoływanie mechanizmów mapowania

informacji między różnymi formatami oraz zwracanie statusu i wyników wykonania. Protokół SOAP bazuje na XML dla wymiany strukturalizowanych informacji.

Implementacja SOAP dostarczana jest wraz z pakietem IMT w postaci dwóch klientów pozwalających wywoływanie mechanizmów mapowania:

- wtyczka IRIS MDE (ang. *message data exchange*) – pozwalająca na wykorzystywanie mechanizmu mapowania z pakietu IRIS MFS;
- IMT Command Line Client – prosty klient pozwalający na wywoływanie mechanizmów mapowania wiadomości przy wykorzystaniu interfejsu linii komend.

Możliwe jest również wykonanie własnych modułów klienckich w oparciu o wykorzystanie protokołu SOAP.



Rysunek 34 Interfejs klientów modułu wykonawczego IMT
/źródło: IRIS Information Mapping Tool/

4.2 Open Access Programs

Programy otwartego dostępu (*ang. Open Access Programs*) pozwalają na dostęp do danych o grze w czasie jej trwania za pośrednictwem bazy danych, w której są składowane. W skład Open Access Programs w chwili obecnej wchodzą (w przyszłości ich liczba może się zwiększyć):

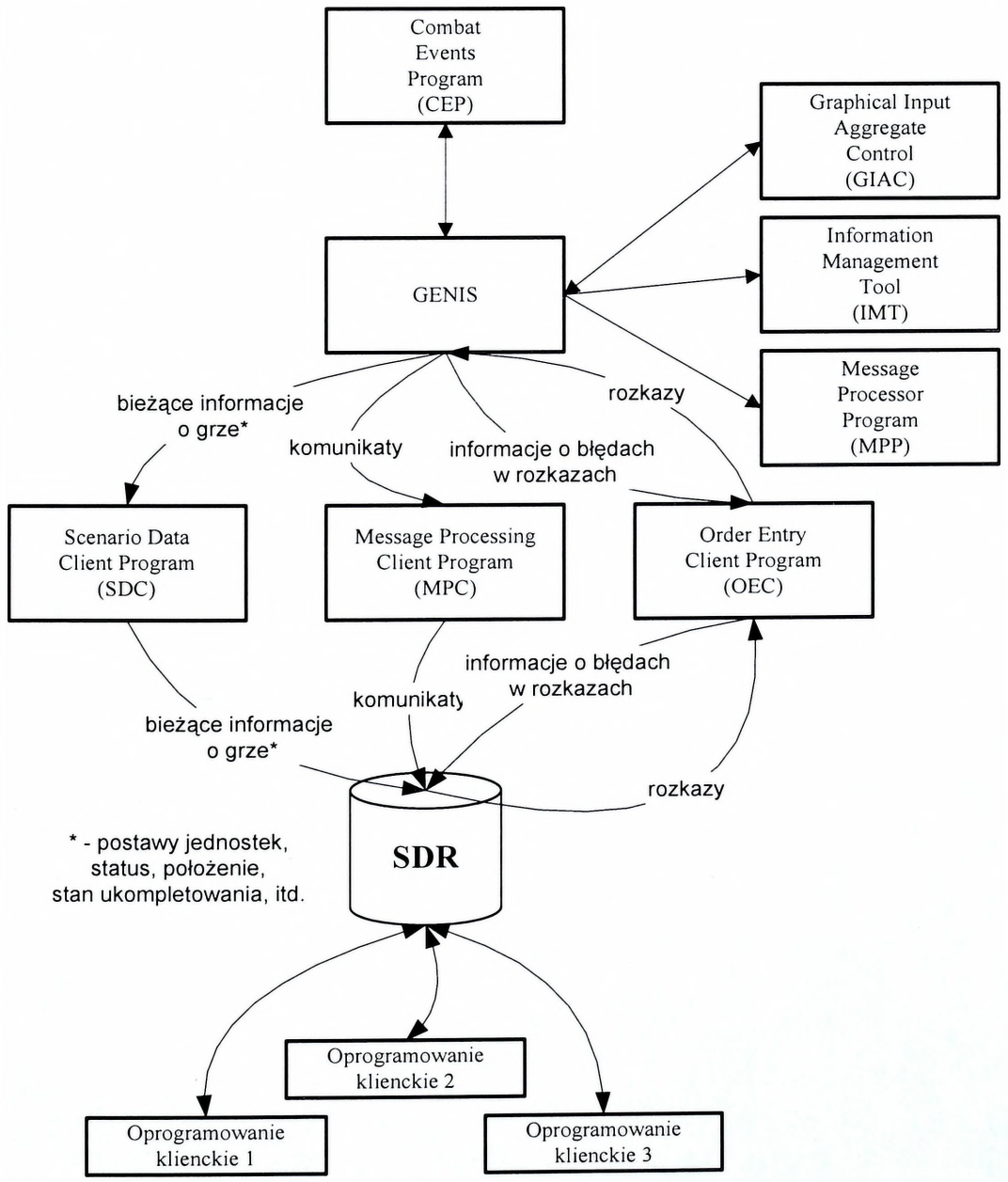
- SDC (ang. *Scenario Data Client*) – program kliencki danych scenariusza;
- MPC (ang. *Message Processing Client*) – program kliencki przetwarzania komunikatów;
- OEC (ang. *Order Entry Client*) – program kliencki wprowadzania rozkazów.

Do przechowywania danych z gry w czasie ćwiczenia wykorzystywane jest repozytorium danych scenariusza (*ang. SDR - Scenario Data Repository*). Jest ono zaimplementowane jako baza danych Oracle zaprojektowana pod kątem przechowywania

aktualnej informacji o sytuacji w grze aktualizowanej w czasie rzeczywistym. Ponieważ wspiera ono otwarte standardy, do danych z gry mogą mieć dostęp aplikacje firm trzecich takie jak narzędzia raportujące lub systemy C4I.

Open Access Programs również (podobnie jak np.: GIAC) są programami klienckimi GENIS komunikującymi się z GENIS w celu odbierania bieżących danych o grze, które następnie przekazują do bazy danych SDR. Podczas inicjalizacji program ustanawia połączenie z modelem JTLS za pośrednictwem GENIS oraz z bazą danych SDR.

Każda aplikacja, która potrzebuje otrzymywać dane o ćwiczeniu, może połączyć się z SDR i uzyskać potrzebne informacje. Przepływ danych pomiędzy tymi programami obrazuje rysunek 35 (schemat nie pokazuje szczegółowo przepływu informacji nieistotnych dla opisu działania Open Access Programs, np.: między procesami GENIS i CEP).



Rysunek 35. Przepływ danych między SDR i JTLS

/źródło: opracowanie własne na podstawie JTLS Technical Coordinator's Guide/

4.2.1 Scenario Data Client (SDC)

Scenario Data Client (SDC) jest odpowiedzialny za przekazywanie i aktualizację Scenario Data Repository (SDR) aktualnymi informacjami o grze dostarczanymi przez GENIS. Wewnątrz SDR istnieją oddzielne tabele Oracle dla każdego typu danych otrzymywanych od GENIS. Przeważająca większość tych tabel dziedziczy swój format bezpośrednio ze struktur danych GENIS.

SDC może zażądać całej informacji o scenariuszu oraz odbierać wszystkie informacje generowane przez GENIS i przekazywać te dane do SDR, lub może zostać skonfigurowany tak, żeby żądać informacji o wybranych stronach (lub nawet jednej stronie) gry.

Od chwili, kiedy początkowe rekordy inicjalizujące zostaną zapisane w tabelach SDR, SDC jest odpowiedzialny za propagowanie informacji aktualizujących oraz usunięć otrzymywanych od GENIS tak, aby informacje zawarte w tabelach odpowiadały bieżącej sytuacji w grze.

System nie dostarcza możliwości równoległej pracy klientów SDC (uruchomienia kilku instancji jednocześnie). W danej chwili powinien być uruchomiony tylko klient dla wszystkich stron (opcja zalecana) lub dla wybranej strony, ale nigdy dwaj klienci.

4.2.2 Message Processing Client (MPC)

Message Processing Client (MPC) jest odpowiedzialny za umieszczanie w Scenario Data Repository (SDR) komunikatów generowanych przez CEP. Powstał on na bazie programu MPP opracowanego przez NC3A, poprzez wprowadzenie modyfikacji i rozszerzenie funkcjonalności, jednak logika przetwarzania komunikatów pozostała taka sama.

Jedna instancja MPC jest uruchamiana dla scenariusza do odbierania i składowania komunikatów o całej sytuacji i wszystkich stronach biorących udział w grze. Każde program kliencki może uzyskiwać dane dla określonej strony w grze z SDR

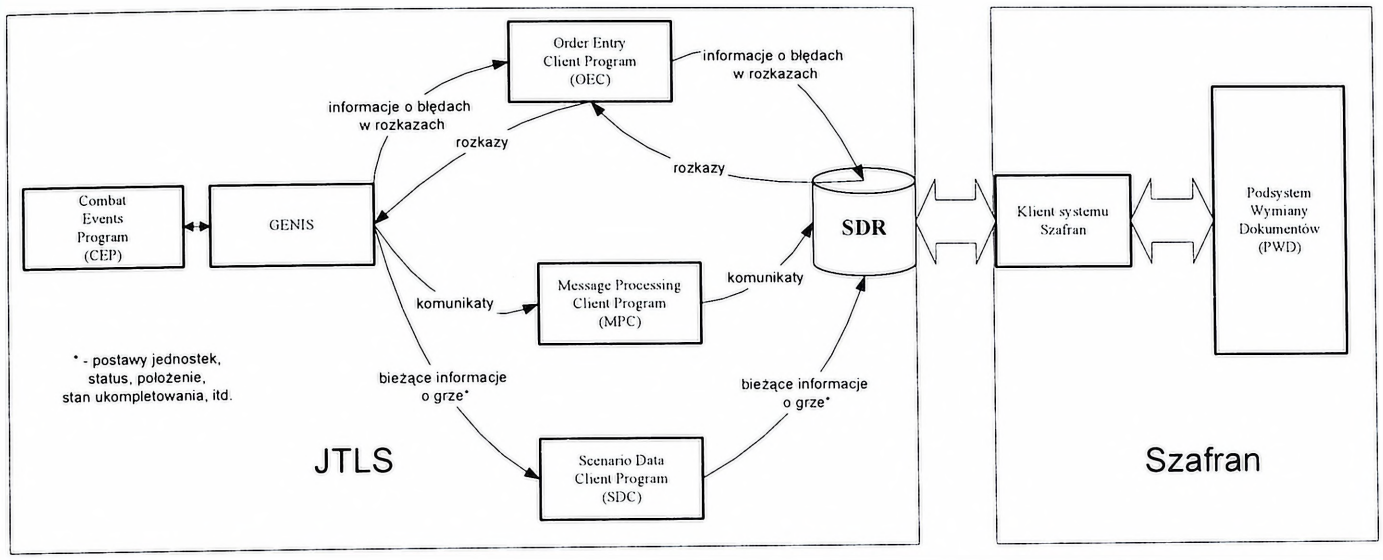
Wewnątrz SDR istnieją oddzielne tabele Oracle do przetwarzania i składowania poszczególnych typów komunikatów. MPC może zostać skonfigurowany do odbierania wszystkich komunikatów od CEP i składowania w tabeli `general_message`.

4.2.3 Order Entry Client (OEC)

Order Entry Client (OEC) sprawdza tabele bazy danych wewnątrz SDR (Scenario Data Repository) pod kątem rozkazów, które zostały w nim umieszczone w celu przesłania ich do CEP. Jeżeli rozkazy zostały umieszczone w SDR, OEC jest odpowiedzialny za ich pobranie, sformatowanie, weryfikację oraz przesłanie do JTLS. Jeżeli rozkaz zawiera błędy, OEC informuje o tym fakcie oprogramowanie klienckie, które go przesłało, za pośrednictwem komunikatów o błędach umieszczanych w tabeli bazy danych. Oprogramowanie klienckie jest odpowiedzialne za przesyłanie rozkazów oraz monitorowanie statusu ich przesłania do JTLS.

Struktura większości tabel bazy danych SDR wykorzystywanych do przekazywania rozkazów do OEC jest oparta na formacie pliku definicji rozkazów `orders.def`. Ponieważ plik `orders.def` jest plikiem danych podlegającym dynamicznym zmianom, którego postać zwykle zmienia się pomiędzy wersjami JTLS, pociąga to za sobą zmiany zarówno w tabelach bazy danych jak i w kodzie używanym do korzystania z nich. Dlatego też istnieje generator kodu do tworzenia skryptów definiujących tabel jak i kodu umożliwiającego korzystanie z nich. Generacja skryptów i kodu jest wykonywana przy użyciu OPP (ang. *Order Pre-Processor*).

