

České vysoké učení technické v Praze
Fakulta elektrotechnická

katedra počítačové grafiky a interakce

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: **Bc. Andreas Stefanidis**

Studijní program: Otevřená informatika (magisterský)
Obor: Počítačová grafika

Název tématu: **Simulace systému EFIS letounu Boeing 787**

Pokyny pro vypracování:


Analyzujte funkci Electronic Flight Instrument Systemu (EFIS) letadla Boeing 787-8 a implementujte jeho věrnou simulaci v maximálním možném rozsahu. Proveďte rešerši vhodných platforem pro poskytnutí fyzikálního modelu letadla a simulaci vnějšího prostředí. Pro zvolenou platformu realizujte rozhraní s vaším simulátorem, které bude umožňovat vzájemnou komunikaci mezi oběma programy. Uvažujte možnosti budoucího rozšíření vašeho simulátoru a to jak po softwarové tak po hardwarové stránce. Implementované řešení ověřte z hlediska funkčnosti a proveďte srovnání s reálným systémem.

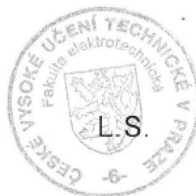
Seznam odborné literatury:


Dodá vedoucí práce

Vedoucí: Ing. Miroslav Macík

Platnost zadání: do konce zimního semestru 2012/2013


prof. Ing. Jiří Žára, CSc.
vedoucí katedry




prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 14. 11. 2011

České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra počítačové grafiky a interakce



Diplomová práce

Simulace systému EFIS letounu Boeing 787

Bc. Andreas Stefanidis

Vedoucí práce: Ing. Miroslav Macík

Studijní program: Otevřená informatika, strukturovaný, navazující magisterský

Obor: Počítačová grafika a interakce

1. ledna 2012

Poděkování

Na tomto místě bych rád poděkoval všem těm, kteří mě během mých studií podporovali a díky kterým mohla tato práce vzniknout.

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

Nemám závažný důvod proti užití tohoto díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne 1. 1. 2012

.....

Abstract

This work should be used to simulate Electronic Flight Instrument System (EFIS) of Boeing 787 aircraft. The solution will simulate various cockpit displays and will also implement the control panels to allow interaction with the EFIS. It will be also implemented an interface between EFIS and the platform which will provide simulation of the external environment and the physical model. The system will be designated for semi-professional use. In future, in the case of successful certification, it could be used for real flight crew training.

Abstrakt

Tato práce by měla posloužit k simulaci systému EFIS (Electronic Flight Instrument System) letounu Boeing 787. Řešení bude simulovat jednotlivé displeje kokpitu a rovněž bude implementovat ovládací panely pro interakci s EFISem. Bude také realizováno rozhraní mezi EFISem a platformou pro poskytnutí simulace vnějšího prostředí a fyzikálního modelu. Systém bude určen pro poloprofesionální využití, v budoucnu by pak po případné certifikaci mohl být využit i jako postupový simulátor pro výcvik posádek.

Obsah

1	Úvod	1
1.1	Motivace.....	1
1.2	Cíle.....	1
2	Analýza	3
2.1	Boeing 787.....	3
2.2	Simulované systémy	4
2.2.1	Primary Flight Display (PFD)	6
2.2.2	Auxiliary Display (AUX).....	8
2.2.3	Navigation Display (ND).....	8
2.2.4	Electronic checklist (CHKL).....	9
2.2.5	Synoptics Display (SYS).....	10
2.2.6	Control Display Unit (CDU)	11
2.2.7	Communication Display (COMM).....	12
2.2.8	Engine Indicating and Crew Alerting System (EICAS)	12
2.2.9	Electronic Flight Bag (EFB)	13
2.2.10	Head-Up Display (HUD)	14
2.2.11	Integrated Standby Flight Display (ISFD)	14
2.2.12	Ovládací panely.....	15
2.3	Druhy simulátorů	15
2.3.1	Kategorie simulátorů.....	16
2.3.2	Certifikace.....	18
2.3.3	Cílová kategorie	19
2.4	Existující platformy.....	19
2.4.1	MS FS 2004.....	19
2.4.2	MS ESP/FSX.....	20
2.4.3	Prepar3D.....	20
2.4.4	XPlane 9	21
2.4.5	FlightGear.....	21
2.4.6	Výběr platformy	22
2.5	Existující řešení.....	23
2.5.1	Abacus 787.....	23
2.5.2	Project Magenta.....	23
2.5.3	Thales 787.....	24

2.5.4	Posouzení existujících řešení.....	24
2.6	Volba vývojové platformy.....	25
3	Návrh.....	27
3.1	Funkční požadavky.....	27
3.2	Nefunkční požadavky.....	27
3.3	Využití knihoven	27
3.3.1	Knihovny třetích stran	27
3.3.2	WinAPI.....	28
3.4	Struktura programu.....	28
3.4.1	Samostatný program / Modul	28
3.4.2	Robustní program / modulární (DLL).....	28
3.4.3	Vlákna.....	29
3.4.4	Řešení připojení více monitorů	30
3.4.5	Klient-server architektura.....	33
3.5	Možnosti rozhraní s FSX	34
3.5.1	SimConnect	34
3.5.2	FSUIPC / WideFS	35
3.5.3	Dekompilace datových souborů	36
3.5.4	AIRAC.....	38
3.5.5	Výběr řešení	39
3.6	Návrh FMC.....	40
3.6.1	Návrh navigační databáze	40
3.7	Návrh uživatelského rozhraní	41
3.8	Návrh rozhraní mezi EFISem a Flight Simulátorem X	42
3.8.1	TCAS.....	42
3.8.2	Meteorologický radar.....	43
3.8.3	Vertical situation display.....	43
3.8.4	Terénní radar	44
3.8.5	BGL dekompilace	44
4	Implementace	45
4.1	Struktura programu.....	45
4.1.1	Třída Core.....	45
4.1.2	Třída Aircraft.....	45
4.2	Antialiasing	46
4.3	Písma	47

4.4	Implementace připojení více monitorů	48
4.4.1	Vykreslovací smyčka	48
4.4.2	Zasílání zpráv	50
4.4.3	Thread Local Storage	51
4.5	Implementace modulů	51
5	Testování	53
5.1	Profiling	53
5.2	Unit testing	54
5.3	Testování uživatelského rozhraní.....	54
5.3.1	Testování s uživatelem A	56
5.3.2	Testování s uživatelem B	58
5.3.3	Zhodnocení testování s uživateli.....	59
6	Závěr.....	63
6.1	Výsledky implementace	63
6.2	Splnění požadavků.....	63
6.3	Budoucí rozvoj a uplatnění.....	64
6.4	Porovnání se skutečným systémem.....	65
	Příloha A – Výsledky profilingu	73
	Příloha B – Class diagram.....	75
	Příloha C – Instalační příručka.....	77
	C.1 Instalace na 1 PC	77
	C.2 Instalace na 2 PC	77
	Příloha D – Uživatelská příručka.....	79
	D.1 Spuštění	79
	D.2 Konfigurace	79
	D.3 Uživatelské rozhraní	81
	D.4 Záznamy letu.....	81
	Příloha E – Stavové proměnné	83
	Příloha F – Struktura navigační databáze	85
	Příloha G – Seznam použitých zkratk.....	87
	Příloha H – Obsah přiloženého DVD.....	89

Seznam obrázků

Obrázek 2.1: Struktura materiálů v draku letadla [1]	3
Obrázek 2.2: EFIS Boeingu 787-8 [7]	5
Obrázek 2.3: Dělení EFISu na jednotlivé displeje [7]	6
Obrázek 2.4: Primary Flight Display	7
Obrázek 2.5: Auxiliary Display	8
Obrázek 2.6: Navigation Display	9
Obrázek 2.7: Electronic Checklist [8]	10
Obrázek 2.8: Synoptics Display	11
Obrázek 2.9: Control Display Unit	11
Obrázek 2.10: Communication Display [8]	12
Obrázek 2.11: Dělení EICAS displeje [8]	13
Obrázek 2.12: EICAS displej	13
Obrázek 2.13: Electronic Flight Bag [8]	14
Obrázek 2.14: Head-Up Display [8]	14
Obrázek 2.15: Integrated Standby Flight Display [8]	15
Obrázek 2.16: Multifunction Keypad	15
Obrázek 2.17: Simulátor Boeingu 737NG v ČR [10]	16
Obrázek 2.18: Simulátor ČSA - A320- úrovně FTD Level 1 [17]	17
Obrázek 2.19: ČSA - ATR42/72 – úrovně FTD Level 2/FNPT II MCC [17]	18
Obrázek 2.20: Simulátor ČSA – A320 – Level D [17]	18
Obrázek 2.21: Microsoft Flight Simulator 2004 [19]	20
Obrázek 2.22: Microsoft Flight Simulátor X [19]	20
Obrázek 2.23: XPlane 9 [19]	21
Obrázek 2.24: FlightGear [19]	22
Obrázek 2.25: Abacus Boeing 787 [24]	23
Obrázek 2.26: Project Magenta Boeing 737NG [25]	24
Obrázek 2.27: Thales Boeing 787 simulátor [26]	24
Obrázek 3.1: Struktura programu	29
Obrázek 3.2: Struktura vláken	30
Obrázek 3.3: Virtuální plocha [29]	31
Obrázek 3.4 Jednoduchá vykreslovací smyčka	32
Obrázek 3.5 Postupné vykreslování v jedné smyčce	32
Obrázek 3.6: Vykreslování pomocí více vláken	33
Obrázek 3.7: Architektura klient-server	34
Obrázek 3.8: Quad Mesh Identifier (Červeně QMID úrovně 4, modře QMID úrovně 7) [34]	37
Obrázek 3.9 Mapa letiště získaná z BGL souborů	37
Obrázek 3.10 Zobrazení navigačních bodů a letišť v okolí LKPR	38
Obrázek 3.11: Digital Elevation Model ČR obsažený v BGL souborech	38
Obrázek 3.12: (a)-SID LKPR (b)-STAR LKPR [36]	39
Obrázek 3.13: Vyhledávání v NDB	41
Obrázek 3.14 Návrh funkce prvků GUI	42
Obrázek 3.15: TCAS [39]	43

Obrázek 3.16: Použití sondy pro získání dat	44
Obrázek 4.1 Třída Core	45
Obrázek 4.2 Třída Aircraft	46
Obrázek 4.3 Aliasing	47
Obrázek 4.4 Antialiasing	47
Obrázek 4.5 Geometrie písma	48
Obrázek 4.6 Malé písmo	48
Obrázek 4.7 Malé písmo včetně obrysu	48
Obrázek 4.8 Implementace vykreslování	49
Obrázek 4.9: Class diagram jednotlivých vykreslovaných částí	50
Obrázek 4.10 Zpracování zpráv	51
Obrázek 5.1 Funkce vykonávající nejvíce práce	53
Obrázek 5.2 Nejčastěji volané funkce	53
Obrázek 5.3 Rozložení systému během testování	55
Obrázek 6.1 Porovnání FSX a MS Flight [19]	64
Obrázek 6.2 Porovnání vytvořeného a skutečného EFISu (nahore vytvořený EFIS, dole fotografie ze simulátoru úrovně Level D)) [26]	66
Obrázek 6.3 Porovnání vytvořeného a skutečného EFISu (nahore vytvořený EFIS, dole fotografie ze simulátoru úrovně Level D)) [26]	67

Seznam tabulek

Tabulka 2.1: Porovnání vlastností jednotlivých platforem	22
Tabulka 2.2: Porovnání vlastností existujících implementací	25
Tabulka 5.1 Pre-test dotazník s uživatelem A	56
Tabulka 5.2 Výsledky testu s uživatelem A	58
Tabulka 5.3 Pre-test dotazník s uživatelem B	58
Tabulka 5.4 Výsledky testu s uživatelem B	59
Tabulka 5.5 Soupis chyb.....	61

1 Úvod

Letectví je v dnešní době opět na vzestupu a světoví výrobci civilních dopravních letadel uvádějí na trh nové stroje, jako je Airbus A380 nebo Boeing 787 a současně vyvíjejí i zcela nové modely letadel. Konkrétně u společností Airbus je to model A350 XWB a u firmy Boeing je plánován nový typ Boeing 737 MAX, který by měl v budoucnu nahradit stávající Boeing 737. Stejně tak, jako se rozvíjí samotný letecký průmysl, rozvíjí se ruku v ruce s ním zájem běžných lidí o létání, a to nejenom jako pasažérů na palubách dopravních letounů, ale roste i počet těch, kteří by si rádi vyzkoušeli, jaké to je, takové letadlo ve skutečnosti pilotovat.

1.1 Motivace

Po událostech 11. září se zpřísnily bezpečnostní předpisy v letectví, přístup do kokpitu se tak stal pro většinu populace téměř nemožným. Jediným řešením zůstává využití leteckých simulátorů a to buď profesionálních, nebo amatérských. Profesionální simulátory zpravidla poskytují zážitky přibližující se skutečnému letu, jejich využití je ale finančně velmi náročné. Druhou možností je varianta amatérských simulátorů. Tyto je možno provozovat i v domácích podmínkách za přijatelnou cenu pro běžného zájemce. Tyto simulátory se zpravidla svým provedením nepřibližují úrovni simulace profesionálních zařízení. Většinou se jedná o produkty zábavního průmyslu, jako jsou PC hry, které jsou převážně cíleny na amatérské uživatele, a proto neposkytují komplexní simulaci.

Tato práce se zaměřuje na skupinu lidí, kteří se o letectví zajímají a rádi by si vyzkoušeli ovládání skutečného dopravního letadla a je pro ně použití profesionálních simulátorů nedostupné. Kvalita domácích simulátorů jim však nepostačuje, a tak hledají produkt, který by v domácích podmínkách, za přijatelnou cenu, poskytl věrnou simulaci kokpitu dopravního letounu a vytvořil tak pocit, jaký zažívají piloti v reálné situaci.

Motivací pro zahájení této práce bylo především vytvoření vlastního simulátoru kokpitu dopravního letadla bez toho, aby se uživatel musel omezovat možnostmi již existujících produktů a mohl si simulátor uzpůsobit podle svých vlastních představ. K této původní myšlence se časem přidala i možnost pracovat na ní i v rámci diplomové práce na FEL ČVUT, a tím i možnost věnovat jí více prostoru a dosáhnout tak kvalitnějšího výstupu, než by tomu bylo, pokud by byl simulátor vyvíjen pouze v rámci volného času. Práce byla rovněž motivována zájmem o vytvoření produktu, který bude mít v budoucnu nějaké reálné využití, ať už na amatérské nebo na profesionální bázi.

1.2 Cíle

Výstupem této práce by měla být simulace systému EFIS Boeingu 787, která by umožňovala všem uživatelům a to i těm, kteří nemají možnost přístupu k profesionálním simulátorům, zprostředkovat co nejvěrnější chování všech simulovaných systémů v kokpitu letadla.

1 ÚVOD

Jako dnes již naprostá většina letadel, má i Boeing 787-8 tzv. glass cockpit, tedy kokpit obsahující místo klasických analogových přístrojů LCD panely pro zobrazení informací. Systému využívající místo klasických analogových přístrojů LCD panely nebo CRT obrazovky se říká Electronic Flight Instrument System (EFIS). Cílem je přiblížit se po softwarové stránce kvalitě výstupu skutečným profesionálním simulátorům a to i v domácích podmínkách za minimální cenu ve srovnání s pořizovací cenou profesionálních řešení.

Simulovaný EFIS by měl obsahovat maximální technicky možné množství simulovaných systémů reálného letadla a to včetně možností jeho budoucího rozšiřování podle vývoje EFISu v letounu Boeing 787.

EFIS by měl být také schopen rovněž komunikovat s různými hardwarovými zařízeními, aby bylo umožněno uživatelům systému nejen provádět simulaci systémů kokpitu po softwarové stránce, ale i pro náročnější uživatele by měla existovat možnost připojení skutečných replik jednotlivých částí kokpitu.

V případě, že by výsledný produkt simuloval systémy Boeingu 787 na vysoké úrovni a i ze strany uživatelů byl o produkt zájem, mohl by simulátor v budoucnu přejít na komerční bázi, což by umožnilo jeho další rozvoj.