OEC przeprowadza weryfikację rozkazów wykorzystując do tego celu OVT-FL (ang. *Order Verification Tool Functional Library*) – bibliotekę funkcji narzędzia weryfikacji rozkazów. OEC inicjalizuje bibliotekę z nazwą scenariusza, aktualizuje nazwę klienta i strony przed wysłaniem rozkazu do weryfikacji i uaktualnia te informacje na bieżąco przekazując do niej wszystkie niezbędne informacje z pakietów GENIS. Ponieważ każdy klient każdej strony może przesłać rozkazy poprzez OEC, musi on dostarczać wszystkie dane dla OVT-FL niezbędne do przeprowadzenia weryfikacji.

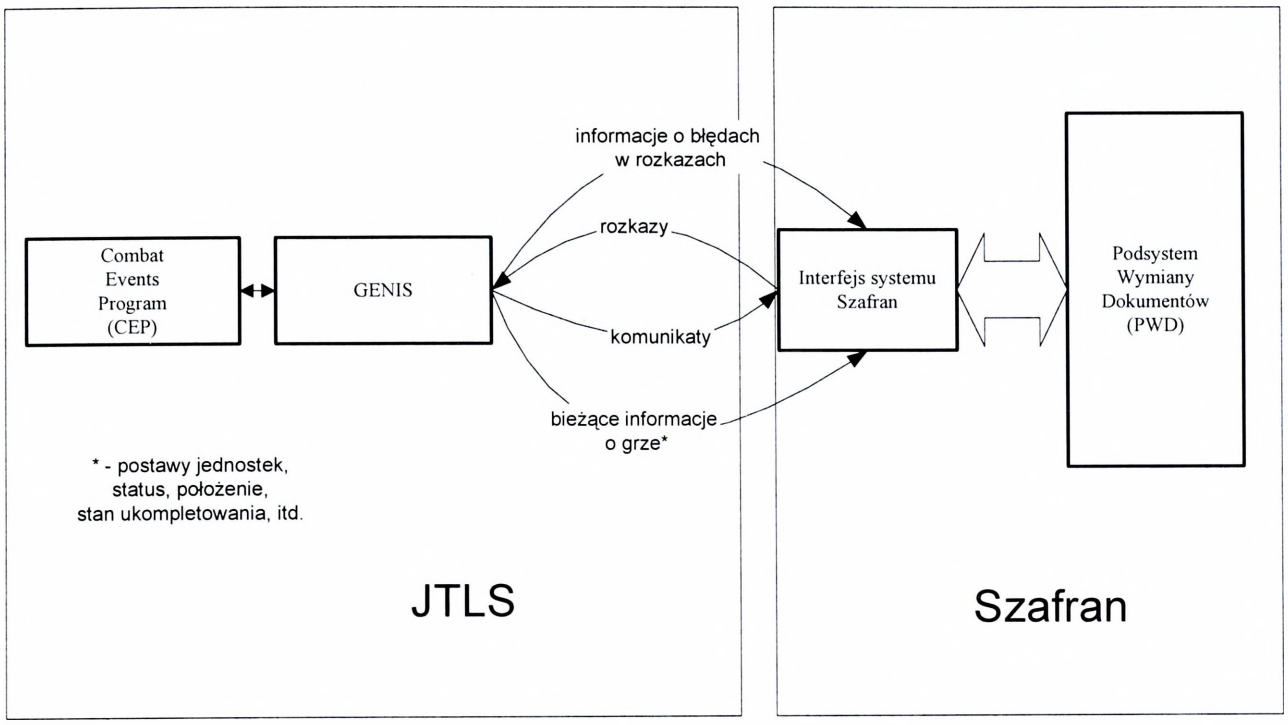


Rysunek 36 Idea integracji systemów JTLS i SZAFRAN z wykorzystaniem mechanizmów Open Access Programs
/źródło: opracowanie własne/

4.3 Integracja – specjalizowane interfejsy

Logicznym następstwem poprzednich dwu rozwiązań byłoby rozwiązanie eliminujące zbędne elementy pośrednie w kontakcie między kluczowym elementem systemu JTLS, czyli modułem CEP implementującym modele zastosowane w tym systemie, a systemem dowodzenia, czy w analizowanym przypadku – systemem SZAFRAN.

Takie rozwiązanie pozwoliłoby na otrzymywanie wszystkich informacji, jakie jest w stanie wygenerować model i konwersję danych wprost do pożądanej postaci (np. wiadomości ADatP-3). Dodatkowo możliwe byłoby przyjmowanie rozkazów z systemu dowodzenia bezpośrednio w formacie wykorzystywanym w nim i translacja ich do postaci akceptowanej przez system symulacyjny.



Rysunek 37 Idea integracji systemów JTLS i SZAFRAN z wykorzystaniem specjalizowanego interfejsu
/źródło: opracowanie własne/

4.4 Integracja na poziomie baz danych

Na podstawie pobieżnej lektury dokumentacji systemu JTLS oraz analizy rozwiązań technologicznych wykorzystanych w systemie SZAFRAN można byłoby wysnuć hipotezę, że integrację między dwoma systemami można próbować przeprowadzić bezpośrednio na poziomie baz danych. Przemawiałoby za tym chociażby wykorzystanie ogólnie znanych i dostępnych standardowych rozwiązań z zakresu RDBMS – Oracle w przypadku JTLS oraz Informix w przypadku systemu SZAFRAN.

Nie jest w tym przypadku istotne, czy rozwiązanie opierałoby się na koncepcji pracy na wspólnej bazie danych, okresowej dystrybucji wybranego zakresu informacji, czy też dystrybucji zmian zachodzących w bazach. Podstawowym problemem w tym przypadku jest fakt, że środowisko RDBMS Oracle w systemie JTLS (moduł DDS) jest wykorzystywany tylko w fazie przygotowywania scenariusza ćwiczeń. Inicjalizacja gry następuje na podstawie plików tekstowych wygenerowanych z modułu DDS, które są następnie ładowane do struktur danych przechowywanych w pamięci operacyjnej komputera, na którym działa moduł CEP. Tak więc w tym przypadku nie odgrywa roli nawet fakt, że dane niezbędne do działania obu systemów różnią się dość zasadniczo z tego powodu, że muszą one ze względu na swoją specyfikę opisywać rzeczywiste obiekty w znacząco różniący się sposób.

Jedynym elementem, który można byłoby wykorzystać z takiej koncepcji byłoby skoordynowanie przygotowania danych początkowych dla obu systemów tak, aby bazy

danych opisujące stan początkowy nie odzwierciedlały różnych sytuacji wyjściowych. Dotyczyłoby to zatem np.: liczby, rodzaju, wielkości i wyposażenia jednostek biorących udział w ćwiczeniach. Osobnym zagadnieniem w tym przypadku jest odwzorowanie terenu, ponieważ system SZAFRAN wykorzystuje rzeczywiste mapy cyfrowe, natomiast system JTLS odwzorowuje teren z wykorzystaniem regularnej siatki heksagonalnej uśredniając parametry terenowe danego obszaru.

4.5 Standard HLA

Aktualnie stosowane i projektowane systemy symulacyjne oraz systemy wspomagania dowodzenia są lub muszą pozwalać na wymianę danych na bazie platformy systemowej nazywanej HLA (ang. *High Level Architecture*). Standard HLA obejmuje w sobie szereg zagadnień ułatwiających projektowanie i integrację systemów symulacyjnych, ograniczając liczbę koniecznych decyzji konstrukcyjnych przez narzucenie określonych rozwiązań. Przy łączeniu systemów opracowanych zgodnie ze standardem HLA, zbiór zagadnień koniecznych do rozwiązania staje się zawężony – lecz należy zaznaczyć – nie oznacza to, że jest to zadanie łatwe. Konieczne jest dokładne zdefiniowanie zasad wymiany danych, kontroli symulacji, modeli obiektowych itp.

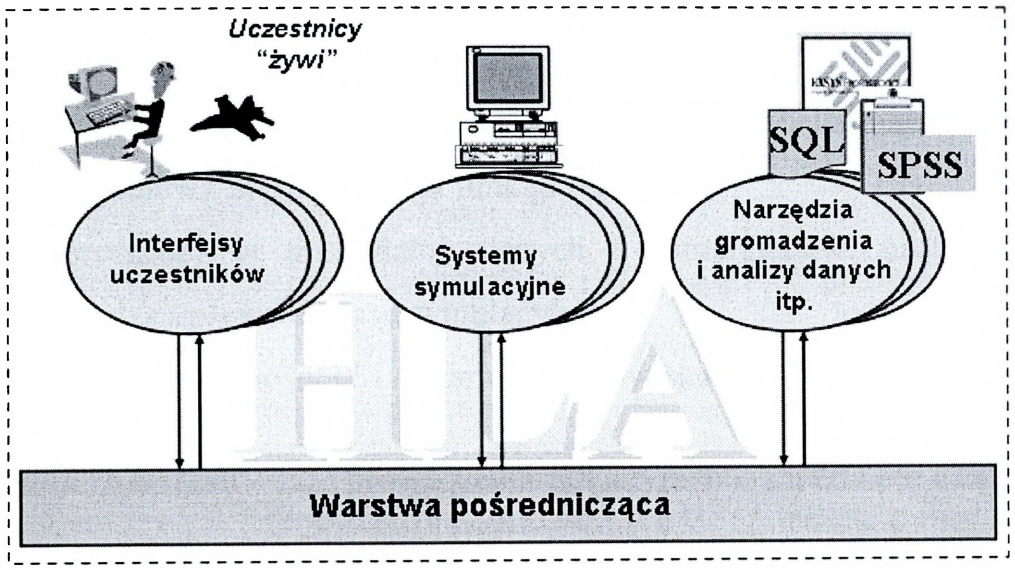
Prace badawcze nad systemami interaktywnej symulacji rozproszonej prowadzone są od wielu lat. Departament Obrony USA od 1996r. rozwija standard architektury i komunikacji w tego typu symulacji, nazwany HLA. Od kwietnia 2000 roku jest on oficjalnym standardem IEEE (IEEE 1516) dla symulacji, uznanym także przez NATO. Specyfikacja HLA definiuje pewne standardy używane przy projektowaniu symulacji oraz szereg usług komunikacyjnych zawartych w RTI (ang. *Runtime Infrastructure*).

HLA to najnowszy standard integracji rozproszonych symulatorów, ludzi i sprzętu rzeczywistego. Kontynuuje wcześniejsze doświadczenia uzyskane przez DoD przy opracowywaniu protokołów pozwalających integrować pojedyncze symulatory do bardziej złożonych systemów rozproszonych:

- DIS (ang. *Distribute Interactive Simulation*) – protokół umożliwiający prowadzenie ćwiczeń w czasie rzeczywistym, definiujący standardowe struktury komunikatów oraz reguły ich wymiany (IEEE 1278-1993);
- ALSP (ang. *Aggregate Level Simulation Protocol*) – protokół umożliwiający współdziałanie konstruktywnych symulatorów pola walki, przeznaczony dla

elementów zagregowanych, konserwatywny mechanizm upływu czasu symulacyjnego.

HLA to wspólna architektura umożliwiająca prowadzenie eksperymentów symulacyjnych w syntetycznym środowisku rozproszonym.



Rysunek 38 Syntetyczne środowisko symulacji

Jest standardem integrującym ludzi, sprzęt i systemy informatyczne, w tym różnego typu symulatory. Jest zdefiniowana przez trzy podstawowe dokumenty:

- HLA Framework and Rules – IEEE 1516;
- HLA Federate Interface Specification – IEEE 1516.1;
- HLA Object Model Template (OMT) – IEEE 1516.2.

Pojęcia wprowadzone przez standard IEEE 1516 to:

- federacja (ang. *federation*) – nazwana trójka:
  - zbiór elementów środowiska symulacyjnego odwzorowującego złożony fragment rzeczywistości, wyróżniony z określonego punktu widzenia, np. wewnętrznych powiązań;
  - model FOM (ang. *Federation Object Model*) opisujący wspólne własności i zachowania elementów federacji;
  - warstwa pośrednicząco-zarządzająca RTI.