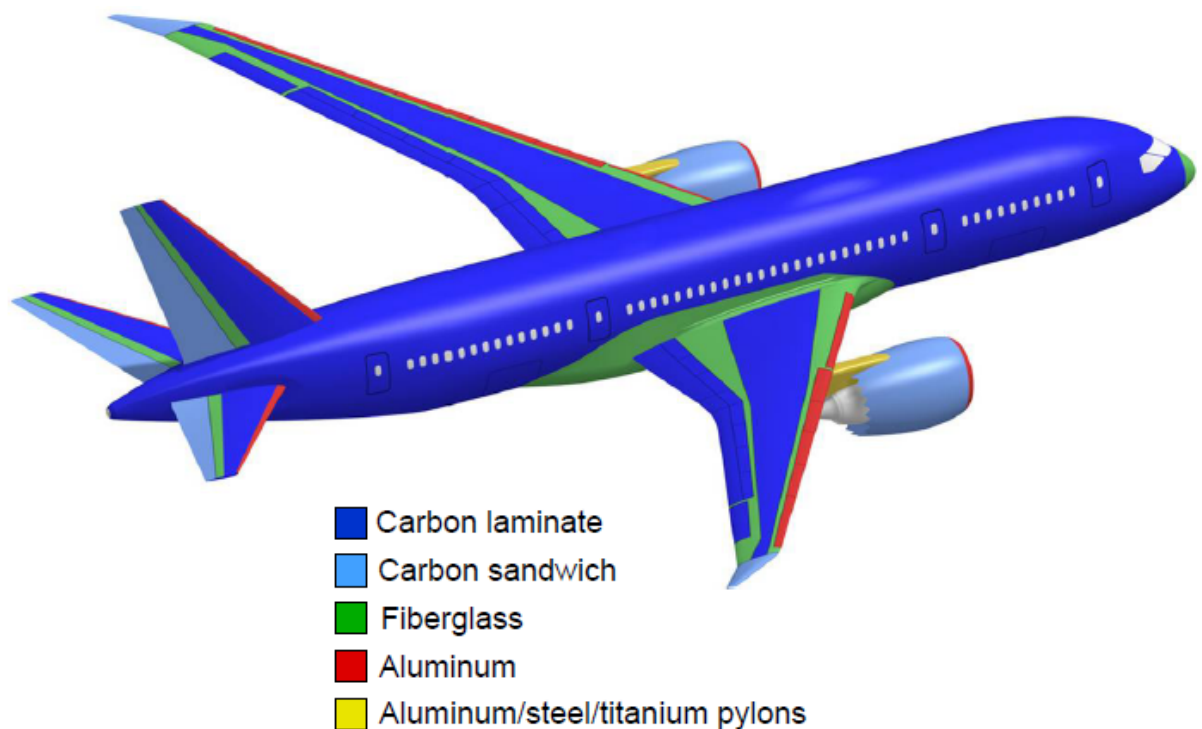
2 Analýza

V této kapitole si představíme letoun Boeing 787 a jeho jednotlivé systémy, jejichž simulací se budeme v této práci zabývat. Dále si ukážeme dělení leteckých simulátorů do různých kategorií. Rovněž zde provedeme analýzu již existujících řešení, která jsou dnes dostupná, a zvolíme platformu, která nám poskytne simulaci vnějšího prostředí a fyzikálního modelu letadla.

2.1 Boeing 787

Boeing 787 Dreamliner je aktuálně posledním modelem civilního dopravního letadla od společnosti Boeing. Jedná se o dvumotorový letoun, který by měl časem nahradit modely Boeing 767, zároveň je zaměřen do stejného segmentu jako konkurenční Airbus A330 a právě vyvíjený Airbus A350 XWB. Model 787 se bude v budoucnu vyrábět ve třech verzích a to 787-3, 787-8 a 787-9 [1]. Jednotlivé verze se od sebe liší především množstvím pasažérů, které jsou schopny přepravit a pak také vnějšími rozměry. Technologie použité v letadle jsou však u všech modelů stejné.

Boeing 787-8 je v mnoha ohledech revolučním strojem. Jedná se o první civilní letoun, který využívá kompozitní materiály místo hliníku na celém trupu v tak velkém rozsahu (Obrázek 2.1). Dále díky novým motorům je hladina hluku produkovaná letadlem nižší o 60% [1] a i v řadě dalších systémů v letadle lze nalézt mnoho inovativních prvků.



Obrázek 2.1: Struktura materiálů v draku letadla [1]

Tyto inovativní postupy spolu však přinášejí i potíže, jako jsou například problémy při spojování jednotlivých sekcí trupu nebo problémy při certifikaci FAA [2] Různé problémy již také několikrát způsobily odložení termínu uvedení do ostrého provozu.

Letoun již splnil certifikační požadavky FAA pro způsobilost k přepravování pasažérů i certifikaci Extended-range Twin-engine Operational Performance Standards (ETOPS), která upravuje podmínky pro dvoumotorová letadla, mimo dosah letišť, zejména nad oceánem. Tento certifikační proces započal 26. června 2011 [3]. První stroj byl uveden do provozu u aerolinií ANA 26. října 2011. [4] V současnosti jsou v komerčním provozu dva stroje, oba u ANA.

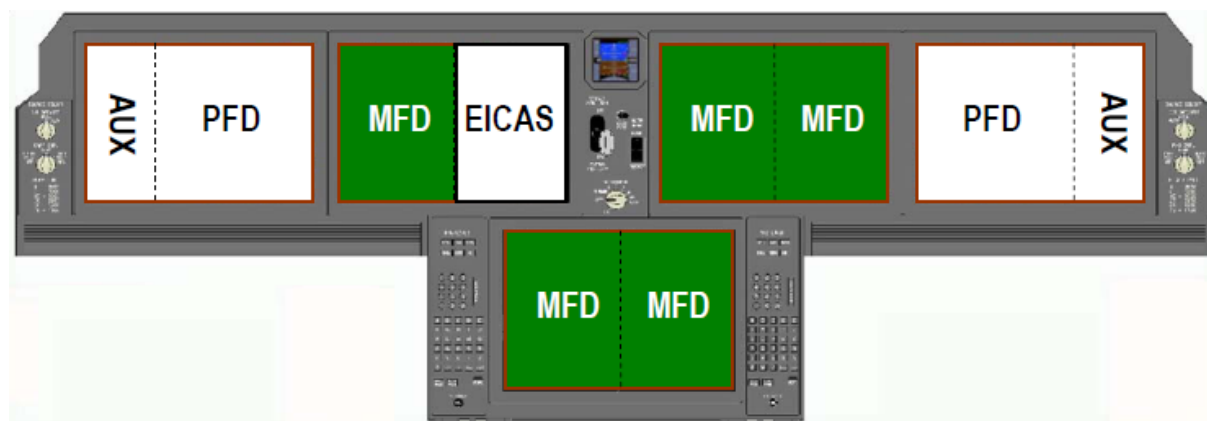
V této práci se zaměřujeme konkrétně na simulaci systémů letadla Boeing 787-8 dle International Civil Aviation Organization (ICAO) označovaného taktéž jako B788 [5], přesněji na model s motory GENx od firmy General Electric, jelikož doposud byly vyrobeny pouze modely tohoto typu a zároveň i na tento typ má společnost Boeing nejvíce objednávek. V České republice si zatím jeden stroj Boeing 787-8 objednala společnost Travel Service [6].

2.2 Simulované systémy

V této práci se zaměříme především na simulaci funkcí LCD panelů přítomných v kokpitu. Hlavní část glass cockpitu Boeingu 787-8 je tvořena pěti patnáctipalcovými LCD panely (Obrázek 2.2). Tyto LCD panely jsou dále virtuálně rozděleny na menší panely (Obrázek 2.3) a to na Primary Flight Display (PFD), Multi-Function Display (MFD) a Engine Indicating and Crew Alerting System (EICAS). MFD znamená, že na jednotlivé pozice MFD lze zobrazit různé panely s údaji podle toho, které jsou v daný okamžik vyžadovány. Konkrétně se dá zobrazit navigační displej (ND), displej s přehledem funkce jednotlivých systémů letadla (SYS), elektronický checklist (CHKL), Control Display Unit (CDU), informační displej (INFO) a menu pro komunikaci (COMM).



Obrázek 2.2: EFIS Boeingu 787-8 [7]



Obrázek 2.3: Dělení EFISu na jednotlivé displeje [7]

V kokpitu se dále nachází záložní systém pro poskytování údajů o letu neboli Integrated Standby Flight Display (ISFD), dvakrát průhledový displej neboli Head-Up Display (HUD) a dvakrát Electronic Flight Bag (EFB) - pomocný displej nahrazující používání papírových dokumentů.

Aby bylo možné se simulovanými systémy dostatečně interagovat, podobně jako je tomu ve skutečném letadle, bude součástí této práce i simulace některých ovládacích panelů z kokpitu. Na jednotlivé simulované systémy se nyní podíváme podrobněji.

2.2.1 Primary Flight Display (PFD)

PFD je základním displejem, který pilotům moderních letadel poskytuje nejdůležitější údaje o letu. PFD vznikl jako spojení šesti základních dříve používaných analogových přístrojů a to výškoměru, rychloměru, zatáčkoměru, variometru, umělého horizontu a ukazatele směru. PFD kromě toho, že do sebe integruje základní šestici přístrojů, obsahuje také množství dalších funkcí, jako je zobrazení režimu letu, informace ze systému varujícího před kolizemi (Traffic Alert and Collision Avoidance System - TCAS) atd.



Obrázek 2.4: Primary Flight Display

PFD (Obrázek 2.4) v kokpitu Boeingu 787-8 je zobrazen v EFISu dvakrát, jednou pro kapitána a jednou pro prvního důstojníka (F/O). Pozice těchto displejů nelze za standardních podmínek nijak změnit. Samotný PFD displej se dělí na dvě části – samotné PFD (horní část) a část s minimapou (spodní část).

PFD poskytuje následující informace [8]:

- Informace o letovém režimu
- Indikovanou letovou rychlost
- Výšku
- Vertikální rychlost
- Informace o náklonu
- Informace o zatáčení
- Informace z radiovýškoměru
- Instrument Landing System (ILS)
- Indikace o minimech při přistání
- Informace o kursu
- Informace o poruchách motorů a varování ze systémů GPWS a PWS

Minimapa zobrazuje následující údaje ve fixním rozsahu 20nm [8]:

- Trasu letu
- Letový plán včetně jednotlivých bodů trasy
- Výškový profil terénu
- Alternativní letiště
- Informace o ranveji
- Informace ze systému TCAS
- Terénní radar

2 ANALÝZA

- Meteorologický radar
- Aktuální kurs

Výhodou toho, že v kokpitu B788 jsou využity rozměrné patnáctipalcové displeje a tedy i rozměrné samotné PFD, je jeho možné budoucí rozšiřování o další funkcionality, jako například o Synthetic Vision System (SVS) [9], umožňující zobrazení trojrozměrného, počítačem generovaného vnějšího prostředí přímo v PFD nebo perspektivní PFD [9] .

2.2.2 Auxiliary Display (AUX)

AUX (Obrázek 2.5) je pomocný panel, umístěný po straně PFD, který zobrazuje identifikační údaje o letu, jako je číslo letu, volací znak, nastavení odpovídače a podobně. Pozice obou AUX panelů je stejně jako u PFD za standardních podmínek fixní.

AUX panel lze také využít i pro zobrazování komunikace s ATC pomocí tzv. „ATC UPLINK“ funkce, která umožňuje komunikaci s ATC čistě v elektronické formě.



Obrázek 2.5: Auxiliary Display

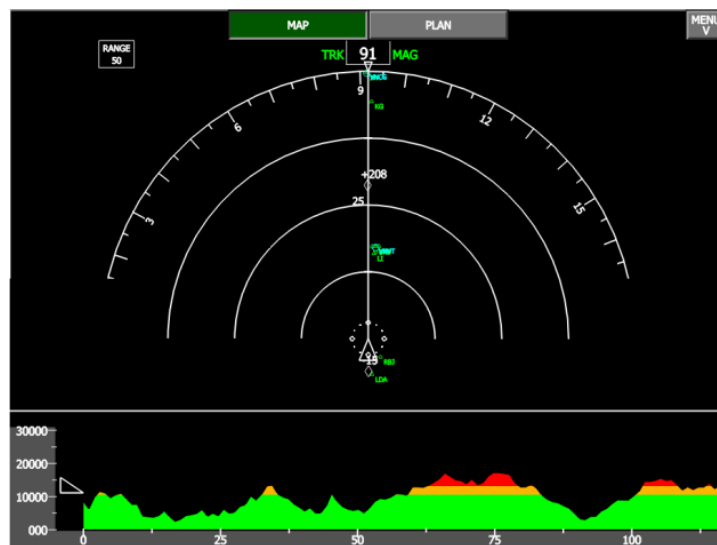
2.2.3 Navigation Display (ND)

Navigační displej je jednou ze základních součástí každého EFISu. Jeho hlavním úkolem je poskytovat pilotům navigační údaje, jako je plán letu včetně aktuální pozice letadla, jednotlivé navigační body v okolí letadla, letiště atd. Součástí navigačního displeje u moderních strojů bývají také mnohé další systémy jako informace ze systému TCAS, odrazy meteorologického nebo pozemního radaru atd.

Letadla, která nejsou vybavena EFISem a obsahují pouze analogové přístroje, nemívají žádný ekvivalent navigačního displeje a piloti se tak musí spoléhat na klasické papírové mapy. Příchod elektronického systému tedy výrazně přispěl k ulehčení práce pilotů, a tím i zvýšení bezpečnosti při navigaci.

Navigační displej v Boeingu 787-8 (Obrázek 2.6) obsahuje jak všechny výše zmíněné prvky, tak i mnoho dalších funkcí. Výhodou ND u Boeingu 787-8 oproti starším modelům je fakt, že ovládání jeho jednotlivých funkcí je možné za použití touchpadu, pomocí kterého piloti ovládají kurzor na obrazovce, a tím je dosaženo výrazného zjednodušení práce při porovnání s ovládáním výhradně vstupem z klávesnice. I u ostatních MFD je umožněno ovládání pomocí touchpadu.

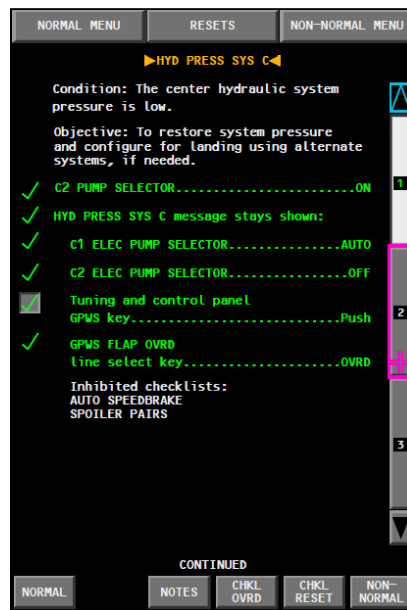
ND dále obsahuje Vertical Situation Display (VSD), což je displej zobrazující výškový profil terénu ve směru letu a v případě, že by hrozilo nebezpečí srážky s terénem, tak systém upozorní na vzniklou situaci piloty. ND stejně jako minimapa v PFD obsahuje informace ze systému TCAS a umožňuje zobrazit informace z pozemního radaru nebo z meteorologického radaru (WXR).



Obrázek 2.6: Navigation Display

2.2.4 Electronic checklist (CHKL)

Checklist je pomůcka pro piloty, která zaručuje, že jednotlivé operace během letu budou provedeny správně. Jedná se tedy o jakýsi seznam jednotlivých úkonů, které je potřeba ke korektnímu ukončení nějakého úkolu v daném pořadí splnit. Pro různé situace jsou k dispozici různé checklisty, a to jak pro běžné situace, tak různé poruchy. Elektronický checklist (Obrázek 2.7) je dalším krokem k zajištění správnosti provedení jednotlivých procedur. Piloti již nepotřebují mít checklist k dispozici v papírové formě a ručně procházet jednotlivé jeho body. Elektronický checklist provádí kontrolu splnění jednotlivých bodů automaticky a na případné nedostatky piloty upozorňuje. Tímto je opět zvýšena jistota, že jednotlivé postupy budou provedeny správně. CHKL obsahuje jak normální postupy, tak nouzové postupy. Nouzové procedury jsou zpravidla provázány s varováním na EICAS displeji.