- federat (ang. *federate, simulation*) – element składowy federacji ze skojarzonym modelem SOM (opis własności i zachowania); może nim być np:
  - program symulacyjny,
  - symulator sprzętowy,

  - człowiek lub element rzeczywisty z interfejsem do systemu,
  - narzędzie wspomagające gromadzenie i przetwarzanie danych symulacyjnych.

- RTI (ang. *Runtime Infrastructure*) – współdzielona przez wszystkie elementy środowiska symulacyjnego (federacje i federatów) warstwa pośrednicząco-zarządzająca:
  - pełni rolę podobną do roli systemu operacyjnego obsługującego programy;
  - programowa implementacja funkcji HLA IS;
  - pośredniczy w przesyłaniu danych i komunikatów, nadzoruje interakcje, zarządza upływem czasu symulacyjnego;
  - zapewnia przenośność i otwartość środowiska.

- Federation Execution – faza prowadzenia eksperymentu z udziałem danej federacji.

- interakcja (ang. *interaction*) – zdarzenie, generowane przez jednego federata do innego (lub innych), oznakowane czasem, przesyłane przez RTI. Istnieją klasy interakcji analogiczne do klas obiektów.

- parametr (ang. *parameter*) – nazwana dana skojarzona z instancją klasy interakcji, podobnie jak atrybut z obiektem.

- SOM (ang. *Simulation Object Model*) – formalny opis danych (obiekty, atrybuty, interakcje, parametry) udostępnianych przez federata („interfejs" federata).

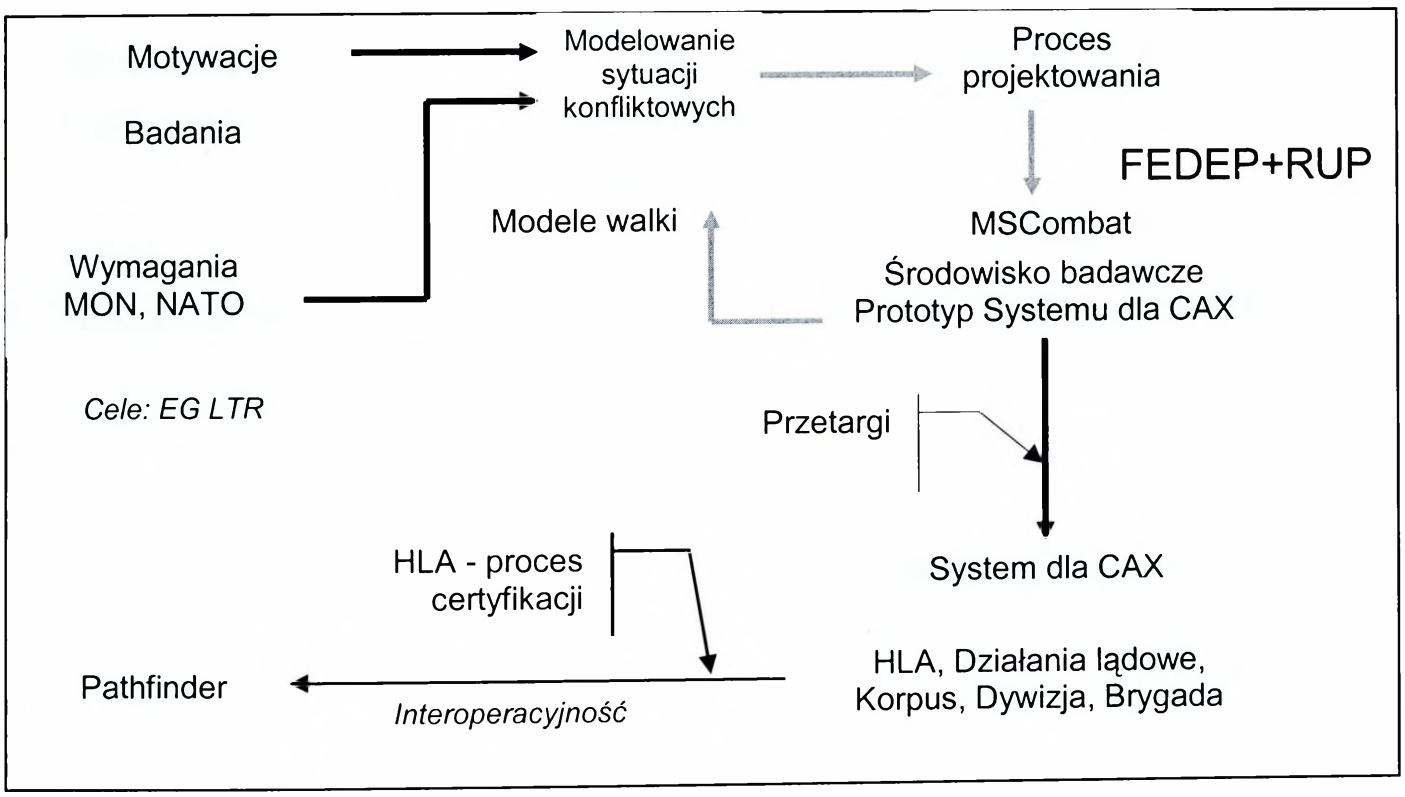
- FOM (ang. *Federation Object Model*) – formalny opis (zgodny z OMT) udostępnianych przez federację danych (obiekty, atrybuty, interakcje, parametry) określony na podstawie modeli SOM wszystkich federatów tej federacji.

Symulacja HLA jest postrzegana jako hierarchia komponentów. Na najniższym poziomie znajduje się albo matematyczny model symulowanego obiektu, albo interfejs niskopoziomowy do rzeczywistego urządzenia. Dane przekazywane z tego poziomu należy opisać w postaci interfejsu dla innych uczestników symulacji. Tak opisany model uczestnika symulacji jest nazywany federatem. Interaktywna symulacja rozproszona może składać się z dowolnej liczby federatów i jest nazywana federacją. W trakcie trwania symulacji poszczególni federaci mogą się dołączać lub odłączać od federacji. Federaci komunikują się

między sobą za pomocą warstwy komunikacyjnej RTI. Zadaniem RTI jest koordynacja oraz kontrola przepływu komunikatów i danych między federatami.

HLA jako najnowszy standard integracji systemów symulacyjnych zapewnia:

- wspólną architekturę umożliwiającą prowadzenie eksperymentów symulacyjnych w syntetycznym środowisku rozproszonym.
- standard integrujący ludzi, sprzęt i systemy informatyczne, oraz przede wszystkim:
  - ÷ interoperacyjność powiązanych systemów,
  - ÷ możliwość wielokrotnego wykorzystania elementów systemów symulacyjnych,
  - ÷ rozdzielenie warstw elementów symulacyjnych i warstwy elementów pomocniczych środowiska.



Rysunek 39 Uwarunkowania projektu systemu symulacyjnego

HLA, jest standardem definiującym zbiór funkcji na potrzeby budowania systemów symulacyjnych. Wykorzystywanie oprogramowania zgodnego z HLA daje następujący korzyści:

- możliwość rozproszonego wykonywania symulacji na wielu komputerach,
- ułatwienie zarządzania czasem symulacji,
- prostą komunikację pomiędzy rozproszonymi elementami symulacji,
- możliwość współpracy z innymi symulacjami zbudowanymi w oparciu o HLA.

Model HLA operuje na kilku podstawowych pojęciach:

- federacja : cała symulacja,
- federat : pojedynczy element symulacji,
- model obiektowy: obiekty[1] i interakcje[2] tworzone i przekazywane przez federatów.

W praktyce na symulację budowaną w oparciu o HLA składa się zbiór federatów podłączających się do federacji (tworzących federację). Federaci tworzą obiekty, zmieniają wartości atrybutów obiektów, rejestrują się na zmiany atrybutów, przesyłają i otrzymują interakcje.

Narzucany jest zbiór reguł HLA, które muszą być spełnione by można było mówić o zgodności ze standardem. Konsekwencją takiego założenia jest przestrzeganie wyspecyfikowanych reguł federacji i federatów.

Reguły federacji:

1. Każda Federacja musi mieć jeden model FOM określony w formacie zgodnym z OMT.
2. Wszystkie instancje obiektów symulacyjnych powinny znajdować się w federacie, a nie w RTI.
3. Podczas eksperymentu wymiana danych określonych w FOM pomiędzy federatami powinna odbywać się za pośrednictwem RTI.
4. Podczas eksperymentu komunikacja pomiędzy federatami a RTI może odbywać się wyłącznie przez funkcje zdefiniowane w interfejsie IS HLA.
5. W każdej chwili eksperymentu atrybuty instancji obiektów należą co najwyżej do jednego federata.

Reguły federatów:

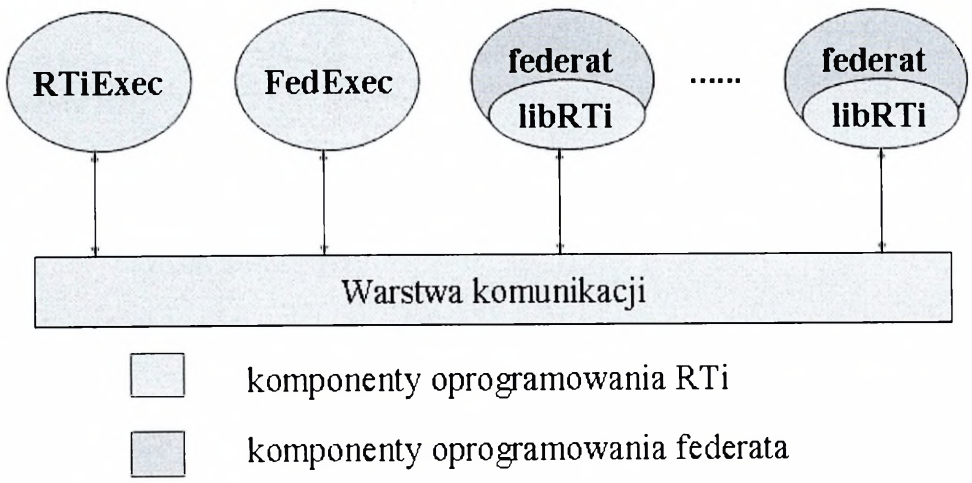
1. Każdy federat powinien mieć model SOM określony w formacie zgodnym z OMT.
2. Federaci powinni potrafić uaktualniać i/lub przekazywać uaktualnienia atrybutów oraz wysyłać i/lub odbierać interakcje zgodnie z ich SOM.
3. Federaci powinni potrafić przekazywać i/lub akceptować „prawo własności" atrybutów dynamicznie podczas eksperymentu zgodnie z ich SOM.

---

[1] Obiektem w sensie HLA jest struktura danych podlegająca znanemu z metodologii obiektowej dziedziczeniu. Istotną różnicą jest fakt, że obiekt HLA nie ma powiązanych ze sobą metod / funkcji. Obiekty HLA mogą istnieć przez dowolnie długi czas trwania symulacji.

[2] Interakcją w sensie HLA jest Obiekt HLA którego czas życia ogranicza się do czasu przesłania informacji, tj. jest on tworzony, wysyłany i niszczony zaraz po odebraniu go przez adresata.

4. Federaci powinni potrafić dokonać zmian wartości progowych warunkujących uaktualnienia atrybutów.
5. Federaci powinni zarządzać czasem lokalnym w sposób zapewniający skoordynowaną wymianę danych.