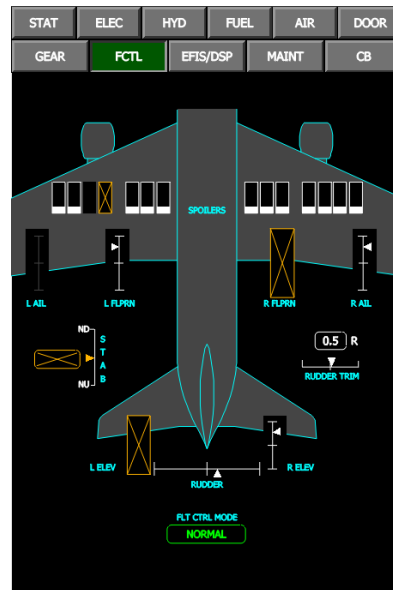
Obrázek 2.7: Electronic Checklist [8]

2.2.5 Synoptics Display (SYS)

SYS displej (Obrázek 2.8) slouží k zobrazení informací o stavu jednotlivých systémů letadla a informací potřebných k údržbě. U jednotlivých zobrazení je pak možno identifikovat stav jednotlivých komponent v daném systému a tomuto přizpůsobit provoz.

SYS displej zobrazuje informace o:

- Elektrickém systému
- Hydraulickém systému
- Palivovém systému
- Klimatizačním systému
- Dveřním systému
- Podvozku / brzdách
- Řídících plochách
- EFISu / Displejích
- Údržbě
- Pojistkách

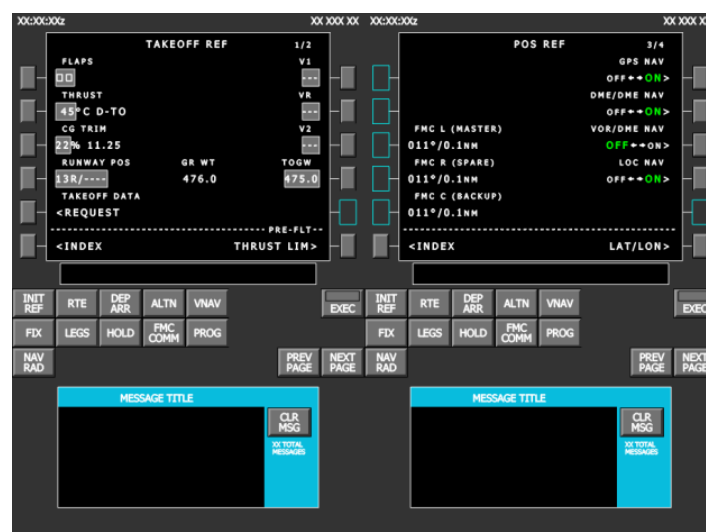


Obrázek 2.8: Synoptics Display

2.2.6 Control Display Unit (CDU)

CDU je prostředek pro piloty, pomocí kterého komunikují s palubním počítačem (Flight Management Computer - FMC). Je to tedy hlavní prostředek pro naprogramování FMC pro let. Dříve bylo CDU řešeno jako mechanická klávesnice s malým displejem, který zobrazoval jednotlivé stránky, avšak u Boeingu 787-8 je CDU (Obrázek 2.9) plně interaktivní a je ovládáno pomocí MFK (Multifunction Keypad) a touchpadu.

Samotné CDU se dělí na dvě základní části. Na displej na kterém se zobrazují jednotlivé stránky z FMC. Na těchto stránkách lze zobrazit veškeré informace dostupné v FMC, některé údaje lze rovněž změnit. Druhou částí CDU jsou tlačítka, pomocí kterých lze mezi jednotlivými stránkami přecházet.



Obrázek 2.9: Control Display Unit

2.2.7 Communication Display (COMM)

Komunikační displej (Obrázek 2.10) umožňuje provádět komunikaci s ATC (Air traffic control) nebo s domovskou společností. Obsah tohoto displeje může být uzpůsoben podle požadavků jednotlivých aerolinií.

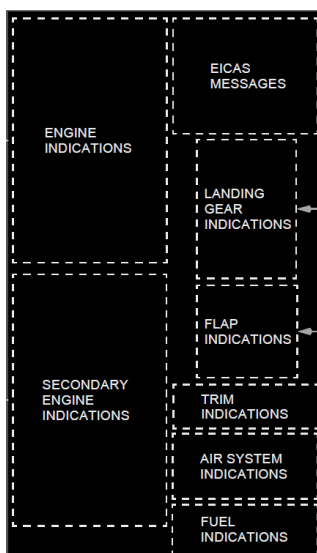


Obrázek 2.10: Communication Display [8]

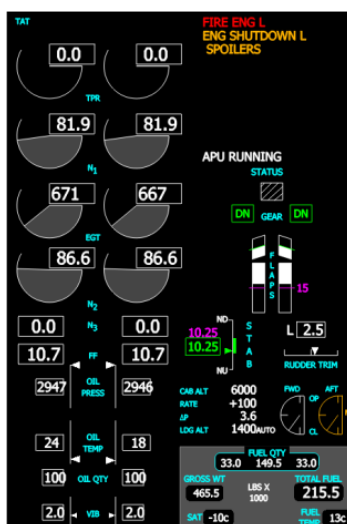
2.2.8 Engine Indicating and Crew Alerting System (EICAS)

EICAS v sobě zahrnuje množství provozních informací. Zejména se jedná o veškeré informace o stavu motorů. Tyto informace se dělí na primární a sekundární. Mezi primární patří otáčky vysokotlaké a nízkotlaké turbíny, teplota plynů v motorech. Mezi sekundární se pak řadí informace o oleji, spotřebě nebo vibracích motorů. Další kategorii informací které zobrazuje panel EICAS, jsou varovné zprávy. Tyto zprávy upozorňují piloty na různé nestandardní situace, které mohou nastat a to od špatného nastavení přístrojů v letadle až po informace o selhání jednotlivých systémů. V dalších kategoriích jsou zobrazeny informace o podvozku, klapkách, nastavení výškového a směrového kormidla. Dále jsou zde kategorie pro zobrazení informací o přetlakování a o palivu.

V letadle Boeing 787-8 je panel EICAS zobrazen vždy na stejné pozici, a to buďto na straně kapitána nebo F/O. Rozmístění indikátorů zapadajících do jednotlivých kategorií v panelu EICAS je na (Obrázek 2.11), příklad situace pak na (Obrázek 2.12).



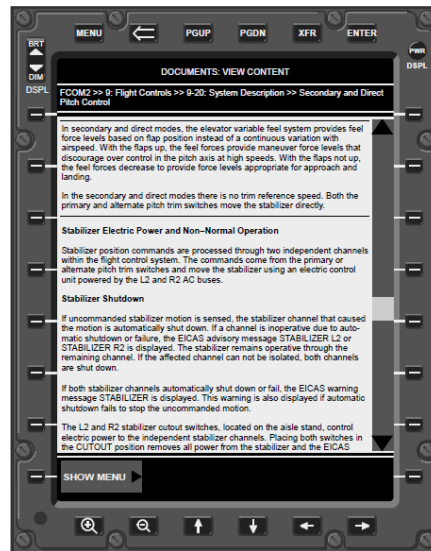
Obrázek 2.11: Dělení EICAS displeje [8]



Obrázek 2.12: EICAS displej

2.2.9 Electronic Flight Bag (EFB)

EFB (Obrázek 2.13) je systém tvořený množstvím aplikací majících za cíl pomoci posádce s rutinními úkoly, se zvýšením bezpečnosti a redukcí závislosti na množství papírových dokumentů [8]. Jednotlivé aplikace a data obsažená v EFB se liší na základě požadavků jednotlivých aerolinií.



Obrázek 2.13: Electronic Flight Bag [8]

2.2.10 Head-Up Display (HUD)

HUD (Obrázek 2.14) je průhledný displej umístěný ve výhledu pilota. Displej tak poskytuje pilotovi potřebné informace bez toho, aby pilot musel jakkoli uhýbat svým pohledem směrem dolů k ostatním přístrojům. HUD byl původně vyvinut pro použití ve vojenských letadlech, dnes se však již běžně využívá i v civilních letadlech a dokonce i automobilech. V Boeingu 787 HUD poskytuje základní letová data z velké části shodná s těmi, která poskytuje PFD.

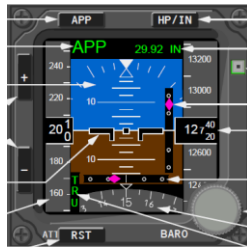


Obrázek 2.14: Head-Up Display [8]

2.2.11 Integrated Standby Flight Display (ISFD)

ISFD je záložní letový systém, který v sobě obsahuje základní šestici přístrojů stejně jako PFD. ISFD je vybaven nezávislými zdroji dat o letu, a proto jej lze použít v případě poruchy primárních systémů k bezpečnému pokračování letu.

Dříve se i u letadel vybavených systémem EFIS využívalo místo ISFD jako zálohy klasických analogových přístrojů. U Boeingu 787-8 je však již i tento systém plně digitální (Obrázek 2.15). Jediným analogovým přístrojem, který v kokpitu B788 zůstal, je již pouze záložní magnetický kompas.



Obrázek 2.15: Integrated Standby Flight Display [8]

2.2.12 Ovládací panely

Jak bylo zmíněno v úvodu kapitoly, aby bylo možné s jednotlivými simulovanými systémy plnohodnotně interagovat, je potřeba provést i simulaci panelů, které slouží k jejich ovládání ve skutečném letadle.

V první řadě je třeba provést simulaci MFK (Obrázek 2.16), jelikož se jedná o panel, pomocí kterého jsou zadávána data do palubního počítače (FMC). Dále je potřeba simulovat i funkci panelu EFIS, pomocí kterého jsou nastavovány parametry navigačního displeje. V neposlední řadě je nutné simulovat MIP panel, ve kterém se nastavují parametry pro autopilota a jednotlivé módy letu. Tyto tři panely jsou pro korektní funkci simulátoru zásadní, u ostatních panelů lze provést jejich implementaci dodatečně.



Obrázek 2.16: Multifunction Keypad

2.3 Druhy simulátorů

Letecké simulátory se dají rozdělit na dvě hlavní skupiny - a to na simulátory amatérské a simulátory profesionální. Mezi amatérské simulátory se především řadí různé PC hry, které umožňují uživatelům vyzkoušet si roli pilota a přitom neabsolvovat letecký výcvik. Mezi tyto simulátory se řadí například hry jako je MS Flight Simulator nebo XPlane. Výhodou využití těchto simulátorů je, že díky rozmanitým možnostem nastavení reálnosti je mohou užívat jak lidé, kteří se s pilotováním do té doby vůbec nesetkali, tak také lidé, kteří se o letectví zajímají hlouběji a chtějí při používání simulátoru zažít pocit, který by se co nejvíce blížil realitě.

2 ANALÝZA

Zmíněné simulátory lze dnes, při existenci množství firem zabývajících se výrobou replik jednotlivých částí letadel, rozšířit i po hardwarové stránce a je tak možné i v kategorii amatérských simulátorů vytvořit věrnou kopii kokpitu letadla a tedy i simulátor na vysoké úrovni (Obrázek 2.17). Nicméně i přes to, že některé amatérské simulátory dosahují vysoké technické úrovně, nelze je bez certifikace (2.3.2) používat pro profesionální účely.



Obrázek 2.17: Simulátor Boeingu 737NG v ČR [10]

Další skupinou simulátorů jsou simulátory profesionální, které se používají pro výcvik posádek letadel a vzhledem k tomu jsou na ně kladeny mnohem vyšší požadavky, co se týká přesnosti provedení a vnímání letu během simulace. Tyto požadavky jsou dány normami, které každý profesionální simulátor, který je využíván k oficiálnímu výcviku musí splňovat. Profesionální simulátory se podle své technické úrovně a zaměření dělí do několika kategorií.

2.3.1 Kategorie simulátorů

Dělení profesionálních simulátorů se ve světě v různých oblastech mírně liší. Ve Spojených státech určuje požadavky na jednotlivé kategorie simulátoru Federal Aviation Administration (FAA), konkrétně jsou v předpisu [11]. V Evropě existuje European Aviation Safety Agency (EASA), u níž jsou jednotlivé kategorie a požadavky na ně stanoveny v dokumentu [12]. Konkrétně pak v České republice, která je součástí EASA, se o dodržování norem pro jednotlivé kategorie simulátorů stará Úřad pro civilní letectví (CAA). V tomto dokumentu budeme uvádět popisy jednotlivých kategorií tak, jak jsou definovány CAA, tedy v dokumentech [13], [14], [15] a [16].

Základní dělení simulátorů je dle CAA na Letové simulátory pro letouny [13], Letová výcviková zařízení pro letouny [14], Trenažéry letových a navigačních postupů pro letouny [15] a Základní přístrojová výcviková zařízení [16].

Základní přístrojové výcvikové zařízení (BITD) je dle [16] pozemní výcvikové zařízení předvádějící stanoviště pilot-žáka třídy letounů. Může být vybaveno panely přístrojů s obrazovkami a pružinami zatěžovaným řízením poskytujícím výcvikovou základnu nejméně pro nácvik postupů letu podle přístrojů. Tedy jedná se o simulátor, na kterém se piloti seznámí se základy fungování letadla, a proto ani požadavky na tuto kategorii simulátorů nejsou co do přesnosti provedení simulátoru, tak do přesnosti reakcí na

pilotovo chování nikterak přísné. Dalo by se zjednodušeně konstatovat, že tato kategorie se blíží některým amatérským simulátorům, jako je MSFS nebo XPlane.

Trenažér letových a navigačních postupů pro letouny (FNPT) je dle [15] výcvikové zařízení představující prostředí pilotní kabiny/pilotního prostoru obsahující sestavu vybavení a počítačových programů nutných k předvádění typu nebo třídy letounů v letových podmínkách v rozsahu odpovídajícímu tomu, jak systém funguje v letounu. U této kategorie je již vyžadováno, aby simulátor obsahoval model kokpitu a to včetně rozmístění přístrojů na stejných pozicích jako ve skutečném letadle, nicméně není požadováno, aby byly naprosto shodné jako u simulovaného stroje. Tato kategorie se dále dělí na FNPT Level I, FNPT Level II a FNPT Level II MCC.FNPT Level II. MCC označuje simulátor pro výcvik vzájemné součinnosti posádky (Multi-Crew Cooperation). Tato kategorie simulátorů by se dala vzdáleně přirovnat k amatérským simulátorům s vybudovaným kokpitem.

Letové výcvikové zařízení pro letouny (FTD) je dle [14] replika přístrojů v letounu, vybavení, panelů a řízení v otevřeném nebo uzavřeném prostoru pilotní kabiny, provedená ve skutečné velikosti včetně sestavy vybavení a softwarových programů nutných k předvádění letounu v pozemních a letových podmínkách. U této kategorie není požadována pohyblivá základna, avšak je již vyžadován kokpit specifického typu letadla a plná simulace jeho systémů. Tato kategorie se dělí na FTD Level 1 (Obrázek 2.18) a FTD Level 2 (Obrázek 2.19).



Obrázek 2.18: Simulátor ČSA - A320– úrovně FTD Level 1 [17]



Obrázek 2.19: ČSA - ATR42/72 – úrovně FTD Level 2/FNPT II MCC [17]

Simulátor pro letoun je dle [13] replika pilotní kabiny provedená ve skutečné velikosti specifického typu nebo provedení, modelu a série včetně úplného vybavení a počítačových programů nutných k předvádění letounu v pozemním a letovém provozu, systém vizuální orientace poskytující výhled z pilotní kabiny a pohybový systém. U této kategorie je již tedy vyžadována uzavřená pilotní kabina provedená ve skutečné velikosti a to včetně simulace všech systémů letadla. Rovněž je u této kategorie vyžadována pohyblivá základna pro navození pocitu skutečného pohybu letadla. Tato kategorie je rozdělena na simulátory Level A-D(Obrázek 2.20).



Obrázek 2.20: Simulátor ČSA – A320 – Level D [17]

2.3.2 Certifikace

Aby mohl být systém používán jako profesionální simulátor patřící do některé kategorie zmíněných výše, musí být certifikován. Certifikace jednotlivým simulátorům uděluje v České republice Úřad pro civilní letectví. Pro získání certifikace je třeba splňovat

všechny parametry pro danou kategorii simulátoru a systém musí projít QTG (Qualification Test Guide) testy.

2.3.3 Cílová kategorie

Při tvorbě návrhu simulátoru bychom se měli rozhodnout do které kategorie jej směřovat. Jak už bylo zmíněno v úvodu, je snahou vytvořit co možná nejvěrnější simulaci EFISu Boeingu 787-8, která by simulovala co možná nejvíce jeho systémů, která by umožnila uživatelům co nejvěrnější pocit ovládání letadla. V kategorii amatérských simulátorů by měl tento simulátor tedy cílit zejména na ty zájemce, kteří buď vlastní, nebo si hodlají vytvořit MOCKUP kokpitu letadla a hledají komplexní software pro simulaci jeho EFISu.

Je potřeba však také uvažovat nad tím, zdali by se dala pro vytvářený systém získat certifikace a tedy ho moci používat i profesionálně. Z kapitoly 2.3.1 však plyne, že jednotlivé kategorie profesionálních simulátorů vždy obsahují konkrétní požadavky, jak na software, tak na hardware a jelikož v této práci se zabýváme pouze vývojem software, není tedy možné pro něj samostatně získat certifikaci. Lze však říci, že po softwarové stránce by měl náš EFIS splňovat požadavky pro kategorii FTD Level 1, jelikož u této kategorie není požadována kompletní simulace všech systémů letadla. V případě, že by se do námi vytvářeného simulátoru systému EFIS podařilo implementovat všechny funkce letadla, mohla by pak výsledná úroveň odpovídat kategorii FTD Level 2 nebo Level A.

2.4 Existující platformy

Vzhledem k tomu, že se zaměřujeme pouze na vývoj systému EFIS, je potřeba zvolit platformu, která by nám poskytla základní simulaci letu, jako je zobrazení vnějšího prostředí a simulace fyziky letu. Vzhledem k tomuto je potřeba se při výběru vhodné platformy zaměřit především na tyto aspekty.

2.4.1 MS FS 2004

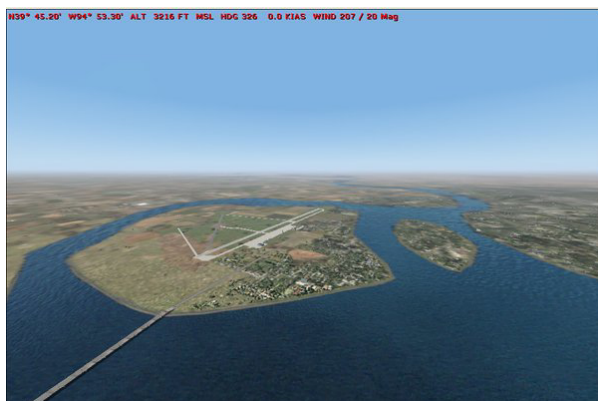
Microsoft Flight Simulator 2004 je devátou verzí simulátoru vydanou v roce 2003 a dnes je nejrozšířenějším amatérským leteckým civilním simulátorem a to především díky tomu, že je pro něj vyvinuto velké množství doplňků (add-onů). Jelikož pro všechny doplňky nejsou dostupné i verze pro novější MSFS, zůstává stále velké množství uživatelů u FS2004.

Po grafické stránce byl u FS2004 nejviditelnější rozdíl, oproti předcházejícím verzím MSFS, zejména v simulaci počasí [18], kde dochází již ke generování oblohy včetně mraků nejen v nejbližším okolí letadla ale po celé obloze. Také simulace scenerie byla oproti předchozím verzím kvalitnější (Obrázek 2.21).

Fyzikální model u simulátorů ze série MSFS nepracuje přímo s geometrií letadla, tedy jednotlivé dynamické parametry nejsou určovány z tvaru modelu letadla, nýbrž všechny aerodynamické, výkonové a jiné parametry jsou uloženy v konfiguračních souborech.

2 ANALÝZA

Lze tedy ve MS Flight Simulátoru provozovat i letadla s nedokonalým modelem, pokud mají k dispozici korektní aerodynamické parametry.



Obrázek 2.21: Microsoft Flight Simulator 2004 [19]

2.4.2 MS ESP/FSX

Microsoft Flight Simulátor X je nástupce FS2004. Jedná se již o desátou a prozatím poslední verzi ze série Flight Simulátorů. Byl vydán v roce 2006 a přinesl řadu novinek. Mezi nejvýraznější se řadí zejména vylepšení grafického enginu a podpora textur ve vysokém rozlišení (Obrázek 2.22) až 7cm/px. Dále FSX simuluje i ostatní dopravní prostředky, jako je automobilový provoz, lodní provoz nebo provoz přímo na jednotlivých letištích. Stejně jako FS2004 i FSX obsahuje model celé zeměkoule a více než 24 000 letišť.

Parametry pro letovou dynamiku jednotlivých strojů jsou rovněž jako v případě FS2004 uloženy v konfiguračních souborech a letová dynamika tedy není přímo závislá na modelech jednotlivých letadel.

Do uvedení oznámeného produktu Microsoft Flight [20] se jedná prozatím o poslední verzi civilního leteckého simulátoru od firmy Microsoft.



Obrázek 2.22: Microsoft Flight Simulator X [19]

2.4.3 Prepar3D

Jedná se o simulátor vyvinutý firmou Lockheed Martin, který funguje stejně jako FSX na platformě ESP [21]. Software je především určen pro společnosti a vládní instituce,

tedy pro profesionální využití, jako například v National Flight Academy v USA [22], kde se využívá pro výuku.

Tomu, že se jedná o software především pro profesionální využití, odpovídá i jeho cena, která činí \$499 za jednu licenci a \$9.95 měsíční poplatek pro vývojáře.

2.4.4 XPlane 9

XPlane je letecký simulátor představený v roce 2007. V posledních letech zaznamenal velký nárůst uživatelů, zejména po oznámení společnosti Microsoft o uzavření studia ACES, které vyvíjelo sérii MSFS. Po grafické stránce se XPlane kvalitou blíží FSX (Obrázek 2.23), nicméně jelikož nepoužívá funkci pro automatické generování 3D objektů na povrchu, je krajina bez použití dalších doplňků znatelně chudší a méně plastická oproti FSX.

Na rozdíl od MSFS pracuje XPlane přímo s geometrií modelu, což znamená, že dynamické letové parametry jsou určeny na základě skutečné geometrie nikoliv pouze na základě předem zadaných parametrů. Použitím tohoto principu se simulace v XPlane blíží více realitě než v Microsoft Flight Simulatoru X.

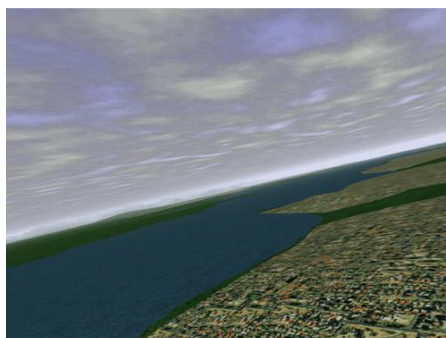
Výhodou XPlane je rovněž častější aktualizace.



Obrázek 2.23: XPlane 9 [19]

2.4.5 FlightGear

FlightGear je neplacený open-source projekt. Jelikož není vyvíjen jako komerční produkt, nedosahuje takové technické úrovně jako ostatní komerční simulátory. Po grafické stránce by se dal FlightGear zařadit kvalitativně mezi FS2002 a FS2004 (Obrázek 2.24), což ho značně handicapuje jako kandidáta pro využití v našem projektu.



Obrázek 2.24: FlightGear [19]

2.4.6 Výběr platformy

Při výběru platformy se zaměříme u zvažovaných variant hlavně na hlediska kvality grafického podání vnějšího prostředí a na kvalitu fyzikálního modelu. Ostatní možnosti těchto platforem náš simulátor nebude využívat, proto je nebudeme brát do úvahy. Hodnocením kvality grafiky jednotlivých možností se zdá být nejlepší volbou simulátor na platformě MS ESP. Tedy FSX nebo Prepar3D. Využití simulátoru Prepar3D brání, jak jeho vysoká pořizovací cena, tak i skutečnost, že jej nelze ani legálně pořídit pro domácí užití.

Z hlediska fyzikálního modelu se dle [23] při porovnání modelů fungujících v FSX a XPlane ukázalo, že princip využívání pro určování dynamiky letu přímo model letadla namísto uložení parametrů pro letovou dynamiku v konfiguračních souborech nepřináší žádnou relevantní výhodu a obě tato řešení jsou s ohledem na výsledky téměř rovnocenná. Rovněž to, že je dynamika letu závislá na datech v konfiguračních souborech, je pro nás výhodnější, jelikož nehodláme v rámci této práce vytvářet model letadla, ale spokojíme se pouze s nastavením letových parametrů.

Porovnání jednotlivých parametrů uvažovaných při volbě platformy jsou v (T1).

	FS2004	FSX	Prepar3D	XPlane	FlightGear
Grafický Engine	-	+	+	+	-
Fyzikální Engine	+	+	+	+	?
Licence	+	+	-	+	+
Rozhraní	-	+	+	-	-

Označení	Popis
+	Implementace dané vlastnosti je dostačující
-	Implementace dané vlastnosti není dostačující
?	Pro určení není dostatek informací

Tabulka 2.1: Porovnání vlastností jednotlivých platforem

Po posouzení jednotlivých možností pro náš simulátor EFISu zvolíme jako platformu Microsoft Flight Simulátor X.

2.5 Existující řešení

Nyní si představíme řešení, která jsou dnes na trhu dostupná a jejich funkce je stejná, nebo se blíží našemu zamýšlenému programu, tedy simulaci systému EFIS letadla Boeing 787-8. Jelikož doplňků pro FSX, případně FS2004 existuje celá řada, avšak mnoho z nich je nedostatečné kvality, tak se v této části textu zaměříme zejména na komerční produkty. Tyto zpravidla dosahují vyšší kvality než neplacené produkty.

2.5.1 Abacus 787

Jedná se o modul (plug-in) do FSX obsahující model letadla Boeing 787 Dreamliner. Nejde přímo o produkt, který by se přímo typově shodoval s naším programem, jelikož se jedná o celý model letadla, avšak nám jde hlavně o simulaci EFISu. Budeme porovnávat uvedenou část modelu Abacus 787 s reálným EFISem Boeingu 787-8.

Při porovnání EFISu modelu Abacus 787 (Obrázek 2.25) a reálného EFISu (Obrázek 2.2) dojdeme k závěru, že sice na první pohled je struktura obou EFISů podobná, avšak produkt firmy Abacus je velmi nepropracovaný a na jejich panelech se zobrazuje pouze zlomek údajů skutečných panelů. Zároveň modelu Abacus 787 zcela chybí FMC, jakákoli reálná simulace letu je zde tudíž vyloučena [24].



Obrázek 2.25: Abacus Boeing 787 [24]

U tohoto produktu by se dalo konstatovat, že se jedná o simulátor zaměřený zejména na kategorii amatérských zájemců, kteří chtějí mít pocit pilotování Boeingu 787. Přesnost simulace pro ně však není stěžejní.

2.5.2 Project Magenta

Tento produkt je dnes z dostupných variant asi nejbliže námi zamýšlenému projektu. Project magenta je simulátor systémů EFIS letadel Boeing 737NG, 747-400, 777 a 757/767 nebo Airbus A320, 330 a 340.

Project Magenta simuluje nejenom kompletní systém EFIS (Obrázek 2.26), ale rovněž i řadu dalších systémů jako jsou ovládací panely nebo stanice pro instruktora.

V naprosté většině amatérských kokpitů se v dnešní době využívá jako základní software právě Project Magenta. Nicméně model, jehož EFIS zamýšlíme simulovat v této práci, Boeing 787 Project Magenta nesimuluje, není tedy přímým konkurentem pro náš program. Nejedná se o otevřený projekt, a proto není ani možné ho rozšířit o plánovanou simulaci B788.



Obrázek 2.26: Project Magenta Boeing 737NG [25]

2.5.3 Thales 787

Jedná se o profesionální simulátor Boeingu 787 úrovně Level D od firmy Thales (Obrázek 2.27). Protože je tento simulátor certifikovaný Level D a jeho cena se pohybuje okolo 15-20 milionů amerických dolarů, tak je pro naprostou většinu zájemců nedostupný.



Obrázek 2.27: Thales Boeing 787 simulátor [26]

2.5.4 Posouzení existujících řešení

Z předcházejících kapitol vyplývá, že dnes existují tři základní typy simulátoru EFISu. První variantou je simulace v rámci využití EFISu obsažených u komerčních doplňků do FSX. V dnešní době je však dostupný pouze jediný model simulující systémy Boeingu 787, a to Abacus 787. Tento model ale, jak již bylo zmíněno, nedosahuje příliš velké kvality zpracování. Druhou možností je využití specializovaného softwaru, simulujícího přímo systém EFIS jednotlivých letadel, jako je například Project Magenta. Tento druh programu je i nejbližší námi zamýšlenému pojetí. V této kategorii však prozatím neexistuje program, který by simuloval přímo EFIS letounu Boeing 787. Poslední možností je využití profesionálního řešení, jako je například simulátor od firmy Thales. U tohoto typu simulátoru je sice zaručena přesná simulace EFISu konkrétního modelu B788, nicméně pro většinu zájemců je takovéto řešení nedostupné.

Přehled vlastností jednotlivých simulátorů je v (Tabulka 2.2).

	Abacus 787	Project Magenta	Thales 787
Kvalita	-	+	+
Dostupnost	+	+	-
Cena	+	+	-

Označení	Popis
+	Implementace dané vlastnosti je dostačující
-	Implementace dané vlastnosti není dostačující
?	Pro určení není dostatek informací

Tabulka 2.2: Porovnání vlastností existujících implementací

Vzhledem k tomu, že v současné době neexistuje dostupné řešení simulace EFISu B788 jiné než využívání profesionálních simulátorů, můžeme tedy konstatovat, že místo na trhu pro náš EFIS existuje a jeho vývoj má smysl.

2.6 Volba vývojové platformy

V předchozích kapitolách jsme provedli volbu simulátoru, který nám poskytne simulaci vnějšího prostředí a fyzikálního modelu letu. Nyní je třeba zvolit vývojovou platformu pro náš zamýšlený simulátor, tzn. vybrat programovací jazyk a grafické API, které využijeme.

Při volbě programovacího jazyka musíme zohlednit naše požadavky na výsledný simulátor, zejména jeho komunikaci s FSX a komunikaci s periferiemi. FSX a jeho vnější rozhraní podporují jazyky C/C++ a .NET [27]. Vzhledem k tomu, že od programu budeme požadovat komunikaci s hardwarem a budeme používat grafické API DirectX nebo OpenGL, jeví se pro nás jako nejvhodnější varianta využití jazyků C/C++.

Při volbě grafického API nejsme vázáni žádnými požadavky, a tak se vzhledem k zvolenému jazyku C/C++ můžeme rozhodnout mezi dvěma hlavními platformami, a to DirectX a OpenGL. Jelikož naše simulace bude 2D a nejde nám o množství podporovaných efektů, zvolíme jako naši platformu OpenGL z důvodu snadnější implementace simulátoru.

Jelikož náš simulátor bude obsahovat i zvukové výstupy, pro které nestačí pouze použití jednoduchých systémových volání, musíme rovněž zvolit některé API pro zvuk. Vzhledem k volbě programovacího jazyka a rovněž i určitým znalostem jsme zvolili pro zvuk využití rozhraní XAudio2 od firmy Microsoft. Rozhraní je součástí DirectX.

3 Návrh

V této kapitole navrhne strukturu našeho programu, a to jak s ohledem na funkčnost, tak s ohledem na variabilitu a možné budoucí rozšiřování. Rovněž v této kapitole provedeme návrh rozhraní mezi naším simulátorem a FSX, které bude umožňovat vzájemnou komunikaci.

3.1 Funkční požadavky

Funkční požadavky nám specifikují požadavky, které by měl výsledný program splňovat:

1. Simulovat displeje systému EFIS, zejména PFD,ND, CDU – simulace by měla být věrná jak po grafické, tak po funkční stránce
2. Simulovat panely kokpitu potřebné pro interakci se simulátorem při nedostupnosti odpovídajícího hardware
3. Simulovat důležité systémy v letadle – TCAS, EGPWS
4. Podporovat zobrazení na více monitorech – umožnit, aby mohly být různé části EFISu zobrazeny na různých monitorech
5. Komunikovat s FSX jako vizuální platformou
6. Umožnit komunikaci s periferními zařízeními

3.2 Nefunkční požadavky

Nefunkční požadavky vyjadřují omezení, která jsou na aplikaci kladena a požadavky, které nejsou přímo souvislé s její primární funkcí:

1. Aplikace bude vyvinuta pro platformu Windows
2. Aplikace bude koncipována modulárně
3. V co nejmenší míře používat knihovny třetích stran
4. Uchovávat nastavení v konfiguračním souboru

3.3 Využití knihoven

V této části textu rozebereme možnosti využívání knihoven třetích stran, případně některých dalších rozhraní.