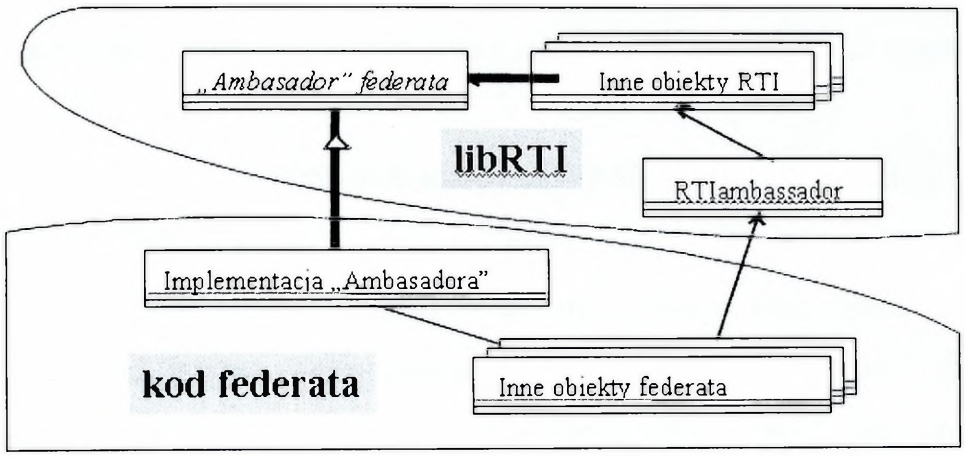


Rysunek 40 Komponenty RTI

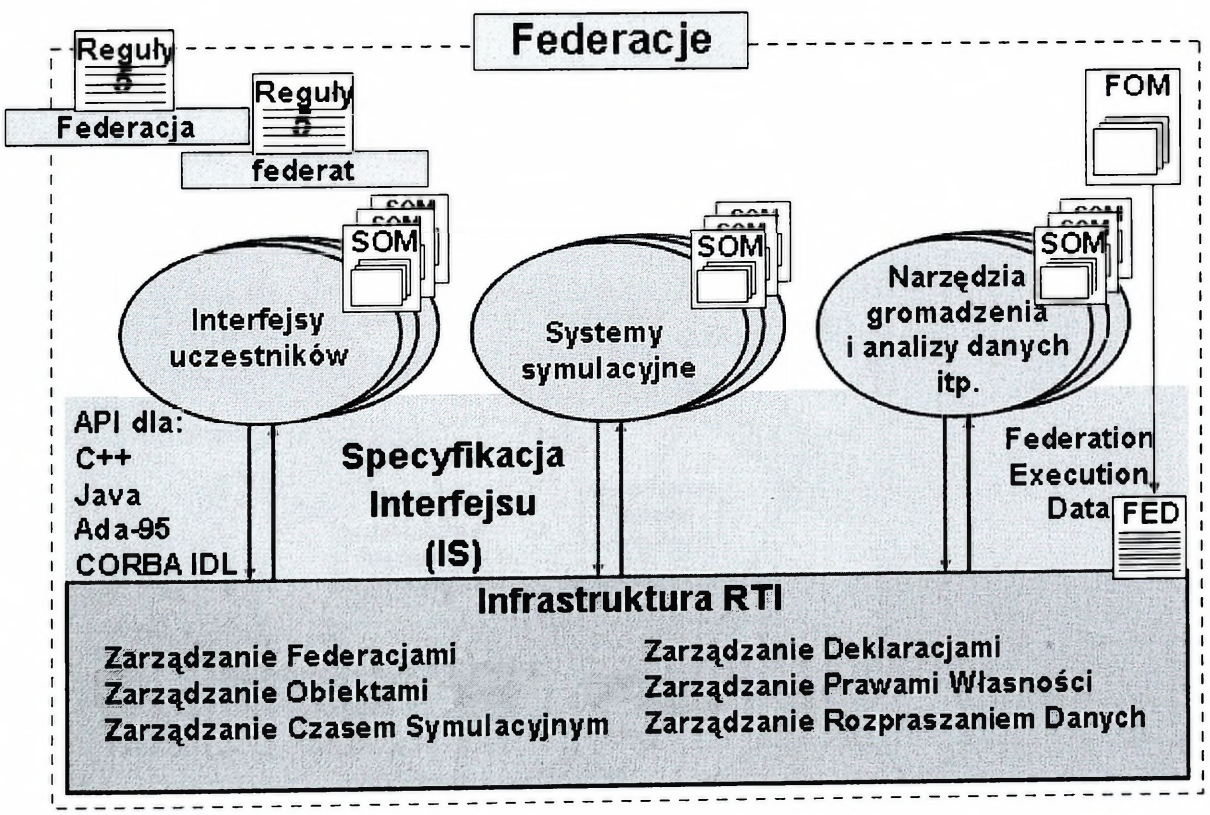
W skład RTI wchodzą następujące komponenty:

- FedExec - Federation Executive:
  - proces jeden dla każdej federacji;
  - powoływany przez pierwszego uruchomionego federata z federacji;
  - zarządza w fazie FE włączaniem oraz wyłączaniem z federacji wielu federatów;
  - przydziela unikalne nazwy federatom;
  - obsługuje wymianę danych między federatami;
- rtiExec - RTI Executive
  - zarządza tworzeniem i usuwaniem wielu FE, tzn. włączaniem do i wyłączaniem z eksperymentu wielu federacji;
  - zapewnia unikalne nazwy procesów FedExec;
  - jest procesem globalnym wykonywanym na jednej platformie sprzętowo-programowej;
  - obsługuje łącze sieciowe na odpowiednich portach;
- libRTI - RTI Library
  - API języków C++, Java, CORBA IDL, Ada;

- umożliwia federatowi dostęp do RTiExec oraz FedExec przez klasę „RTIambasador", a w drugą stronę przez implementację „Ambasadora" federata.



Rysunek 41 Zasada działania libRTI



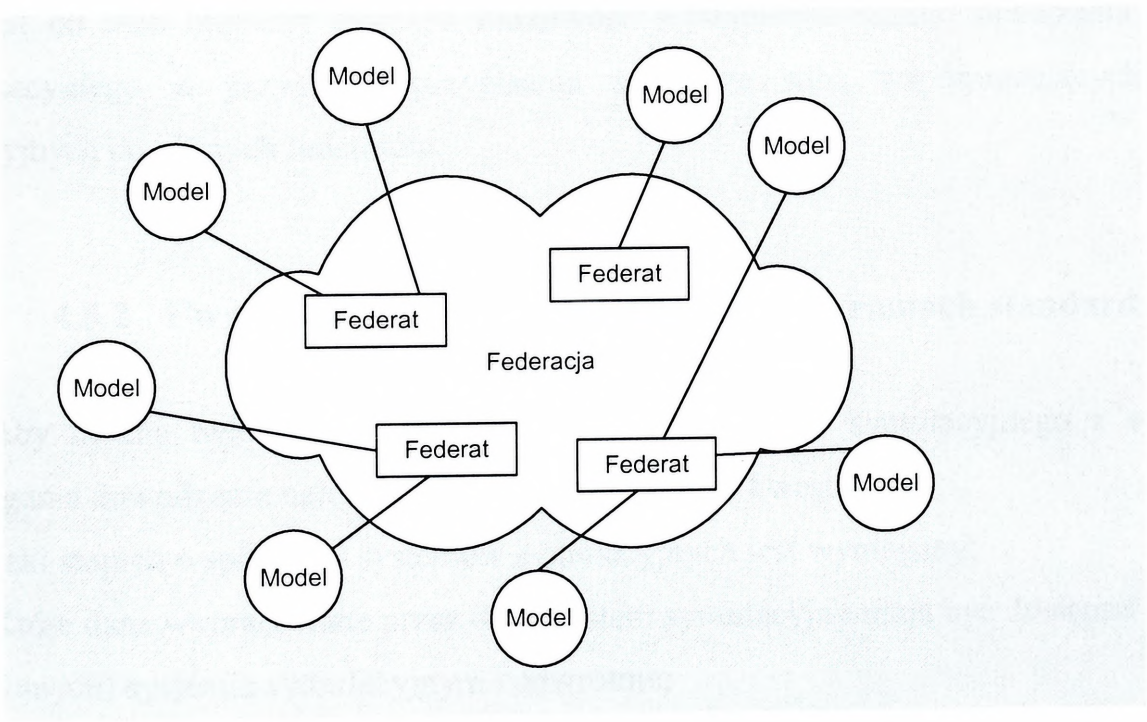
Rysunek 42 Środowisko symulacji oparte na HLA

4.5.1 Infrastruktura programowa systemów w środowisku HLA

Systematyzując oprogramowanie systemów w standardzie HLA można podzielić na następujące bloki:

- federacja – kompletny system składający się z pojedynczych modułów symulacyjnych i modułów dodatkowych (np. zobrazowania) – federatów
- federat – najmniejszy samodzielny element systemu realizujący określoną część funkcjonalności. Istotną cechą federata jest możliwość komunikacji tylko i wyłącznie za pośrednictwem SHLA. Federaci mogą zawierać w sobie jeden lub więcej modeli symulacyjnych.
- model symulacyjny oraz system wspomagania dowodzenia – elementarne moduły oprogramowania zapewniające funkcjonalność elementu symulowanego świata pracujący w ramach federata.

W tak przygotowanej infrastrukturze mogą być osadzane poszczególne modele symulacyjne i systemy wspomagania dowodzenia.



Rysunek 43 Koncepcja modeli w ramach federatów.

Grupowanie modeli w ramach jednego federata zostało podyktowane dwoma powodami:

wydajnościowym – ograniczenie liczby federatów,

logicznym – łączenie modeli operujących na podobnych lub tych samych danych.

Odpowiedzialność modeli ogranicza się do:

- sterowania czasem rzeczywistym i czasem symulacji,
- rejestrowania się na zmiany atrybutów i odbiór interakcji,
- wysyłania zmian atrybutów obiektów oraz wysyłanie interakcji,
- przetwarzania informacji o zmianach atrybutów i o zaistniałych interakcjach,
- właściwą współpracę w infrastrukturą HLA.

W istocie cała „logika" symulatora zawarta jest w modelach. Dodatkowo modele czerpią wiedzę o symulowanym świecie z bazy danych sił i środków. Specjalizowane zadania systemu symulacyjnego takie jak archiwizacja czy też wizualizacja aktualnej sytuacji także realizowane będą przez odpowiednio zaimplementowane specjalizowane modele. Traktowanie całego systemu jak zbioru modeli ma swoje zalety i wady; do zalet możemy zaliczyć:

- podział pracy na ściśle zdefiniowane zadania (modele),
- łatwość testowania,
- łatwość przyszłej rozbudowy systemu.

Natomiast do wad możemy zaliczyć możliwość wystąpienia dużego obciążenia medium komunikacyjnego w przypadku przypisania ściśle ze sobą współpracujących modeli symulacyjnych do różnych federatów.

4.5.2 Uwarunkowania integracji systemów w ramach standardu HLA

Aby można było mówić o pełnej integracji systemu symulacyjnego z systemem wspomagania dowodzenia należy uwzględnić następujące kwestie:

1. Jaki stopień współpracy systemów symulacyjnych jest wymagany;
2. Które dane wypracowane przez dany system symulacyjny mają być dostępne w innym (innych) systemie symulacyjnym i odwrotnie;
3. Jaki jest stopień zgodności danych które mają być wymieniane;
4. W jaki sposób system symulacyjny kontroluje upływ czasu;
5. Czy system symulacyjny wykorzystuje zgodną infrastrukturę HLA;

Powyższe zagadnienia nie wyczerpują oczywiście wszystkich możliwych pytań, które mogą wystąpić, a w następstwie muszą znaleźć odpowiedź, zanim dojdzie do zakończonej sukcesem integracji.

Wymagany stopień interoperacyjności systemu symulacyjnego z systemem wspomagania dowodzenia.

W zależności od obszaru zagadnień, które obejmowane są przez systemy symulacyjne, można mówić o systemach symulacyjnych (samodzielnych, kompleksowych), które nie wymagają współpracy z innymi systemami symulacyjnymi (do nich można zaliczyć system JTLS), a także o systemach symulacyjnych elementarnych (dziedzinowych), które realizują tylko wybrany fragment świata symulowanego i do pełnego wykorzystania wymagają współpracy innych systemami.

Bazując na takim podziale można zidentyfikować dwa typy integracji systemów symulacyjnych:

- integrację informacyjną;
- integrację funkcjonalną.

Integracja informacyjna polega na sprzężeniu informacyjnym systemów symulacyjnych, tj. przekazywaniu pomiędzy nimi określonego zestawu informacji, w celu odwzorowania jednego świata symulowanego. Taka integracja faktycznie pozostawia dwa osobne światy symulowane, których interakcja pomiędzy sobą jest dość ograniczona. Bardzo niewielki, lub żaden jest w takim przypadku wpływ na siebie symulowanych obiektów i modeli nimi zawiadujących.

W przypadku integracji funkcjonalnej można już mówić o współpracy modeli symulacyjnego nad wypracowaniem pojedynczego środowiska symulowanego. Taki typ integracji gotowych już systemów symulacyjnych jest dość trudny i może wymagać wykonania wielu modyfikacji w już funkcjonujących systemach.

Integracja funkcjonalna jest prostsza do uzyskania w początkowych fazach tworzenia systemu symulacyjnego, tj. gdy system symulacyjny tworzony są równolegle i możliwe jest wczesne dostosowanie modeli do współpracy już w fazie projektowej.