3.3.1 Knihovny třetích stran

Vzhledem k tomu, že uvažujeme i o budoucím možném komerčním využití našeho programu, je potřeba při využívání knihoven nebo částí kódů třetích stran mít tento fakt na paměti. U většiny volně dostupných knihoven není možné je začlenit do programu a tento následně komerčně využít, proto se budeme během vývoje snažit eliminovat využívání takovýchto modulů.

3.3.2 WinAPI

Jelikož jsme se rozhodli co možná nejvíce omezit využívání knihoven třetích stran a pokud nechceme sami vytvářet všechny jimi poskytované funkce, musíme najít jiné řešení. Toto řešení nám poskytuje WinAPI, rozhraní operačního systému Windows, pomocí kterého lze eliminovat potřebu využití knihoven, jako je například GLUT [28].

Dalším výrazným místem, kde nám využití rozhraní WinAPI přinese užitek, je práce s písmem. Nebude nutné vytvářet části kódu, které by obstarávaly práci s fonty včetně výpisů textů na obrazovku a bude stačit pouze zavolat potřebné funkce z WinAPI.

Využitím WinAPI však svážeme náš simulátor se systémem Windows, avšak toto navázání pro nás nebude omezením, jelikož FSX lze provozovat výhradně v systému Windows. I kdybychom náš EFIS provozovali na jiném PC než samotné FSX a obě zařízení by spolu komunikovala skrze rozhraní SimConnect, tak bychom byli na systému Windows i nadále závislí, jelikož klientská část rozhraní SimConnect funguje také pouze na operačním systému Windows.

3.4 Struktura programu

Nyní provedeme návrh struktury našeho simulátoru s ohledem na kladené požadavky, jako je budoucí možné rozšíření, tak s ohledem na efektivitu běhu programu.

3.4.1 Samostatný program / Modul

Protože jsme se rozhodli jako platformu pro náš simulátor využít FSX, existují dvě možnosti, jak náš program k FSX připojit. První je vytvoření modulu a druhou je vytvoření samostatné aplikace komunikující s FSX.

Pokud bychom program realizovali jako modul do FSX, musel by zároveň využívat i engine FSX, plánovaný simulátor bude většího rozsahu a využívání pouze prostředků FSX by nás značně omezovalo. Z těchto důvodů program realizujeme jako samostatnou aplikaci. Při vývoji nebudeme nikterak omezeni, ale bude nutné při tvorbě simulátoru vytvořit všechny části a to i včetně funkcí, které bychom v případě tvorby modulu mohli využít. Jedná se zejména o všechny vykreslovací funkce.

3.4.2 Robustní program / modulární (DLL)

Je třeba rovněž rozhodnout, zda celý program koncipovat jako jeden robustní program a všechny jeho části spojit do jednoho modulu, nebo zda program rozdělit na více částí. Spojení do jedné části by přineslo jednodušší kooperaci mezi jednotlivými moduly, oproti tomu rozdělení na jednotlivé moduly přináší větší variabilitu.

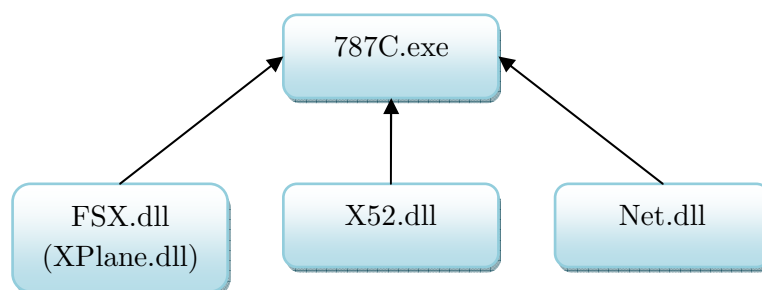
V případě této práce, jelikož se jedná o poměrně rozsáhlý program, provedeme oddělení některých částí od samotné vykreslovací části. Toto rozdělení nám přinese především zpřehlednění samotného programu. Tímto způsobem se vyhneme problémům, které mohou nastat v případě, že některý modul nelze spustit. Pokud by byl program implementován jako robustní program, pak by v této situaci došlo k pádu celého programu a odhalení příčiny by bylo obtížné. Při oddělení jednotlivých modulů je

možné program spustit a případné moduly, u kterých dojde k chybě identifikovat a o tomto informovat uživatele.

Náš program se bude dělit do následujících částí (Obrázek 3.1):

- Vykreslovací část
- Část pro komunikaci s FSX
- Část pro práci s periferními zařízeními
- Část pro síťovou komunikaci

Výhodou oddělení knihovny pro komunikaci s FSX je rovněž i fakt, že pokud by byla implementována v budoucnu knihovna, která by umožňovala i komunikaci s jiným simulátorem, například s XPlane, stačilo by pouze tyto knihovny vyměnit bez toho, aby bylo nutné zasahovat do samotného programu.

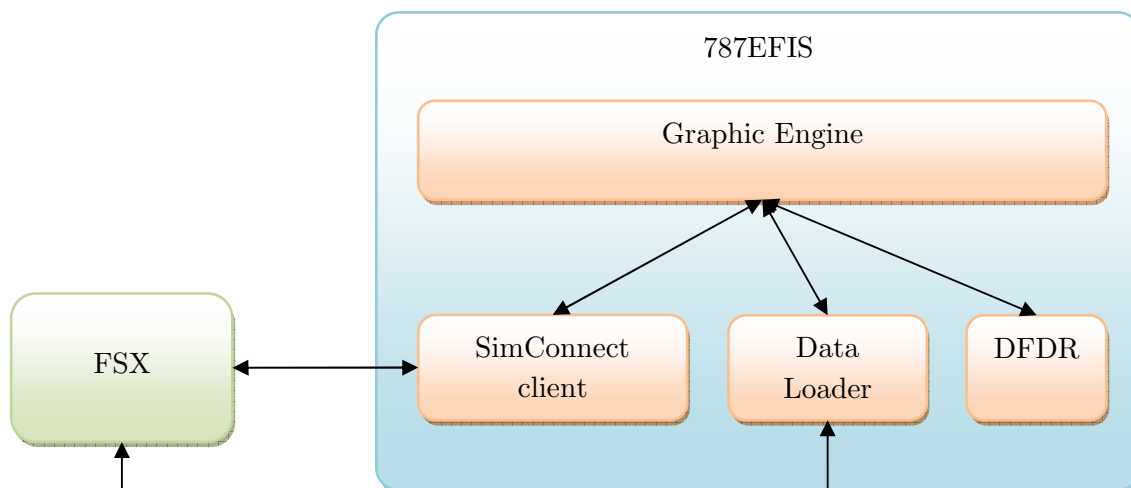


Obrázek 3.1: Struktura programu

3.4.3 Vlákna

Jelikož se program bude skládat z několika modulů, které budou fungovat nezávisle na sobě, je proto nutné zajistit i to, aby se tyto části nebrzdily navzájem mezi sebou, zejména pak aby u částí, které musí fungovat v přesných časových intervalech (DFDR), nedocházelo ke zpoždění. Program bude tedy rozdělen na následující vlákna (Obrázek 3.2):

- Vykreslovací vlákno
- DFDR
- Modul zajišťující komunikaci s FSX
- Modul zajišťující načítání dat z navigační databáze



Obrázek 3.2: Struktura vláken

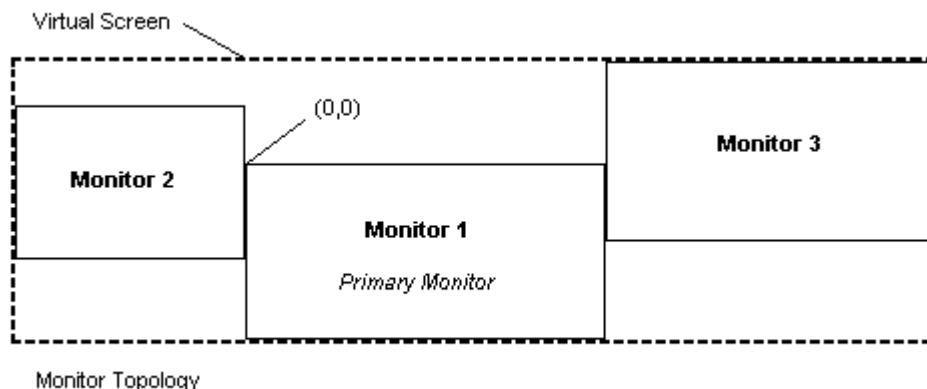
3.4.4 Řešení připojení více monitorů

Jak jsme již popsali v kapitole 2.2 je EFIS Boeingu 787-8 tvořen pěti monitory, tedy i naše simulace bude koncipována pro zobrazení na pěti, případně i více monitorech. V dnešní době existují dvě základní možnosti pro připojení více monitorů k jednomu PC. První možností je připojení více monitorů k jedné grafické kartě, avšak většina grafických karet umožňuje současné připojení maximálně dvou monitorů. Je tudíž nutné použít takovou grafickou kartu, která umožňuje připojení více jak dvou monitorů. Jde například o karty disponující technologií Eyefinity nebo profesionální karty od firmy Matrox. Druhou možností je využít v jednom systému více nezávislých grafických karet. Náš simulátor by měl být připraven na obě tyto varianty, jelikož použití více monitorů je pro simulaci EFISu zásadní.

Soustředíme se nyní zejména na návrh multimonitorového zobrazení na jednom PC. Při renderování obsahu z jednoho PC existují dvě základní varianty. První možností je připojení všech monitorů k jedné grafické kartě disponující více výstupy. Běžně je tato varianta omezena na dva současně připojené monitory, ale existují i výjimky jako například již zmíněné karty podporující technologii Eyefinity, nebo různé specializované karty. Druhou variantou je použití více grafických karet v jednom PC. Tuto variantu je potřeba zvolit tehdy, pokud není k dispozici grafická karta s požadovaným počtem současně využitelných výstupů anebo je-li nedostatečný grafický výkon jedné karty. Pokud chceme, aby náš program umožňoval použití obou popsaných konfigurací, musíme tento požadavek zohlednit již v návrhu naší aplikace. Další požadavek, který musíme vzít v potaz, je maximální variabilita, tedy možnost uživatele systému si jednoduše nastavit, jaké části simulace budou zobrazeny v konkrétních oblastech jednotlivých monitorů.

Při renderování obsahu na více monitorů současně existují v zásadě dvě základní varianty postupu. Buďto je možné vytvořit okno přes celou virtuální plochu (Obrázek 3.3) a podle souřadnic monitorů ve virtuální ploše vykreslovat jednotlivé požadované části. Tato metoda je z pohledu programátora nejjednodušší, jelikož je vytvořeno pouze

jedno okno, do kterého je kreslen veškerý obsah, nicméně nelze tuto variantu použít ve všech situacích. První problém nastává s využitím hardwarové akcelerace, jelikož při použití jediného okna přes více monitorů může nastat situace, že hardwarová akcelerace bude dostupná pouze na části okna zobrazené na primárním monitoru, na ostatních však již nikoliv. Další problémy nastávají, pokud monitor není součástí virtuální plochy, anebo když jsou různé části okna renderovány různými grafickými kartami.

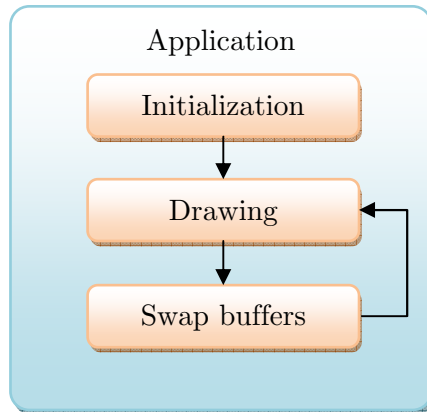


Obrázek 3.3: Virtuální plocha [29]

Druhou variantou renderování na více monitorů současně je vytvoření nezávislého okna pro každý monitor zvlášť. Pomocí této metody lze zajistit, že každé naše okno bude disponovat hardwarovou akcelerací nezávisle na tom, v jaké konfiguraci máme grafické karty a monitory zapojeny. Avšak nastává zde jiný problém, a to nutnost sdílení obsahu grafické paměti mezi jednotlivými okny, aby nebylo nutné načítat data do grafické paměti zvlášť pro všechna okna, a tím i dosahovat značné redundance a zbytečné časové prodlevy.

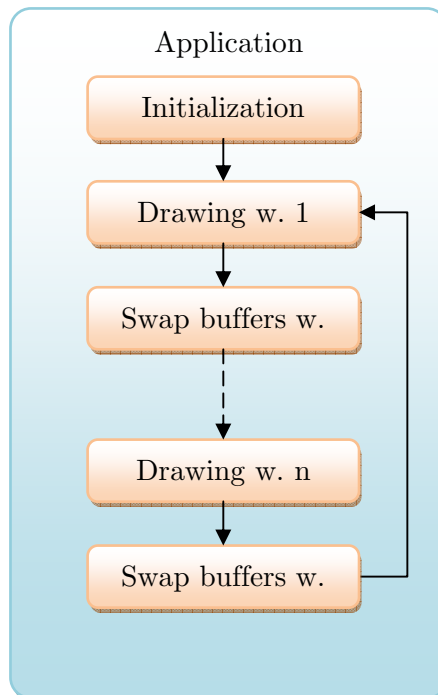
Vzhledem k tomu, že vyžadujeme hardwarovou akceleraci ve všech částech obrazovky a také umožnění použití konfigurace s více nezávislými grafickými kartami, využijeme pro náš program druhou variantu, tedy vytvoření nezávislého okna pro každý monitor.

Když jsme se rozhodli, jak bude naše aplikace vytvářet okna pro renderování obsahu, je potřeba navrhnout, jak bude fungovat samotná vykreslovací smyčka. U jednoduchých programů funguje vykreslování v jedné nekonečné smyčce (Obrázek 3.4), kde je stále opakovaně vykreslován veškerý obsah.



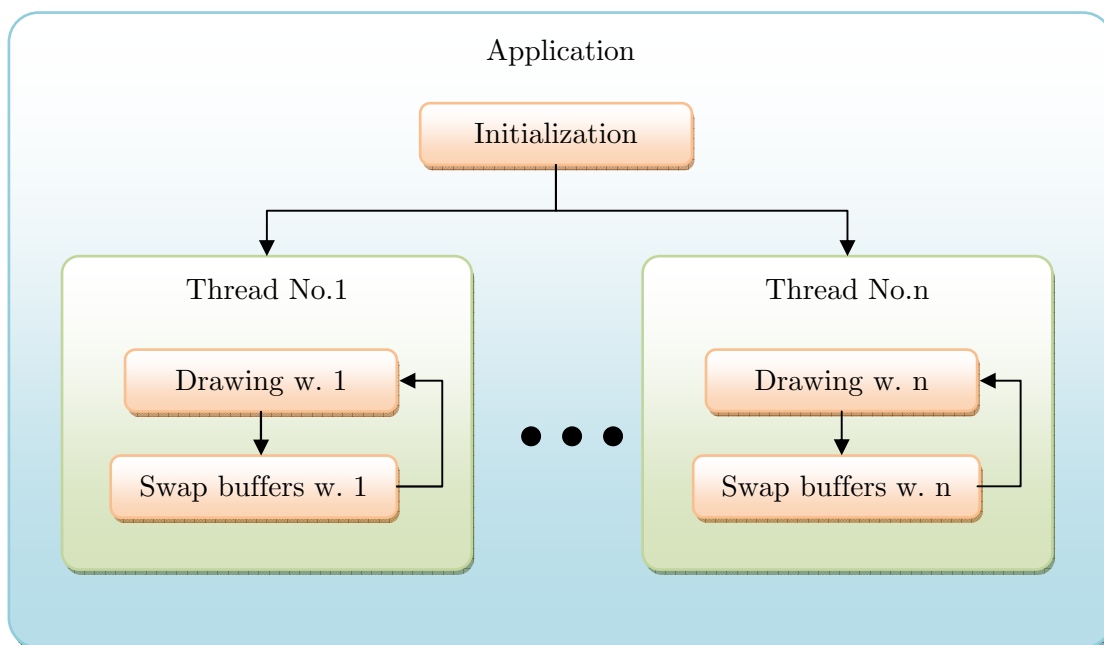
Obrázek 3.4 Jednoduchá vykreslovací smyčka

Jelikož ale náš program má vykreslovat nezávisle do více oken, je zapotřebí navrhnout vykreslovací smyčku, aby tento požadavek splňovala a zároveň fungovala co možná nejefektivněji. Naivní řešení tohoto problému by bylo pouze upravit klasickou smyčku tak, že by vykreslovala v každém cyklu postupně obsah jednotlivých oken (Obrázek 3.5). Avšak tento princip je značně neefektivní. Protože dochází k postupnému vykreslování obsahu jednotlivých oken, probíhají vždy všechny potřebné výpočty pro vykreslení jednoho snímku pouze u jednoho okna. V době vykreslování snímku grafickou kartou by však mohlo docházet například k výpočtům na CPU potřebným k vykreslení jiného okna. Tento problém je ještě markantnější, pokud máme k dispozici PC s více GPU, a tedy pokud dochází k vykreslování pomocí jedné grafické karty, je druhá karta naprosto nečinná.