Zakres przesyłanych danych pomiędzy systemami symulacyjnymi

To zagadnienie brzmi nieco przewrotnie, ponieważ w ramach infrastruktury HLA każdy uczestnik/model symulacji (federat) ma dostęp do wszystkich danych udostępnianych przez inne modele. Połączone systemy symulacyjne mają zatem dostęp do wszystkich

wypracowanych danych. Sama dostępność danych nie sprawia, że możemy mówić o integracji – istotne jest wykorzystanie udostępnianych danych.
Każdy system, wykonany zgodnie z standardem HLA, musi posiadać zdefiniowany model obiektowy danych. Model ten opisuje zbiór obiektów wraz z ich atrybutami, które mogą pojawiać się w symulowanym środowisku. Model obiektowy jest deklaracją interfejsu, którą implementuje dany system symulacyjny.

Wykorzystanie danych udostępnianych w innym systemie można osiągnąć poprzez dostosowanie modelu obiektowego obu systemów. Dane które mają być wymieniane muszą być zdefiniowane przez identyczny model obiektowy, który może być fragmentem całościowego modelu obiektowego.

Stopień zgodności danych, które mają być wymieniane

Podczas dopasowywania modeli obiektowych należy szczegółowo ocenić zgodność danych. Pod tym pojęciem rozumiany jest zbiór atrybutów danych, który prowadzi do poprawnego przekazania i 'zrozumienia' danych pomiędzy systemem symulacyjnym a wspomagania systemem dowodzenia. Pierwszym elementem jest zgodność typów danych (liczba całkowita, zmiennoprzecinkowa, literał). Drugim elementem jest zgodność charakteru danych (przyjęta jednostka, listy wyliczeniowe).

Ponieważ infrastruktura HLA samodzielnie nie zapewnia szeregowania (*marshalling*u), toteż należy zadbać o jednakową reprezentację maszynową przesyłanych danych. Zagadnieniem tym objęta jest bajtowa wielkość poszczególnych typów danych jak i ich bajtowa kolejność.

Kontrola czasu

W ramach HLA kontrola czasu jest bardzo szczegółowo zdefiniowana. Różne modele upływu czasu skonstruowane są w taki sposób, by mogły być bez problemu ze sobą łączone. Jednak by w łączonych systemach symulacyjnych nie było problemów z upływem czasu, należy pamiętać o:

- zunifikowaniu jednostki czasu (np. sekunda, milisekunda, itp.) - tak by jednostka upływu czasu była tak samo rozumiana w integrowanych systemach;
- zunifikowaniu powiązania czasu z kalendarzem – tak by czas w symulacji oznaczał ten sam moment;

- podjęciu decyzji o sterowaniu powiązaniem czasu symulowanego z czasem rzeczywistym.

Zgodność infrastruktury HLA

Standard HLA rozwiązuje wiele problemów przy konstruowaniu systemu. Niestety standard nie precyzuje sposobów w jaki infrastruktura HLA przekazuje dane pomiędzy poszczególnymi federatami. Konsekwencją tego faktu jest uwarunkowanie interoperacyjności systemów symulacyjnych wspólną wersją infrastruktury. Infrastruktura HLA wyprodukowana przez dwie różne firmy nie będzie mogła porozumieć się ze sobą – zatem należy zapewnić tą samą infrastrukturę HLA. Standard HLA definiuje interfejs programisty (API) do infrastruktury, stąd w większości przypadków wymiana infrastruktury sprowadza się do prostej rekompilacji kodu. Jednak pomiędzy poszczególnymi implementacjami RTI HLA występują subtelne różnice, które sprawiają, że proces ten może być dość długotrwały.

5. Problemy do rozważenia

Niezależnie od przyjętego wariantu rozwiązania umożliwiającego wymianę informacji między systemem symulacyjnym JTLS oraz systemem dowodzenia SZAFRAN, należy wziąć pod uwagę następujące zagadnienia, których pominięcie lub zbagatelizowanie może znacząco wpłynąć na obniżyć wartości walorów użytkowych przyjętego rozwiązania:

1. Identyfikatory jednostek – należy dążyć do sytuacji, w której dane przekazywane pomiędzy dwoma systemami będą etykietowane jednoznacznymi identyfikatorami jednostek ćwiczących. W chwili obecnej ograniczenie dotyczące identyfikatorów jest wymuszane ze względów technologicznych w systemie JTLS.

2. Parametry sprzętu – należy wziąć pod uwagę przy kalibracji systemu JTLS (parametry systemów walki) oraz przy konfiguracji systemu SZAFRAN (dane dla modułu wyznaczania stosunków sił). Nie powinna mieć miejsca sytuacja, w której stosunek sił wyznaczony w systemie dowodzenia SZAFRAN dla poszczególnych wariantów działania znacząco odbiega od wyników symulacji opartej na parametrach dostarczonych do JTLS.

3. Agregacja danych – w systemie JTLS najmniejszą jednostką (poza specjalnie wydzielonymi tzw. HRU) jest batalion. Informacje dotyczące położenia, zdolności bojowej, zobrazowanie sytuacji odbywa się na tym szczeblu. Jest to wynikiem założenia, że podstawową jednostką ćwiczącą jest dywizja.

4. Synchronizacja czasu – system JTLS ma możliwość sterowania szybkością upływu czasu. W systemie SZAFRAN nie istnieje w chwili obecnej taka możliwość, ani nawet pojęcie przyspieszenia czasu.

6. Analiza porównawcza rozwiązań

W niniejszym rozdziale podjęta zostanie próba przedstawienia analizy porównawczej rozwiązań umożliwiających przeprowadzenie integracji systemu symulacyjnego JTLS z systemem dowodzenia SZAFRAN. W porównanie rozwiązań nie uwzględnia propozycji integracji obu systemów na poziomie bazy danych. Odrzucenie tego rozwiązania już na wstępie wydaje się oczywiste ze względu na to, że trudno jest oceniać jego walory w momencie, gdy możliwość jego wdrożenia jest w zasadzie nikła i dotyczyć może ewentualnie tylko fazy przygotowywania ćwiczeń, natomiast nie może być stosowana w fazie jego prowadzenia.

Porównanie pod kątem wybranych kryteriów.

1. Wymiana danych (jednostronna, dwustronna).

	Możliwości	Uwagi
Wiadomości niesformalizowane	jednostronna	
Wiadomości sformalizowane	jednostronna	
Open Access Programs	dwustronna	
Specjalizowany interfejs	dwustronna	
Standard HLA	dwustronna	

2. Możliwość automatycznego przetwarzania danych.

	Możliwości	Uwagi
Wiadomości niesformalizowane	nie	
Wiadomości sformalizowane	tak	
Open Access Programs	tak	
Specjalizowany interfejs	tak	
Standard HLA	tak	

3. Odporność na zmiany w systemie JTLS.

	Możliwości	Uwagi
Wiadomości niesformalizowane	duża	
Wiadomości sformalizowane	duża	Zależna od zmian standardów USMTF i ADatP-3

Open Access Programs	średnia	Zmiany wystąpiły w wersji 3.0. Dotyczą m.in. nazw tabel i pól w tabelach.
Specjalizowany interfejs	niska	
Standard HLA	duża	

4. Otwartość

	Możliwości	Uwagi
Wiadomości niesformalizowane	tak	Protokół SMTP
Wiadomości sformalizowane	tak	Protokół SMTP, USMTF, ADatP-3
Open Access Programs	tak	RDBMS Oracle
Specjalizowany interfejs	nie	
Standard HLA	tak	

5. Sterowanie upływem czasu.

	Możliwości	Uwagi
Wiadomości niesformalizowane	nie	
Wiadomości sformalizowane	nie	
Open Access Programs	nie	
Specjalizowany interfejs	nie	
Standard HLA	tak	

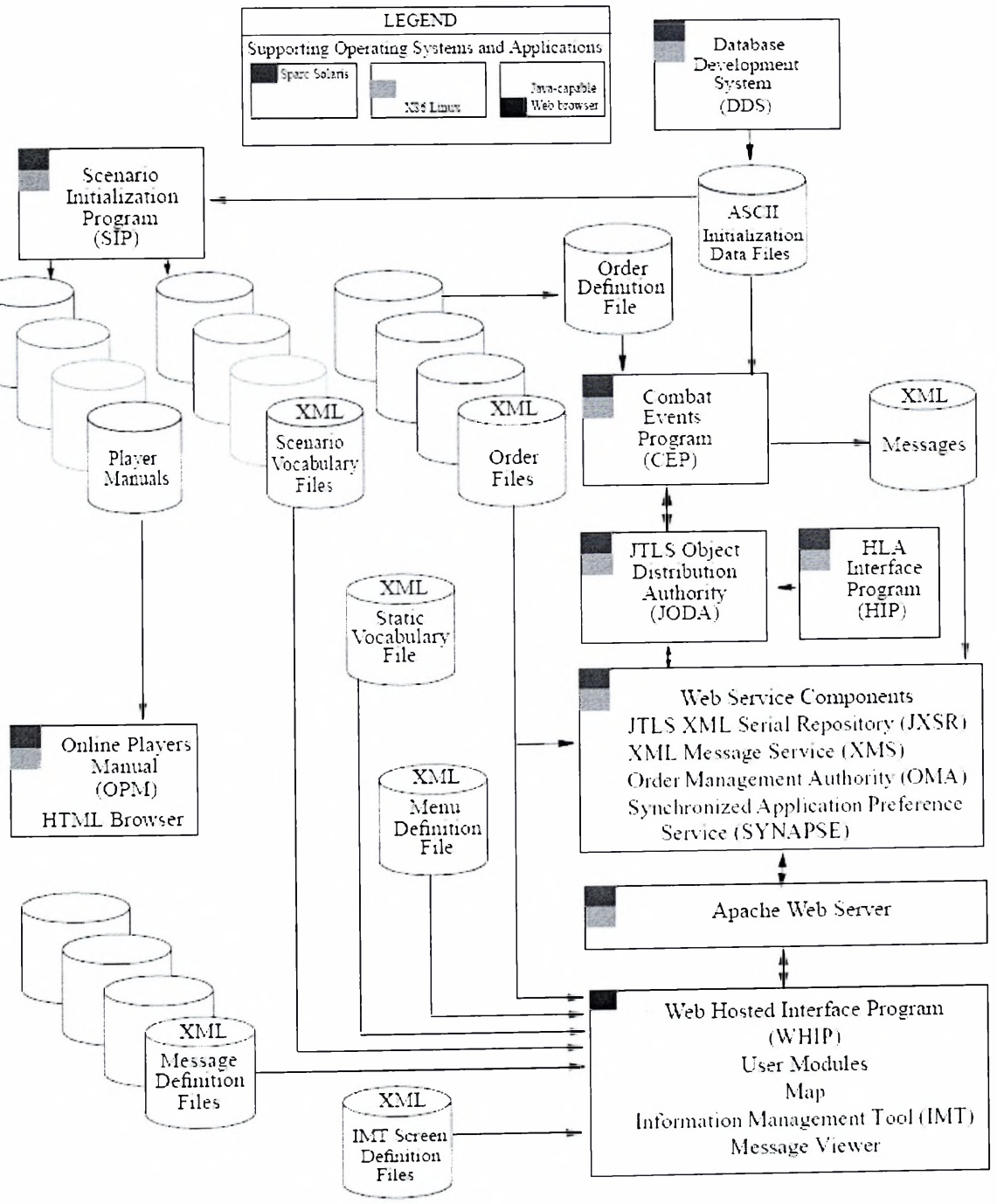
6. Czas potrzebny na wdrożenie.

	Możliwości	Uwagi
Wiadomości niesformalizowane	bardzo krótki	
Wiadomości sformalizowane	krótki	
Open Access Programs	średni	
Specjalizowany interfejs	długi	
Standard HLA	krótki*/długi	* Przy założeniu, że system dowodzenia był budowany z założeniem wykorzystania standardu HLA

7. Kierunki rozwoju systemów

7.1 JTLS

W chwili obecnej główne zmiany w systemie JTLS, jakie wprowadził w niej producent, to zmiany dotyczące interfejsu systemu oraz wykorzystanie technologii otwartych stosowanych szeroko w rozwiązaniach biznesowych. Całość kierunku tych zmian producent już w zapowiedziach ich przeprowadzenia określał jako WEJ (ang. *Web Enabled JTLS*).

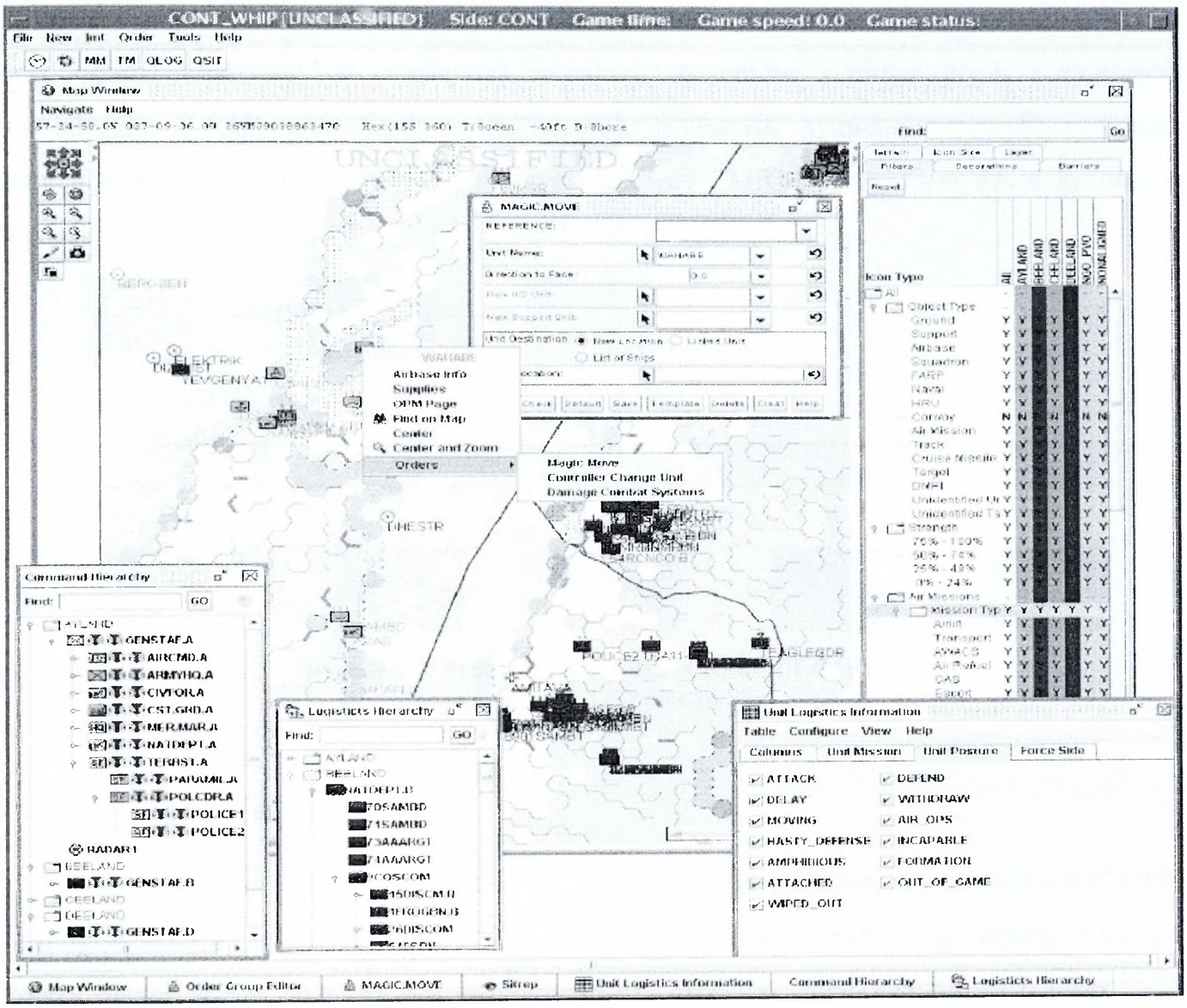


Rysunek 44 Struktura programowa JTLS v.3.0 z wykorzystaniem WEJ
/źródło: JTLS Executive Overview v.3.0.0.0/

Zasadnicze elementy systemu takie jak: narzędzia przygotowywania bazy danych (DDS) oraz program implementujący modele walki (CEP) nie uległy znacznym modyfikacjom. Z punktu widzenia interfejsu HLA zmiany w systemie również nie są istotne. Wewnętrznie natomiast pliki tekstowe ASCII zostały zastąpione plikami w formacie XML, co jest zresztą zbieżne z tendencjami w innych obszarach (np. standard XML-MTF).

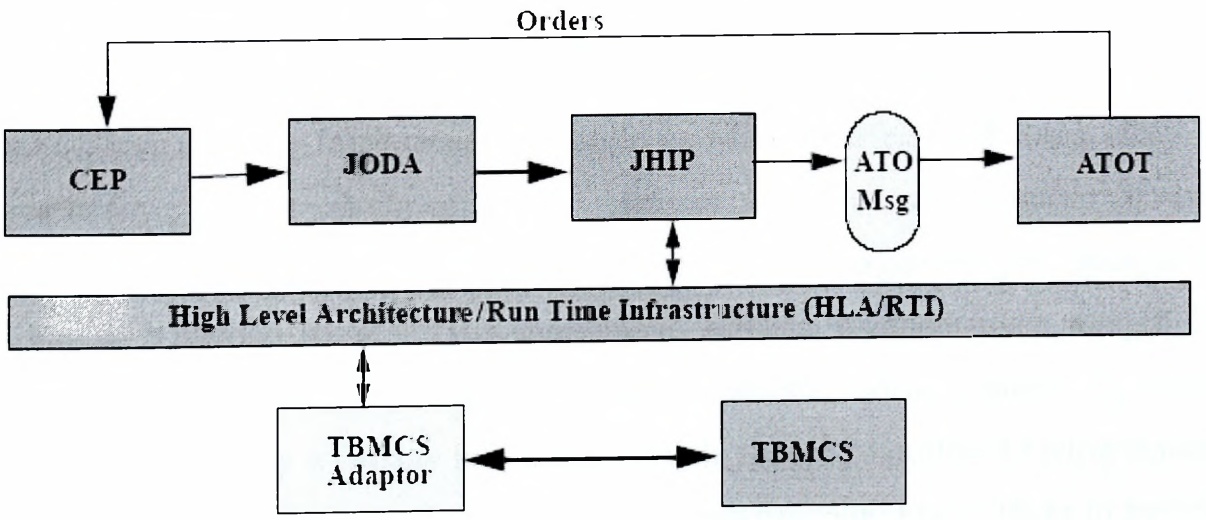
Zmienił się sposób dystrybucji informacji generowanych przez CEP oraz przesyłanych do modelu. W nowej wersji pojawił się moduł JODA (ang. *JTLS Object Distribution Authority*), który docelowo zastąpi GENIS. W wersji 3.0 istnieje możliwość eksploatacji systemu zarówno w nowej, jak i starej konfiguracji. Według informacji producenta systemu kolejne wersje nie będą już wspierały starej konfiguracji z modułem GENIS.

Dużemu przeobrażeniu uległ interfejs użytkownika, jednak najprawdopodobniej będzie bardziej przyjazny dla użytkownika. Nowy moduł, mający zastąpić GIAC, nosi nazwę WHIP (ang. *Web Hosted Interface Program*).



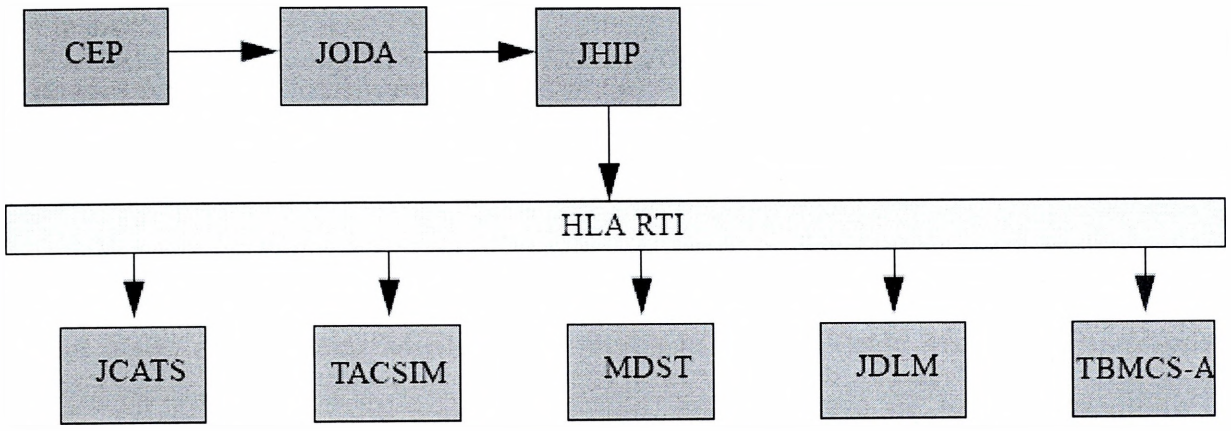
Rysunek 45 Okno modułu WHIP

Patrząc przez pryzmat integracji systemu symulacyjnego JTLS z innymi systemami, zarówno symulacyjnymi jak i systemami dowodzenia, można zauważyć wyraźną tendencję rozwojową, która uwzględnia mechanizmy wymiany wiadomości sformatowanych, tak jak to się ma w przypadku np. systemu TBMCS (ang. *Theater Battle Management Core System*)



Rysunek 46 Integracja systemów JTLS i TBMCS
/źródło: JTLS 3.1.0.0 Design Plan/

oraz ogólną tendencję do ściślejszej integracji systemów symulacyjnych i systemów dowodzenia w celu stworzenia syntetycznych środowisk symulacji z wykorzystaniem standardu HLA w ramach wielomodelowej federacji JMRM (ang. *Joint Multi-Resolution Model*).



Rysunek 47 Praca JTLS w ramach federacji JMRM
/źródło: JTLS 3.1.0.0 Design Plan/

7.2 SZAFRAN

Dalszy rozwój systemu SZAFRAN będzie determinowany między innymi spełnieniem wymogów stawianych przez MIP (ang. *Multilateral Interoperability Programme*), w którym Polska bierze udział jako członek stowarzyszony. MIP jest międzynarodowym programem mającym na celu opracowanie specyfikacji rozwiązań zapewniających osiągnięcie interoperacyjności systemów wspomagania dowodzenia (ang. *Command and Control Information Systems - C2IS*) różnych krajów na szczeblach dowodzenia od korpusu do batalionu w czasie przygotowania i prowadzenia operacji połączonych przez zgrupowania międzynarodowe.

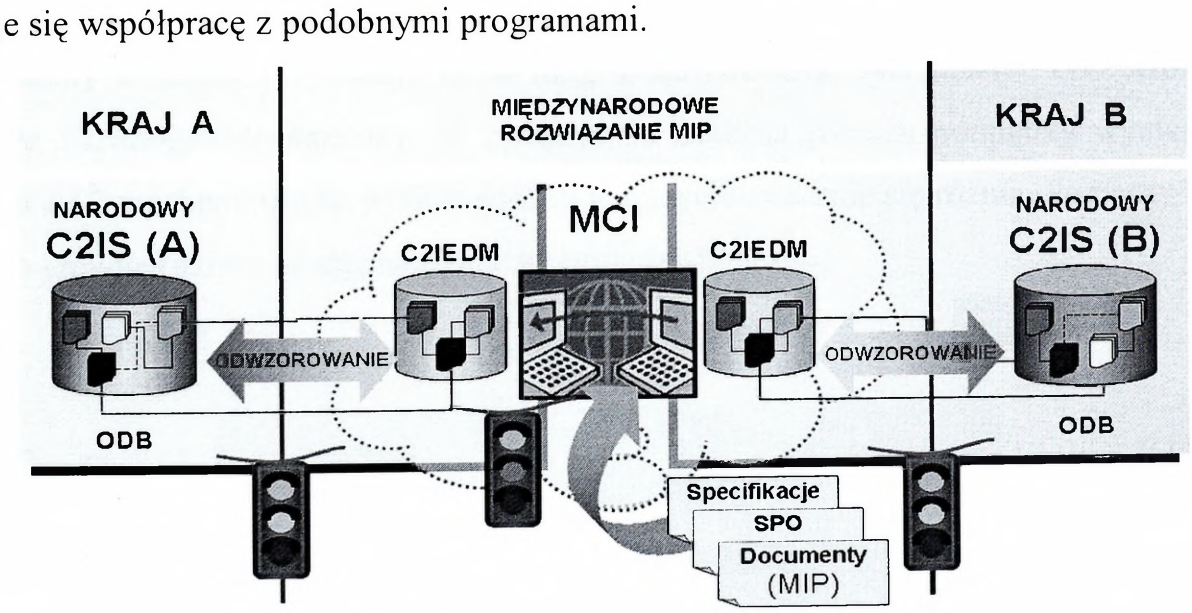
Osiągnięcie interoperacyjności narodowych systemów wspomagania dowodzenia ma być zapewnione poprzez implementację interfejsu MIP. Interfejs MIP (ang. *MCI – MIP Common Interface*) jest rozumiany jako element systemu wspomagania dowodzenia implementowany przez każdy kraj niezależnie, zgodnie z uzgodnionymi specyfikacjami. Składa się on z dwóch zasadniczych elementów: modelu wymienianych danych i zasad wymiany informacji. Model wymienianych danych określa rodzaj i zakres wymienianych danych, natomiast zasady wymiany informacji ustalają sposoby i protokoły ich wymiany. Nie narzuca się sposobu implementacji interfejsu przez poszczególne kraje. Może to być element

wbudowany w narodowy system lub dodatkowe urządzenie zapewniające wymianę informacji zgodnie z uzgodnionymi specyfikacjami.