Obrázek 3.5 Postupné vykreslování v jedné smyčce

Lépe by tedy bylo, aby pro každé okno bylo k dispozici vlastní vlákno, které by obsahovalo vykreslovací smyčku pro dané okno, a tak by vykreslování nebylo bržděno čekáním na zpracování ostatních oken. Máme-li k dispozici více GPU nebo CPU jader, je toto řešení výrazně efektivnější než naivní způsob (Obrázek 3.6).

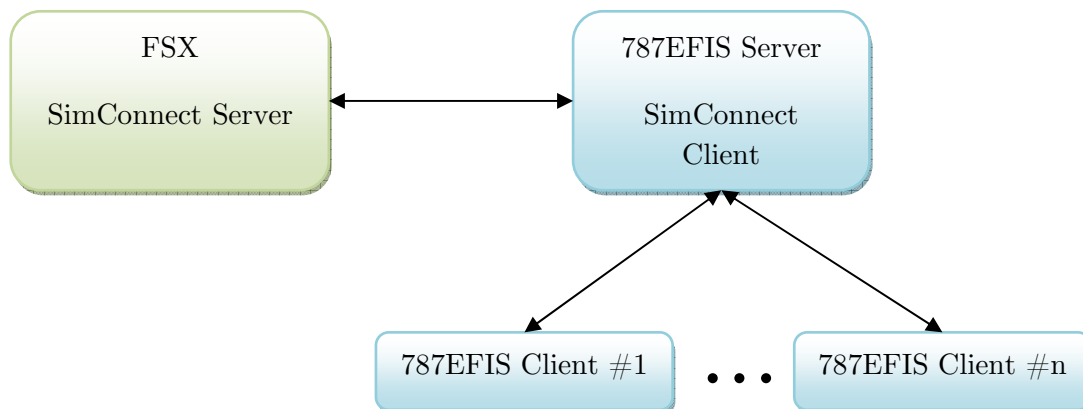


Obrázek 3.6: Vykreslování pomocí více vláken

Pro náš EFIS tedy zvolíme druhou variantu s vykreslovací smyčkou zvlášť pro každé okno. Požadavek na variabilitu nastavení uspokojíme tak, že veškeré nastavení pozic oken a jejich renderovaný obsah uložíme do konfiguračního souboru, kde bude snadno k dispozici uživatelům pro provedení změn podle jejich požadavků.

3.4.5 Klient-server architektura

Protože ne vždy je možné použít PC umožňující připojení více monitorů, je vhodné u programu implementovat i možnost plnohodnotného provozování i bez hardware umožňujícího připojení více monitorů k jednomu počítači. Řešením takovéto situace je využití architektury klient-server (Obrázek 3.7), kde by náš simulátor byl schopen běžet na více PC současně s tím, že na každém PC by byly simulovány různé systémy. Vzájemná komunikace jednotlivých částí bude zajištěna pomocí počítačové sítě.



Obrázek 3.7: Architektura klient-server

3.5 Možnosti rozhraní s FSX

Jak plyne z předchozích kapitol, bude program pro svůj chod potřebovat značné množství údajů o průběhu simulace získaných z FSX. Tyto údaje se dají rozdělit do tří hlavních kategorií. Na údaje, u nichž je pro funkci simulátoru kritická přesnost a tedy jejich častá aktualizace. Do této kategorie spadají například údaje, jako je pozice letadla, náklon a podobně. Jsou to tedy údaje, u kterých během letu dochází k častým změnám a pro piloty je důležitá jejich korektnost a přesnost. Do druhé kategorie spadají údaje, u kterých během simulace nedochází k rychlým změnám, a tedy není potřeba je obnovovat s takovou frekvencí jako údaje z první kategorie. Do této skupiny patří například informace o počasí, pozice ostatních letadel v oblasti a podobně. U některých údajů během simulace dokonce nedochází ke změnám vůbec, stačí je tedy získat pouze jednou. Jsou to například informace o terénu, informace z navigační databáze a podobně.

Při návrhu komunikace mezi simulátorem a FSX je potřeba tedy zohlednit požadavky na přesnost údajů. K jejich získávání je možné využít různé metody, které jsou dnes k dispozici. Některé metody jsou přímo podporovány společností Microsoft a na jejich funkci se lze spolehnout. U jiných metod se jedná o neoficiální postupy, případně programy třetích stran, z čehož plyne i za určitých okolností nižší spolehlivost. Pro vytvoření rozhraní mezi naším simulátorem a FSX je tedy potřeba zvolit vhodnou metodu, případně jejich kombinaci, k zajištění požadovaných funkcí.

3.5.1 SimConnect

SimConnect [27] je součástí Microsoft ESP. Jedná se o rozhraní umožňující komunikaci mezi ESP a vytvořenými doplňky (add-ony). SimConnect funguje na principu klient-server, tedy program na platformě ESP, v tomto případě FSX tvoří server a komunikuje s klienty. Jelikož je SimConnect oficiální součástí FSX, umožňuje proto spolehlivě získávat potřebné údaje.

Z dokumentace [30] je patrné, že pomocí rozhraní SimConnect je poměrně snadné získat značné množství dat o průběhu simulace. Jsou to především o informace o stavu

letadla, dále pak některé informace o prostředí, jako údaje o počasí nebo o ostatních objektech v simulaci.

Údaje z kategorie časově kritických umožňuje rozhraní dodávat až se stejnou frekvencí, jako je snímkovací frekvence běhu simulátoru. Klientský program potažmo pilot má tudíž stále k dispozici aktuální data.

Podle [27] je sice možné získat také informace o pozici mraků v okolí letadla (*SimConnect_WeatherRequestCloudState*), což jsou informace důležité pro funkci systému WXR. Při využívání této funkcionality se ale vyskytuje problém, že získaná data nekorespondují se skutečnou situací, a tedy není možné tuto funkci bez dalších úprav použít a ani tak simulovat WXR. Z tohoto vyplývá, že ač se jedná o oficiální rozhraní pro ESP, nelze se na jeho funkčnost plně spolehnout.

Naopak některá data s informacemi o prostředí, jako jsou například informace z navigační databáze nebo informace o letištích, je sice možné pomocí SimConnectu získat, avšak vždy pouze z bezprostředního okolí letadla, což je pro některé simulované systémy zejména FMC nedostačující.

Některé údaje dokonce nelze získat vůbec. Jsou to zejména informace o terénu, jako jeho reliéf nebo povrch.

3.5.2 FSUIPC / WideFS

Flight Simulator Universal Inter-Process Communication (FSUIPC) je modul vytvořený Peterem Dowsonem [31] pro komunikaci mezi MSFS a ostatními doplňky. FSUIPC bylo vytvořeno původně pro MS FS98, jelikož v době jeho vzniku až do verze FS2004 FS neobsahoval žádné rozhraní, které by vývojářům umožnilo přistupovat k různým informacím o letu, a tedy nebylo proto možno vytvářet komplexní doplňky (add-ons), které by vyžadovaly pro svoji funkci přístup k těmto datům.

V současnosti existují dvě verze FSUIPC3 a FSUIPC4. Verze FSUIPC3 je určena pro FS2004 a získává, potažmo předává data přímo procesu Flight Simulátoru, jelikož, jak již bylo zmíněno, do verze FS2004 simulátor neobsahoval žádné rozhraní pro jejich poskytování. Verze FSUIPC4 je určena pro komunikaci s FSX a ke své funkci používá rozhraní SimConnect zmiňované v předchozí kapitole [32]. Protože je modul FSUIPC4 založen výhradně na komunikaci skrze rozhraní SimConnect, je tedy možné, pomocí něj získat stejné informace jako při použití přímo rozhraní SimConnect. Rovněž tedy FSUIPC4 přejímá i problémy ohledně práce s počasím v FS [32], jak bylo popsáno v předešlé kapitole.

Samotný modul FSUIPC je schopen pracovat pouze s doplňky, které jsou vytvořeny přímo jako moduly samotného FSX, takzvané moduly (plug-ins). Aby bylo možno komunikovat skrze FSUIPC i z ostatních aplikací nebo prostřednictvím sítě, je potřeba samotné FSUIPC doplnit o modul WideFS, který obsahuje server a klienta a je pomocí něho možné komunikovat s FSX i skrze síť, podobně jak tomu je u rozhraní SimConnect [33].

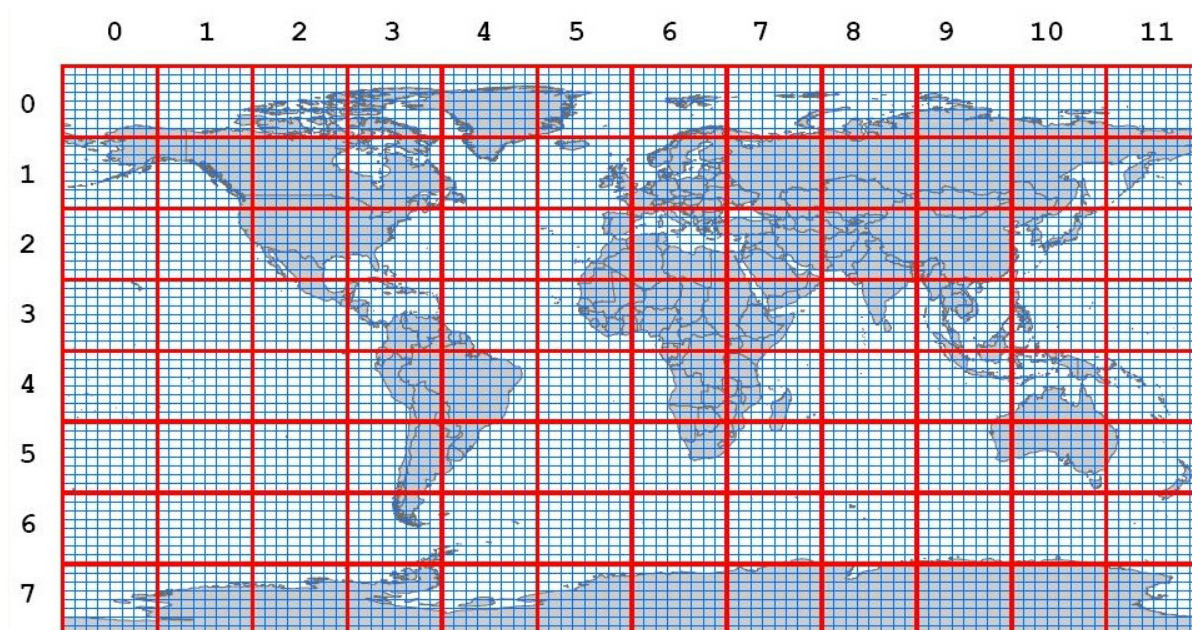
Jelikož se u FSUIPC / WideFS jedná o komerční software, jeho využití není zcela zdarma. Základní verzi modulu FSUIPC lze pořídit bez úhrady. Existuje i placená verze v ceně €24, která oproti volné verzi přináší některé benefity, jako například rozšířená kalibrace joysticku, možnost uložení stavu letadla během simulace a další. Žádná s těchto doplňkových funkcí nepřináší užitek vytvářenému simulátoru EFISu. WideFS existuje pouze v jedné, a to placené verzi v ceně €24. Obě součásti tedy FSUIPC i WideFS lze v placených verzích pořídit dohromady za €36. [32]

Výhoda ve využití FSUIPC je pro vývojáře v tom, že pokud již vyvinuli některý svůj modul pro starší verzi FS a využili FSUIPC, tak jej při přechodu k FSX nemusejí upravovat a mohou modul rovnou použít s novou verzí FSUIPC.

3.5.3 Dekompilace datových souborů

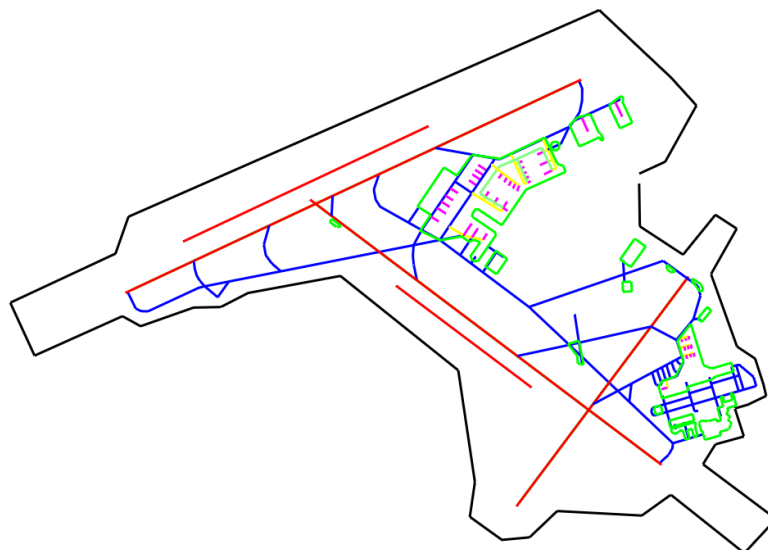
V předchozích kapitolách jsme si představili rozhraní SimConnect a FSUIPC, pomocí kterých lze z FSX poměrně snadno získat značné množství dat o průběhu simulace, zejména tedy časově kritických informací o průběhu letu jako jsou informace o poloze, náklonu a podobně. Jak již bylo řečeno, nelze pomocí rozhraní SimConnect či FSUIPC získat dostatečné množství informací o terénu, které jsou potřebné pro simulaci systémů, jako je EGPWS, VSD a další. Je proto zapotřebí zajistit tyto data jinou cestou. Podobně, jak je tomu u informací o terénu, rovněž i množství informací z navigační databáze FSX získaných pomocí rozhraní SimConnect/FSUIPC je značně omezené vždy jen na aktuální okolí letadla, a tedy i zde je potřeba data získat jinou cestou.

Jednou z možností, jak tyto údaje získat, je jejich načtení přímo z datových souborů FSX. FSX obsahuje velké množství souborů *.BGL, které obsahují informace o letištích, navigačních bodech, vektorové informace jako jsou cesty a mnoho dalších. Jednotlivé BGL soubory vždy obsahují informace z jedné zeměpisné lokace, kterou lze identifikovat z názvu souboru pomocí mřížky QMID (Quad Mesh Identifier) (Obrázek 3.8) [34].