Narodowe systemy wspomagania dowodzenia państw – uczestników programu MIP powinny wymieniać między sobą:

- sytuację operacyjno-taktyczną (włącznie z możliwościami i stanami sił własnych i przeciwnika);
- plany i rozkazy (dokumenty dowodzenia);
- alarmy NBC i informacje krytyczne.

Główny nacisk położony jest na wymianę istotnych informacji, w każdym rodzaju operacji od walki do Operacji Militarnych innych niż wojna (ang. *Military Operations Other Than War - MOOTW*). W celu osiągnięcia tego, wykorzystuje się w maksymalnym stopniu operacyjne, proceduralne i techniczne wyniki prac uzyskane w poprzednich programach takich jak Battlefield Interoperability Programme, Quadrilateral Interoperability Programme i programie ATCCIS (ang. *Army Tactical Command and Control Information System*) oraz inicjuje się współpracę z podobnymi programami.



Rysunek 48 Ogólna idea interfejsu MIP
/źródło: Piela J., Kołacz M., Nieroda P.: Multilateral Interoperability Programme. Stan aktualny i kierunki rozwoju/

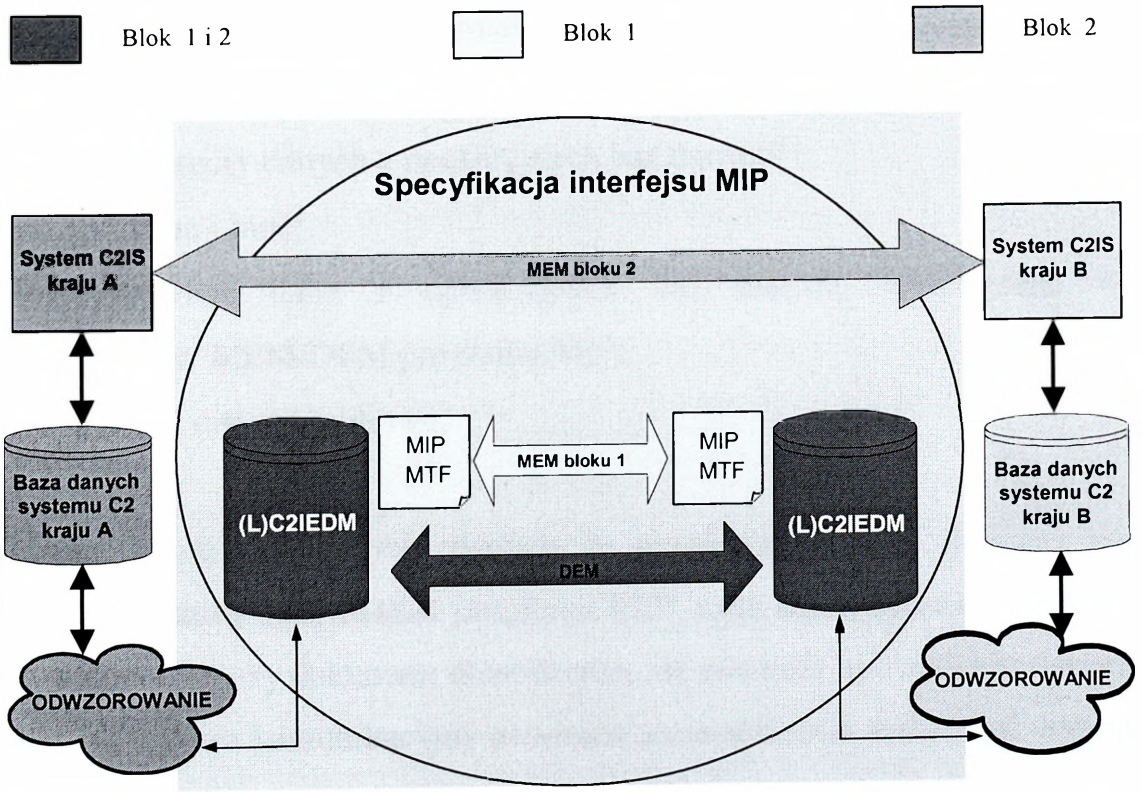
Pierwszą wersją modelu wymiany informacji był GH (ang. *Generic Hub*), który tworzony i rozwijany był w programie ATCCIS do roku 2001. Zawiera on podstawowe i wspólne dla wszystkich obszarów funkcjonalnych elementy pola walki. Do tych wspólnych, ogólnych informacji pola walki należą m. in. obiekty pola walki, ich typy, położenie, status oraz powiązania pomiędzy nimi. Na bazie modelu GH można dowolnie rozwinąć inny model z zachowaniem kompatybilności z GH.

Dalszym etapem w rozwoju modelu danych wymiany informacji było stworzenie w fazie 1 programu MIP modelu LC2IEDM, który jest rozszerzeniem funkcjonalnym modelu GH o elementy pola walki niezbędne dla prowadzenia militarnych operacji lądowych. Rozszerzono go na podstawie wymagań informacyjnych dla operacji lądowych, które wywodzą się głównie z wiadomości ADatP-3.

W bloku drugim postawiono nowe wymagania operacyjne, zawierające m.in. wsparcie operacji pokojowych, antykryzysowych oraz innych operacji z użyciem wojska. Wymagania te odzwierciedliły się także w modelu danych C2IEDM, który został poszerzony na podstawie LC2IEDM, aby zapewnić wsparcie w/w wymienionych wymagań.

Program MIP postawił także wymaganie wsparcia operacji połączonych w rozwiązaniu MIP rezultatem, czego są także zmiany w modelu danych. Rezultatem tego wymagania jest m.in. tworzenie modelu danych JC3IEDM, który aktualnie jest w fazie projektowania. Etapem przejściowym, na czas trwania etapu 2 MIP, jest model C2IEDM, który zawiera szereg elementów potrzebnych podczas operacji połączonych (określanie przynależności, porty, okręty i inne).

Zarówno w bloku pierwszym jak i drugim program MIP zdefiniował i stosuje dwa mechanizmy wymiany informacji: DEM (ang. *Data Exchange Mechanism*) oraz MEM (ang. *Message Exchange Mechanism*). W związku ze znaczną różnicą pomiędzy wymaganiami w bloku 1 i 2 mechanizmy te, a także sposób ich użycia znacznie się różnią. Poniższy rysunek ilustruje główne różnice w mechanizmach pomiędzy blokami.



Rysunek 49 Porównanie zakresu specyfikacji mechanizmów MEM i DEM w bloku 1 i 2

/źródło: Piela J., Kołacz M., Nieroda P.: Multilateral Interoperability Programme. Stan aktualny i kierunki rozwoju/

Mechanizm MEM w bloku pierwszym miał podobne zastosowanie jak mechanizm DEM, służył do wymiany danych pomiędzy dwoma bazami danych. Zdecydowano więc, że tak dublowana funkcjonalność jest niepotrzebna, poza tym w bloku drugim postawiono nowe wymagania przesyłania informacji niesformalizowanych pomiędzy dwoma rozwiązaniami MIP. W bloku drugim mechanizm MEM stał się mechanizmem przesyłania poczty elektronicznej, która zawiera załączniki binarne zdefiniowane przez MIP. Do załączników tych zalicza się: zdjęcia, dokumenty programu biurowego, wiadomości ADatP-3 i inne.

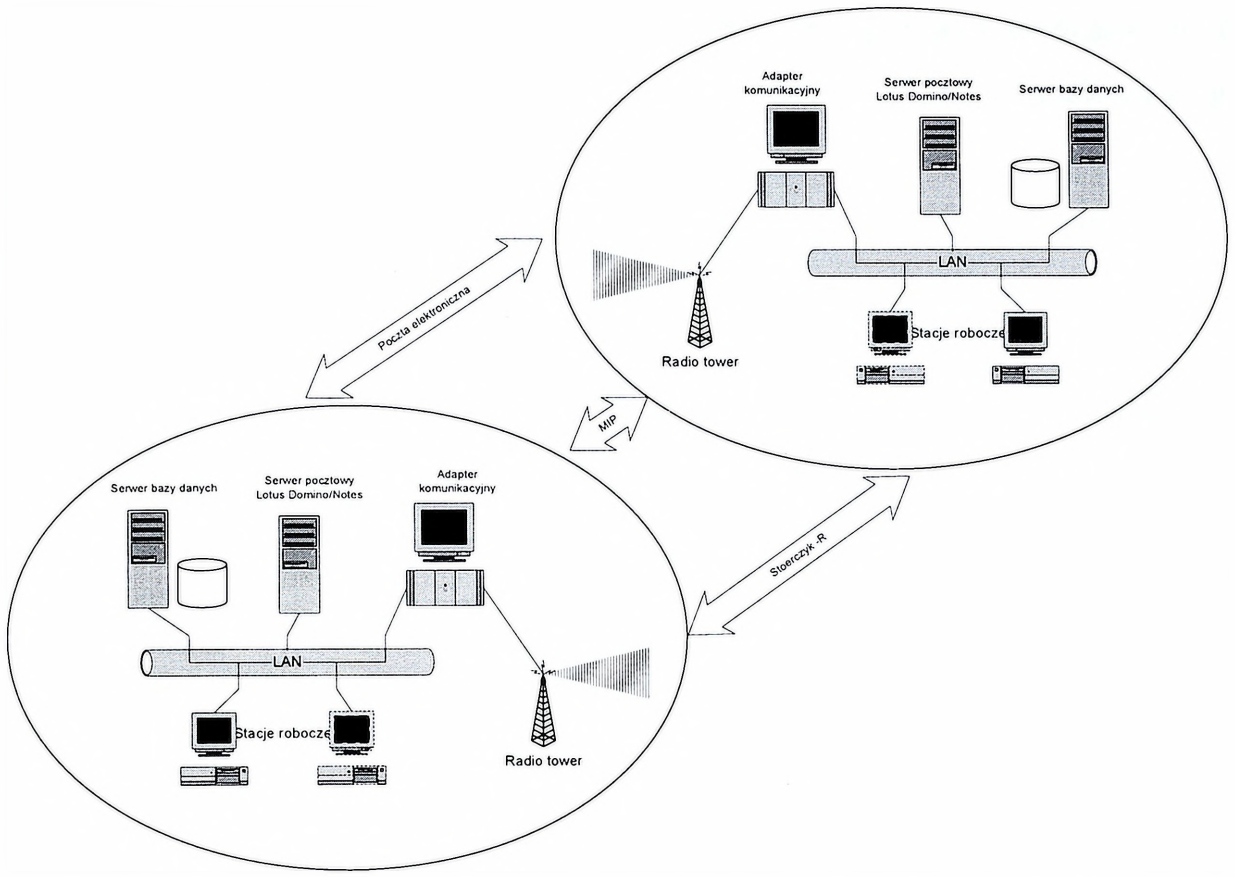
Mechanizm DEM bloku 2 funkcjonalnie nie różni się od DEM bloku 1. Istnieją natomiast różnice w technicznym rozwiązaniu obu wersji.

Obecnie autorzy systemu SZAFRAN podjęli decyzję, że niezbędnym jest podjęcie działań projektowych, które powinny doprowadzić do jego integracji z innymi systemami projektowanymi i eksploatowanymi w SZ RP. W tym celu należy opracować rozwiązanie, przez które, należy rozumieć struktury powiązanych serwerów PWD i adapterów komunikacyjnych, zlokalizowanych w węzłach informatycznych stanowisk dowodzenia, zapewniające wymianę informacji między różnymi systemami dowodzenia i kierowania.

Wymieniane informacje powinny obejmować:

- dokumenty dowodzenia;
- wiadomości sformatowane i sformalizowane;
- wiadomości niesformalizowane;
- selektywne zrzuty danych z operacyjnych baz danych;
- dane graficzne i inne;
- zakontraktowane dane z operacyjnych baz danych, wymieniane w ramach mechanizmów MEM/DEM programu MIP;
- wiadomości i depesze radiowe.

Wymiana zakontraktowanych danych z operacyjnych baz danych, realizowana w ramach mechanizmów MEM/DEM programu MIP, czyli dane, które nie stanowią pewnej całości, jak wiadomość czy dokument dowodzenia, nie powinny być realizowane przez PWD. Innymi słowy to adapter komunikacyjny powinien autonomicznie realizować wymianę takich danych.



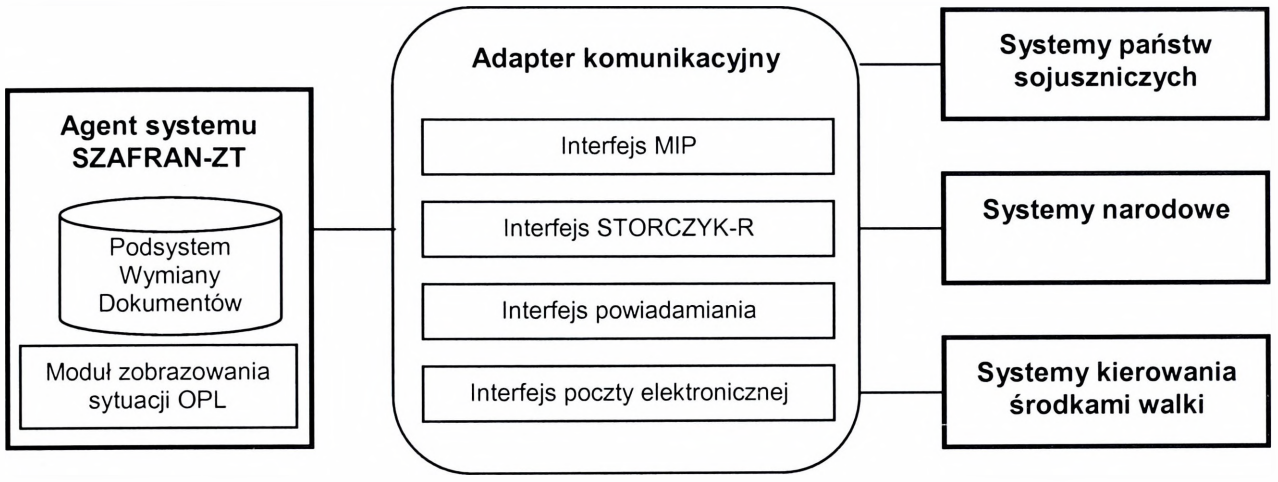
Rysunek 50 Schemat ideowy wymiany danych między ogniwami systemów
/źródło: Bryła D., Sass S.: Integracja ZSyD Kolorado i SZAFRAN ZT na poziomie dokumentów i wiadomości/

W związku z tym opracowana została koncepcja modułu integrującego systemu SZAFRAN z innymi systemami. Całość wymiany informacji w systemie SZAFRAN odbywa się za pośrednictwem serwera Podsystemu Wymiany Dokumentów (PWD). Założenie to jest istotne m.in. z punktu widzenia rejestracji obiegu informacji w systemie. W przypadku przesyłania komunikatów alarmowych zakłada się możliwość ich przesłania bezpośrednio pomiędzy odpowiednią aplikacją a adapterem komunikacyjnym.

Podstawowym trybem pracy serwera PWD jest współpraca z PWD innych stanowisk dowodzenia. W przypadku stwierdzenia, że abonent, do którego adresowana jest informacja nie jest dostępny w sieci dokonywana jest translacja informacji do postaci pliku tekstowego, kopertowana i przesyłana do adaptera komunikacyjnego. Oprogramowanie adaptera komunikacyjnego, na podstawie bazy danych adresowych „rozpoznaje" rodzaj abonenta i protokół transmisji. Dokonuje „obudowania" informacji w sposób właściwy dla odpowiedniego protokołu komunikacyjnego oraz przesłania jej przy pomocy odpowiednich pakietów programowych:

- serwera LOTUS NOTES lub innego serwera pocztowego i standardowych protokołów poczty elektronicznej;

- interfejsu MIP wykorzystującego protokół ESMTP i protokół wymiany informacji zdefiniowany w programie MIP, dla zapewnienia wymiany z systemami państw NATO;
- dedykowany interfejs zapewniające wykorzystanie Systemu Wymiany Danych STORCZYK-R.



Rysunek 51 Model integracji systemu SZAFRAN z innymi systemami
/źródło: Grochalski A., Owczarski D.: Integracja zautomatyzowanych systemów kierowania środkami walki i systemów zarządzania z zautomatyzowanymi systemami dowodzenia/

7.3 Wnioski

Na podstawie analizy dostępnych materiałów, dokumentacji oraz prac związanych z testowaniem i użytkowaniem zainstalowanych w CSiKGW systemów JTLS i SZAFRAN można wysnuć wnioski co do ich architektury, technologii, funkcjonalności oraz kierunków i trendów w rozwoju obu systemów. Każdy z nich jest systemem innej klasy – jeden systemem symulacyjnym, drugi systemem wspomagania dowodzenia, każdy z nich jest również rozwijany w innych uwarunkowaniach przez różnych producentów, które determinują ich rozwój – pierwszy jest już produktem okrzepłym, którego kolejne wersje są wynikami wprowadzenia zmian na podstawie zdobytych doświadczeń. Drugi jest systemem, który jeszcze nie tą drogę ma przed sobą. Ponadto system JTLS powstaje w USA, natomiast system SZAFRAN w Polsce, co skutkuje zupełnie innymi uwarunkowaniami typu: nakłady, standardy jakościowe, świadomość docelowych użytkowników systemu, stopień nasycenia armii nowoczesnymi technologiami oraz wiele innych czynników, w tym również wynikających z już wymienionych.

Wszystko to wpływa zauważalnie na sposób podejścia do systemów. W przypadku systemu JTLS widoczna jest tendencja do coraz ściślejszej integracji zarówno tego systemu symulacyjnego, jak i innych z systemami dowodzenia w celu tworzenia środowisk

syntetycznych pozwalających na jak najbardziej efektywne wykorzystanie możliwości każdego z ich elementów oraz, co najważniejsze, uzyskania jak najlepszego efektu synergii po połączeniu wszystkich elementów składowych.

W przypadku systemu SZAFRAN w chwili obecnej prace są ukierunkowane głównie na osiągnięcie interoperacyjności z systemami dowodzenia innych krajów członkowskich NATO oraz własnymi. Kwestie związane z integracją z systemami symulacyjnymi nie są głównym przedmiotem rozważań, możliwe, że nie są jeszcze brane w ogóle pod uwagę. Trudno jest również zauważyć, aby tworzył się jakiś ośrodek będący w stanie merytorycznie wyznaczyć kierunki integracji oraz posiadający możliwość spowodowania ich uwzględnienia w pracach nad systemami.

Zakończenie

Przedstawione opracowanie pt. **Analiza możliwości integracji systemu wspomagania dowodzenia SZAFRAN z systemem symulacyjnym JTLS** jest próbą identyfikacji wspólnych cech symulacyjnego systemu walki oraz systemu wspomagania dowodzenia, zrealizowanych na potrzeby szkoleniowe różnych armii i poszukiwania rozwiązań umożliwiających integrację lub automatyzację wymiany danych pomiędzy tego typu systemami.

Symulacyjny system walki JTLS, będący na wyposażeniu CSiKGW, jest produktem firmy amerykańskiej. Aktualnie system JTLS jest integrowany z systemem JCATS. W zamyśle planistów stanowić będą platformę realizacji szkolenia wszystkich szczebli dowodzenia w armii USA. Natomiast Polowy Zautomatyzowany System Dowodzenia Związkiem Taktycznym (PZSD ZT) SZAFRAN jest produktem polskim, który aktualnie znajduje się na etapie badań testowych i jest przeznaczony do informatycznego wspomagania procesów dowodzenia wojsk lądowych na szczeblach korpusu, dywizji, brygady i batalionu.

Analiza funkcjonalności oraz architektury technologicznej tych systemów uwidoczniła pewną „wspólną część", zarówno w aspekcie operacyjnym, jak i informatycznym. Ujęcie formalne w postaci modeli jest w zasadzie na podobnym poziomie wiedzy ogólnoteoretycznej w tej dziedzinie. Chociaż o stopniu ich adekwatności autorzy nie mają podstaw wypowiadać się.

Przeprowadzona analiza możliwości integracji obu systemów, której syntezę autorzy ujęli w postaci prezentowanego opracowania, będzie przez nich pogłębiana, rozwijana i konceptualizowana w przyszłości. Problem integracji symulacyjnych systemów walki z systemami wspomagania dowodzenia jest interesujący poznawczo, znaczący teoretycznie, inspirujący i niosący nowe wyzwania w dziedzinie nowoczesnych technologii informatycznych. Sprawcą odrodzenia się zainteresowania tego typu systemami jest utrwalony już pogląd na efektywność i ekonomiczność prowadzenia ćwiczeń dowódczo – sztabowych w formie ćwiczeń wspomaganych komputerowo. Po raz kolejny potrzeby utylitarne wpłynęły na rozwój teorii i technologii. Autorzy pracy wyrażają przekonanie, że efekty pracy badawczej będą wykorzystywane w procesie prowadzenia ćwiczeń dowódczo-sztabowych prowadzonych w Centrum Symulacji i Komputerowych Gier wojennych oraz w działalności szkoleniowej dowództw i sztabów prowadzących i biorących udział w ćwiczeniach wspomaganych komputerowo.

Literatura

1. Aggregated Combat Models, Naval Postgraduate School Operations Research Department, Monterey, 2000
2. Bieniek S.: Narzędzia wsparcia fazy analizy i oceny ćwiczeń CAX w systemie symulacji walki JTLS, AON, Warszawa, 2004
3. Bryła D., Sass S.: Integracja ZSyD Kolorado i SZAFRAN ZT na poziomie dokumentów i wiadomości, CIiŁ ON, 2005
4. GIAC Overview Release 1.7, Los Alamos National Laboratory, Los Alamos, 1997
5. GIAC Player's Guide Release 1.9, Los Alamos National Laboratory, Los Alamos, 1999
6. GIAC Model Controller's Guide Release 1.9, Los Alamos National Laboratory, Los Alamos, 1999
7. Grochalski A., Owczarski D.: Integracja zautomatyzowanych systemów kierowania środkami walki i systemów zarządzania z zautomatyzowanymi systemami dowodzenia, CIiŁ ON, 2005
8. IRIS Definition System (IRIS/DEF), Systematic Software Engineering A/S, Århus, 2004
9. IRIS Information Mapping Tool, Systematic Software Engineering A/S, Århus, 2004
10. JTLS 3.1.0.0 Design Plan, U.S. JFCOM Joint Warfighting Center, Suffolk, 2006
11. JTLS Executive Overview v.2.7, U.S. JFCOM Joint Warfighting Center, Suffolk, 2004
12. JTLS Executive Overview v.3.0.0.0, U.S. JFCOM Joint Warfighting Center, Suffolk, 2005
13. JTLS Technical Coordinator's Guide v.2.7, U.S. JFCOM Joint Warfighting Center, Suffolk, 2004
14. JTLS Technical Coordinator's Guide v.3.0.0.0, U.S. JFCOM Joint Warfighting Center, Suffolk, 2005
15. JTLS Software Maintenance Manual v.2.7, U.S. JFCOM Joint Warfighting Center, Suffolk, 2004
16. JTLS Software Maintenance Manual v.3.0.0.0, U.S. JFCOM Joint Warfighting Center, Suffolk, 2005
17. Kaliński K., Pietrzak G., Boryn P.: Symulacyjny system walki JTLS, AON, Warszawa, 2003
18. Piela J.: Polowy Zautomatyzowany System Dowodzenia Związkiem Taktycznym. Stan obecny i perspektywy rozwoju, PIT, Warszawa, 2004

19. Piela J., Kołacz M., Nieroda P.: Multilateral Interoperability Programme. Stan aktualny i kierunki rozwoju, PIT, Warszawa, 2004
20. Polish Army Tactical C2I System „SZAFRAN", PIT, 2005
21. Kaliński K., Boryn P., Bieniek S., Pietrzak G.: Opracowanie eksperymentu symulacyjnego – realizacja epizodu taktycznego, AON, 2004

BIBLIOTEKA GŁÓWNA - ARCHIWUM
Akademii Obrony Narodowej
Nr ewid. 39581
II

S/6429
9,50