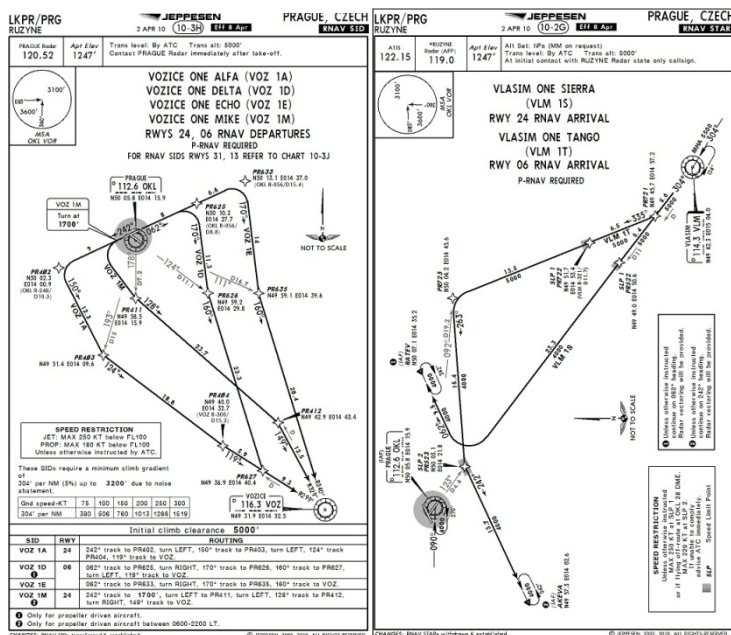


Obrázek 3.8: Quad Mesh Identifier (Červeně QMID úrovně 4, modře QMID úrovně 7)
[34]

Z jednotlivých souborů lze pak díky práci Winfrieda Orthmanna, který analyzoval a zveřejnil strukturu datových souborů .BGL [35], získat podrobné informace o letištích (Obrázek 3.9) a také informace z navigační databáze FSX, jako jsou pozice a frekvence radiomajáků a podobně (Obrázek 3.10).



Obrázek 3.9 Mapa letiště získaná z BGL souborů



Obrázek 3.12: (a)-SID LKPR (b)-STAR LKPR [36]

Jelikož cílovou skupinou uživatelů jsou lidé, kteří se o létání zajímají na vysoké úrovni a provozují lety i v online komunitě, nestačí tedy pouze získat navigační databázi včetně SID/STAR procedur, ale rovněž je vyžadována její aktuálnost, ať už kvůli reálnější simulaci, nebo aby při online letech měli všichni účastníci k dispozici stejná data.

Společnost Navigraph [37] poskytuje navigační databázi včetně SID/STAR procedur zvanou AIRAC pro velké množství různých doplňků (add-onů) do FS. Rovněž jsou poskytovaná data aktualizována v pravidelných měsíčních intervalech. Od roku 2006, kdy byla vydána poslední bezplatná verze, se zde jedná opět o komerční produkt a je nutno novou verzi AIRACu hradit při každé aktualizaci. Nejnižší cena, za kterou lze AIRAC pořídit, je €10.

3.5.5 Výběr řešení

V předcházejících kapitolách jsme si představili různé oficiální i neoficiální možnosti, jak získat z FSX či jiných zdrojů potřebná data, a je tedy nutné zvolit některé řešení. Pro získání dat o průběhu aktuálního letu v FSX máme dvě možnosti, buďto použít rozhraní SimConnect nebo FSUIPC/WideFS. Jak bylo zmíněno v kapitole 3.3.2, modul FSUIPC4, který je určen pro FSX, je postaven nad rozhraním SimConnect a přejímá tak jeho klady i zápory. Jelikož se v této práci zabýváme vytvořením zcela nového projektu, který bude spolupracovat výhradně s FSX, nelze brát ani jako výhodu to, že při použití FSUIPC by přechod mezi jednotlivými verzemi byl snadný. Naopak je třeba zohlednit fakt, že náš program a FSX budou provozovány nejspíše na nezávislých počítačích, komunikace mezi nimi bude tak probíhat po síti a při použití FSUIPC by bylo zapotřebí i WideFS, což je na rozdíl od SimConnectu, který je přímo součástí FSX, placená služba. Vzhledem k těmto okolnostem bylo rozhodnuto, že komunikace mezi naším programem a FSX bude probíhat výhradně pomocí rozhraní SimConnect.

Jelikož skrze rozhraní SimConnect nejsme schopni získat veškeré potřebné informace, je tedy nutné použít i další metody pro získání potřebných dat. Využijeme tedy dekompilaci souborů .BGL, ze kterých lze, jak již bylo popsáno v kapitole 3.3.3, získat informace o letištích a dále pozice a vlastnosti jednotlivých navigačních bodů pro navigační databázi. Pokud jde o využití AIRAC dat, došli jsme k závěru, že do ukončení vývoje není potřeba disponovat aktuálními daty pro navigační databázi a že z AIRACu budou využity pouze SID/STAR procedury, s tím že do programu bude dodatečně implementována možnost načíst kompletní AIRAC, aby měli uživatelé k dispozici data aktuální.

3.6 Návrh FMC

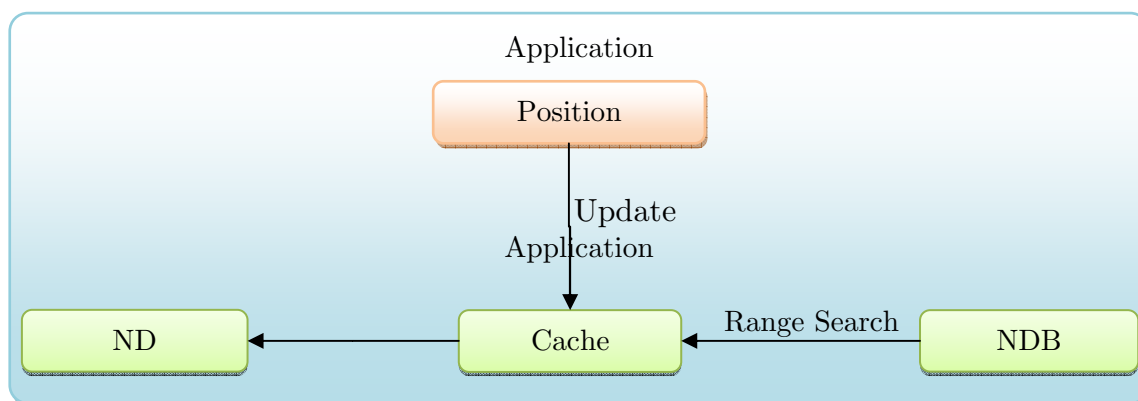
Jak již bylo zmíněno výše, tak jedním z nejdůležitějších systémů v letadle je FMC, neboli palubní počítač, a proto i v naší simulaci je funkce FMC zásadní. Boeing 787-8 obsahuje tři FMC, z nichž vždy jedno funguje jako master, druhé jako náhradní (spare) a třetí jako záložní (backup). FMC se stará o tzv. Thrust Management, tedy o výpočty tahu motorů v jednotlivých fázích letu a příkazy pro systém automatického řízení tahu, rovněž o monitoring stavu paliva během letu. Další důležitou součástí každého FMC jsou tři databáze. První je navigační databáze (NDB), druhou databází, která je obsažená v FMC je databáze obsahující informace o výkonových charakteristikách letadla, jako je například maximální nebo optimální rychlost. Poslední DB obsahuje specifická data pro jednotlivé provozovatele. Nyní si popíšeme návrh simulace některých částí FMC.

3.6.1 Návrh navigační databáze

Navigační databáze obsahuje informace o všech navigačních bodech, informace o jednotlivých letištích včetně údajů o runwayích, dále pak obsahuje procedury pro odlet a pro přiblížení a převodní výšky (transition altitude). Z tohoto plyne, že v NDB je obsaženo velké množství dat, řádově se jedná o stovky tisíc záznamů a je nutné při jejím návrhu tento fakt zohlednit.

Při návrhu NDB je potřeba zvážit, jaké na ni budeme klást požadavky. V našem případě to budou dva základní typy přístupů. Jedním bude dotaz na konkrétní údaj v databázi, druhým pak dotaz na objekty z NDB nacházející se v daném okolí. První kategorie dotazů bude využívána zejména při zadávání údajů do CDU, a tedy lze říci, že dotazy tohoto druhu nebudou příliš četné a tudíž nebudeme klást velké požadavky na rychlost jejich vyhodnocení. Naopak u dotazů z druhé kategorie, které budou využity například k vykreslování údajů na ND, budeme vyžadovat aktuální údaje během každé vykreslovací smyčky, z čehož plyne, že zde klademe vysoké požadavky na efektivitu zpracování dotazu v NDB.

Po vyhodnocení požadavků kladených na NDB bylo rozhodnuto, že realizace prvního druhu požadavků, jelikož u něho nejsou kladeny vysoké nároky na rychlost, může být provedena jednoduchým sekvenčním prohledáním celé NDB. Oproti tomu u druhé kategorie, kde je rychlost zásadní, bylo rozhodnuto o použití algoritmu Exact Circular Range Search dle [38]. Tento algoritmus bude pak v zájmu zvýšení efektivity doplněn o mezipaměť na výsledky (Obrázek 3.13).



Obrázek 3.13: Vyhledávání v NDB

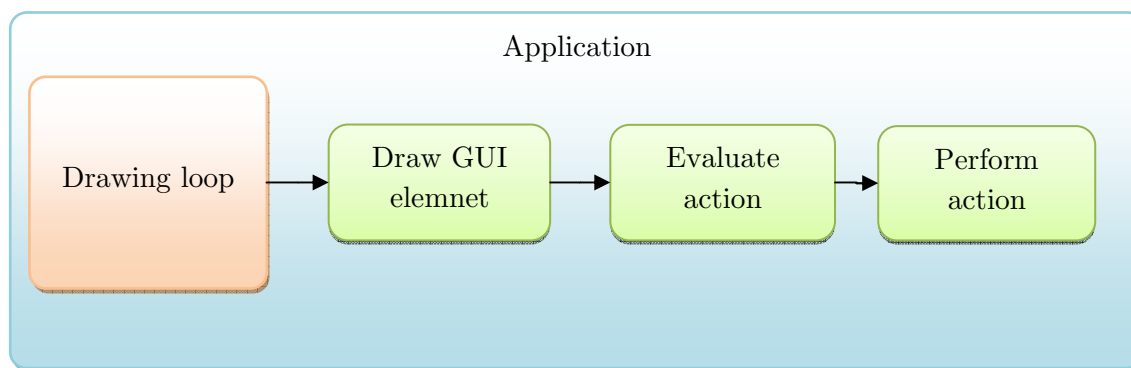
Využitím kombinace těchto metod bude dosaženo, jak zvýšení rychlosti zpracování dotazu, tak i omezení počtu potřebných dotazů. Tohoto je dosaženo tím, že při změně polohy středu vyhledávání jsou pomocí metody range search vyhledány požadované výsledky, které jsou následně uloženy do mezipaměti. Další vyhledávání však nenastane okamžitě po změně pozice, nýbrž až při posunu o předem stanovenou vzdálenost a během této doby jsou používány pouze výsledky z mezipaměti. Po překročení stanovené hranice vzdálenosti od původního vyhledávání jsou opět pomocí metody range search vyhledány nové výsledky, a data v mezipaměti jsou aktualizována. Takto tedy dojde, jak k urychlení samotného vyhledávání využitím range search, tak i k omezení počtu hledání, a to díky využití mezipaměti.

3.7 Návrh uživatelského rozhraní

Důležitou součástí naší simulace je vytvoření interaktivních prvků uživatelského rozhraní. Naše aplikace bude obsahovat tři základní druhy uživatelského rozhraní.

1. Rozhraní, které je simulací GUI z EFISu B788 a jeho funkce bude tedy stejná jako u simulovaného systému.
2. Prvky, které simulují jednotlivé panely letounu. Tyto prvky jsou vizuální reprezentací skutečných fyzických prvků v letadle, avšak pro nedostupnost adekvátního hardwaru jsou pouze simulovány.
3. Prvky, které nebudou součástí samotné simulace letadla, nýbrž se bude jednat o GUI, které bude uživatelům našeho systému umožňovat provádět jeho konfiguraci podle jejich požadavků.

U všech těchto skupin pak budou muset být splněny základní požadavky, a to na přesnost zpracování, jak po grafické stránce, tak po stránce funkční, čímž se rozumí odezva na jednotlivé události. U prvků GUI z druhé kategorie je pak možno provést doplnění o zvukový doprovod pro nahrazení odezvy při používání skutečných hardwarových panelů. Návrh funkce GUI je znázorněn na (Obrázek 3.14). U každého prvku GUI je přímo během vykreslování vyhodnoceno, zdali má být provedena nějaká reakce na událost.



Obrázek 3.14 Návrh funkce prvků GUI

3.8 Návrh rozhraní mezi EFISem a Flight Simulátorem X

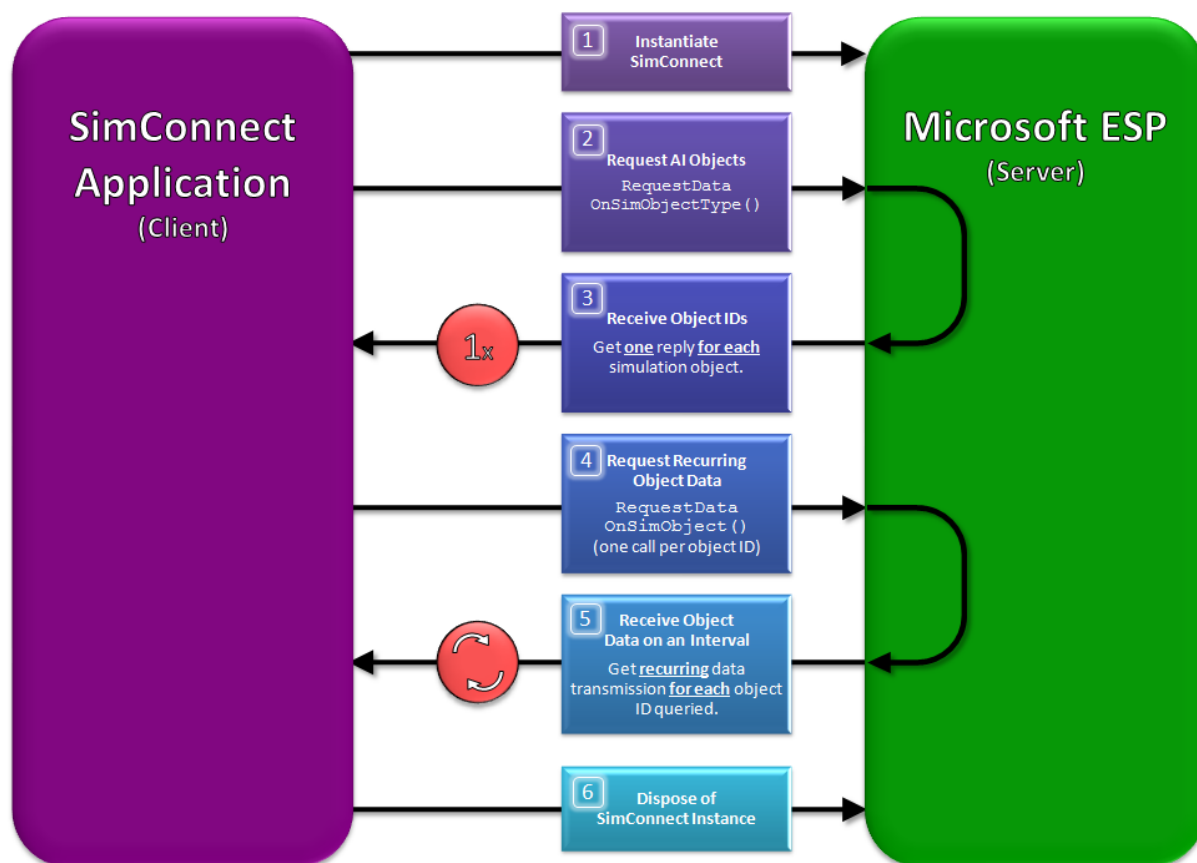
Jak jsme již zmiňovali v předchozích kapitolách, pro zjišťování časově kritických dat využijeme rozhraní SimConnect. V našem simulátoru je toto rozhraní realizováno pomocí nezávislého modulu FSX.dll. Veškerá komunikace mezi naším programem a FSX by měla probíhat nezávisle na vykreslování.

Získání informací o stavu našeho letadla jako je pozice, výška, stav motorů a podobně se realizuje snadno pomocí zaslání požadavku na data do FSX. Data jsou pak získávána periodicky s požadovanou frekvencí.

Tento princip však nelze aplikovat na všechny naše požadavky, jelikož přes rozhraní SimConnect nelze získat všechna požadovaná data, nebo je nelze získat takto přímočarým postupem. Mezi tyto požadavky patří zejména požadavky na data pro systémy TCAS, WXR, VSD, terénní radar a podobně.

3.8.1 TCAS

Pro realizaci systému TCAS je zapotřebí získat informace o ostatních letadlech v oblasti, a to zejména o jejich poloze a výšce. Tyto údaje lze získat pomocí rozhraní SimConnect metodou, jak je popsáno v [39] (Obrázek 3.15).



Obrázek 3.15: TCAS [39]

Nejprve je zaslán požadavek pro zjištění všech objektů v simulaci. Po získání těchto dat jsou pro jednotlivé objekty vyžádány informace o jejich pozici a výšce. Tyto informace jsou pak periodicky požadovány, aby byly stále aktuální. Dále jsou sledovány události, kdy některý z objektů opustí simulaci, nebo když do simulace nový objekt vstoupí, abychom stále sledovali všechny relevantní objekty.

3.8.2 Meteorologický radar

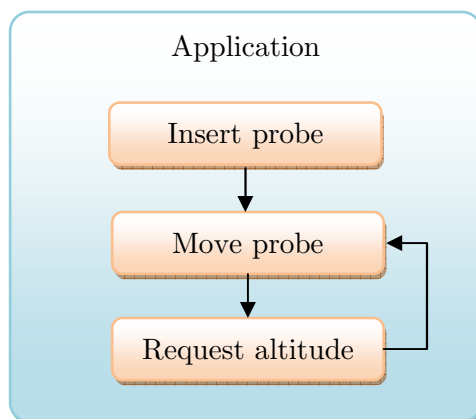
Pro realizaci simulace meteorologického radaru je potřeba získat data o hustotě srážek v okolí letadla. Tento druh informací ale API FSX nenabízí. Je zde pouze k dispozici funkce pro zjištění hustoty mraků v okolí letadla, nicméně tato funkce, jak jsme již zmínili v kapitole analýza, nevrací korektní výsledky. Proto v současné době neexistuje možnost jak simulovat funkci meteorologického radaru, tak aby zobrazovaný výstup korespondoval s aktuálním počasím v FSX.

3.8.3 Vertical situation display

Pro VSD je potřeba získat informace o reliéfu terénu před letadlem. Jak jsme již zmínili během analýzy, skrze rozhraní SimConnect nelze získávat informace o terénu v požadovaných místech. Lze však použít podobný princip, jaký je využit pro simulaci systému TCAS, a tedy faktu, že lze zjistit výšku jiných objektů v simulaci. V našem programu tedy budeme postupovat tak, že nejprve do FSX vložíme objekt. Bude se jednat o neviditelný objekt, který bude fungovat jako sonda. Pozici námi vloženého

3 NÁVRH

objektu pak budeme postupně posouvat v ose letadla a při každé změně pozice odečteme výšku terénu. Získané informace využijeme k simulaci VSD (Obrázek 3.16).



Obrázek 3.16: Použití sondy pro získání dat

3.8.4 Terénní radar

Pro vytvoření terénního radaru je potřeba získat množství informací o terénu v okolí letadla. Pro jeho realizaci lze použít stejný princip jako při simulaci VSD. Nevýhodou tohoto postupu je však velká náročnost na komunikaci mezi EFISem a FSX a bez dalších optimalizací metody ji nelze použít. V tomto případě můžeme užít, podobně jako při realizaci NDB, mezipaměť, do které bychom si ukládali již jednou načtená data. Rovněž lze tuto mezipaměť uložit i ve formě souboru na pevný disk, aby nebylo nutné v budoucnu data opět získávat skrze rozhraní SimConnect z FSX. Z této mezipaměti by mohla být poskytována i data pro VSD.

3.8.5 BGL dekompilace

Data, která jsme si označili jako časově nekritická, jsme se rozhodli získat pomocí dekompilace souborů BGL. Pro tuto dekompilaci použijeme, do dokončení zcela nového dekompilátoru, upravenou verzi programu BGL2XML [40].

4 Implementace

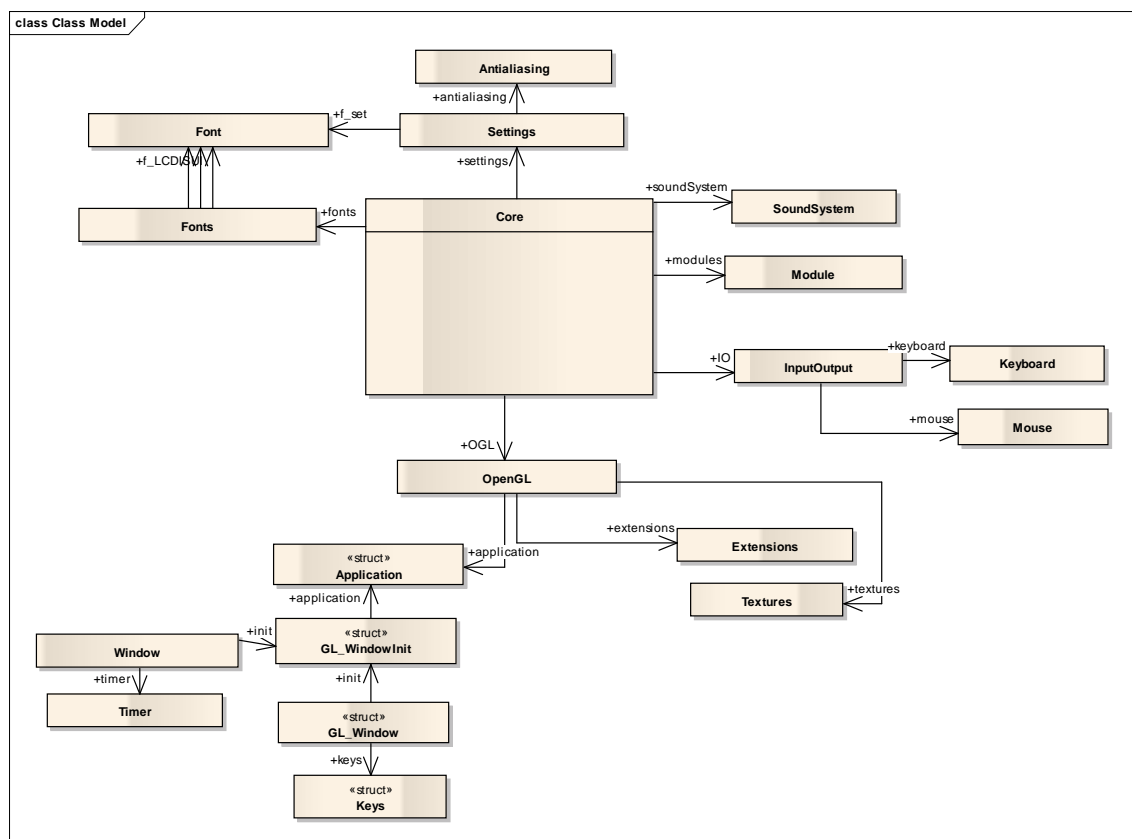
V této kapitole si popíšeme základní rysy implementace našeho systému. Jako vývojové prostředí bylo zvoleno Microsoft Visual Studio 2008 [41]. Toto IDE bylo zvoleno zejména pro značné zkušenosti s vývojem v tomto prostředí. Nyní si popíšeme strukturu programu, poté se zaměříme na některé jeho části podrobněji.

4.1 Struktura programu

Základní modul programu je rozdělen do dvou částí. Jádra reprezentovaného globální instancí třídy Core a simulací letadla reprezentované globální instancí třídy Aircraft.

4.1.1 Třída Core

Třída Core reprezentuje jádro programu. Obsahuje všechny funkce starající se komunikaci s operačním systémem. Třída se stará také o veškerou práci s vlákny a o všechny inicializace a nastavení. Částečný class diagram je na (Obrázek 4.1).



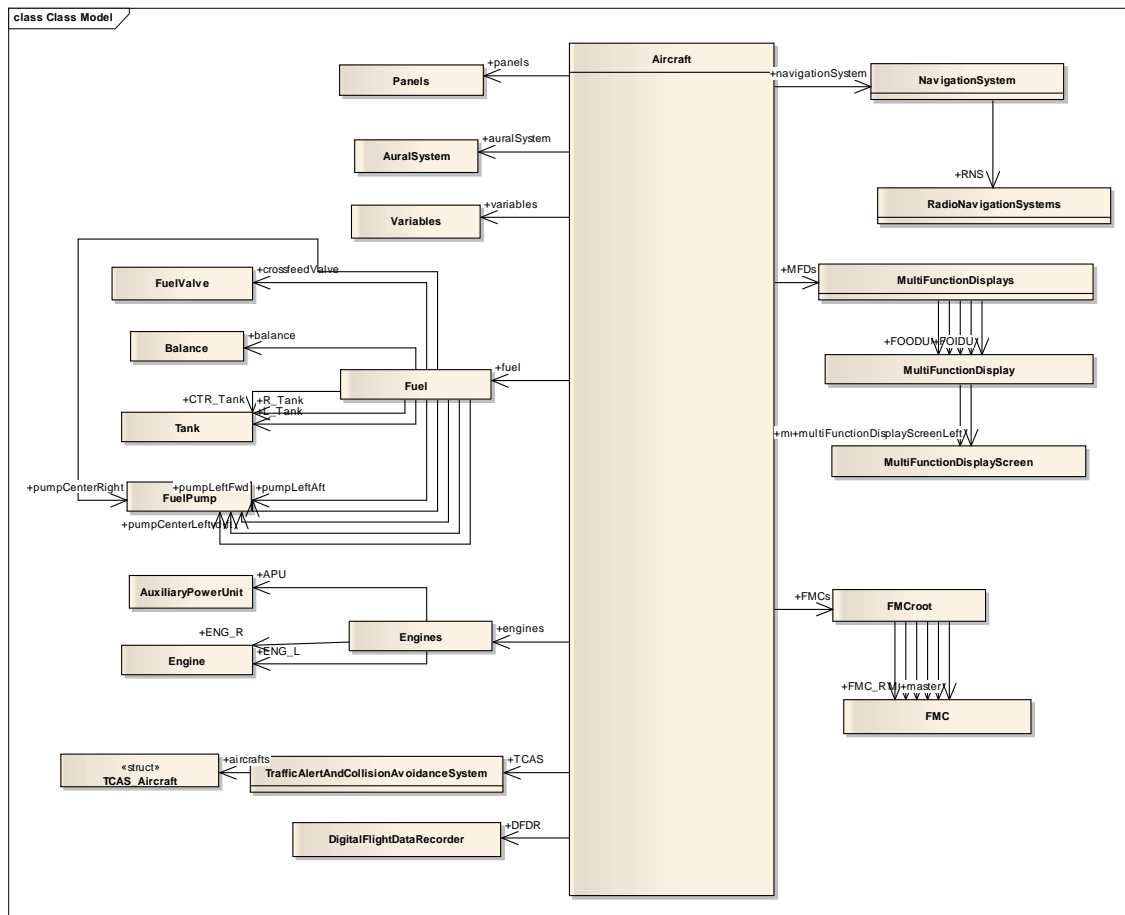
Obrázek 4.1 Třída Core

4.1.2 Třída Aircraft

Třída Aircraft je reprezentací samotného letadla. Obsahuje veškeré vykreslovací funkce pro všechny displeje a panely. Dále jsou zde funkce starající se o simulaci ostatních systémů letadla jako je FMC a podobně. Částečný class diagram je na (Obrázek 4.2).

4 IMPLEMENTACE

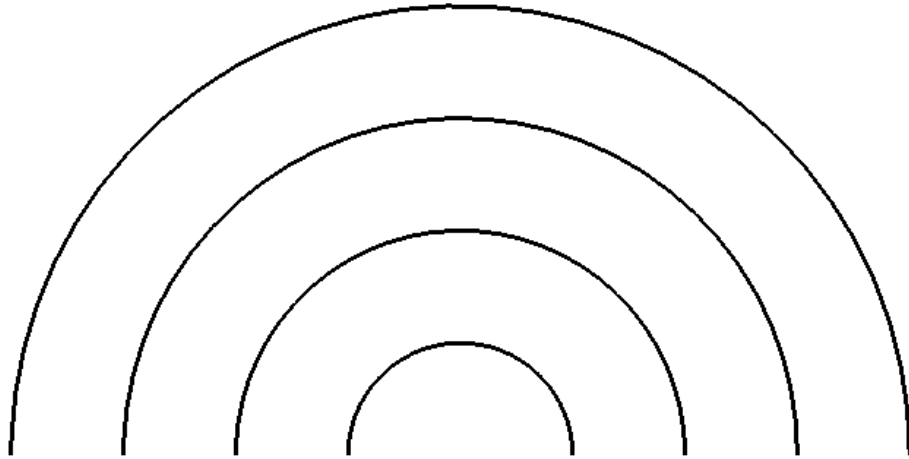
Třidu využívají další moduly pro přístup ke všem stavovým proměnným, které jsou uloženy ve členské proměnné variables. Jejich soupis je v Příloze E.



Obrázek 4.2 Třída Aircraft

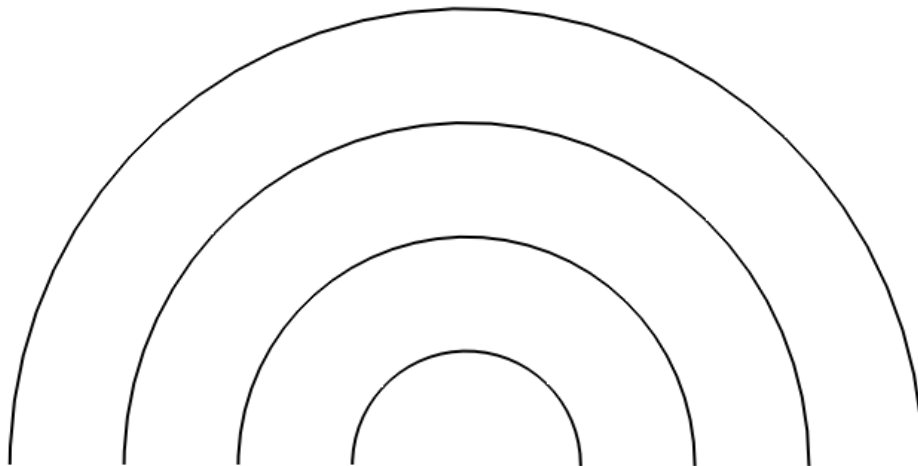
4.2 Antialiasing

Jelikož pro vykreslování většiny částí EFISu používáme grafická primitiva, jako jsou především čáry, je potřeba řešit problémy, které nám toto přináší. Jde zejména o aliasing, tedy o ostré přechody mezi vykreslovanými primitivami a pozadím. Tento efekt vzniká tím, že je čára vykreslena jako jednotlivé pixely. Tento fakt nepůsobí potíže, pokud čára, kterou kreslíme je kolmá nebo vodorovná. Pokud kreslíme například oblouk, je tento jev markantní (Obrázek 4.3).



Obrázek 4.3 Aliasing

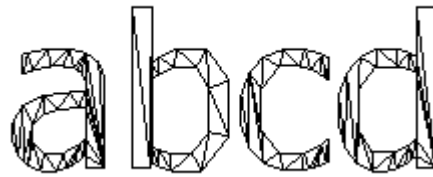
Pro jeho potlačení je proto nutné použít některou metodu antialiasingu. V našem případě byly implementovány dvě metody. První pomocí rozšíření *WGL_ARB_multisample* [42]. Druhá metoda, kterou jsme implementovali, využívá blendingu, tedy prolnutí kreslené geometrie a pozadí. Výsledky implementace antialiasingu jsou na (Obrázek 4.4).



Obrázek 4.4 Antialiasing

4.3 Písma

Dalším důležitým aspektem implementace bylo řešení vykreslování písem, jelikož vykreslované texty tvoří nezanedbatelnou část simulace. Bylo tedy potřeba vytvořit funkce, které by umožnily, jak běžné vypisování textu, tak jeho horizontální nebo vertikální zarovnání. Dále bylo nutné umožnit rotaci písma a změnu jeho velikosti podle požadavků. Vzhledem k těmto okolnostem jsme zavrhlí použití bitmapových písem, jelikož při jejich využití není možné písmo rotovat či měnit jeho velikost. V našem programu je proto využito písem vektorových, kdy každé vykreslované písmeno je tvořeno trojúhelníky (Obrázek 4.5).



Obrázek 4.5 Geometrie písma

Dalším požadavkem pro kreslení písmen je jejich dobrá čitelnost. U velkých písmen není při vykreslování pomocí trojúhelníků s čitelností problém, avšak při kreslení malých písmen je již čitelnost slabá (Obrázek 4.6).



Obrázek 4.6 Malé písmo

Problém s čitelností malých písem byl vyřešen pomocí obkreslení obrysu každého písmene čarou a využitím antialiasingu (Obrázek 4.7).



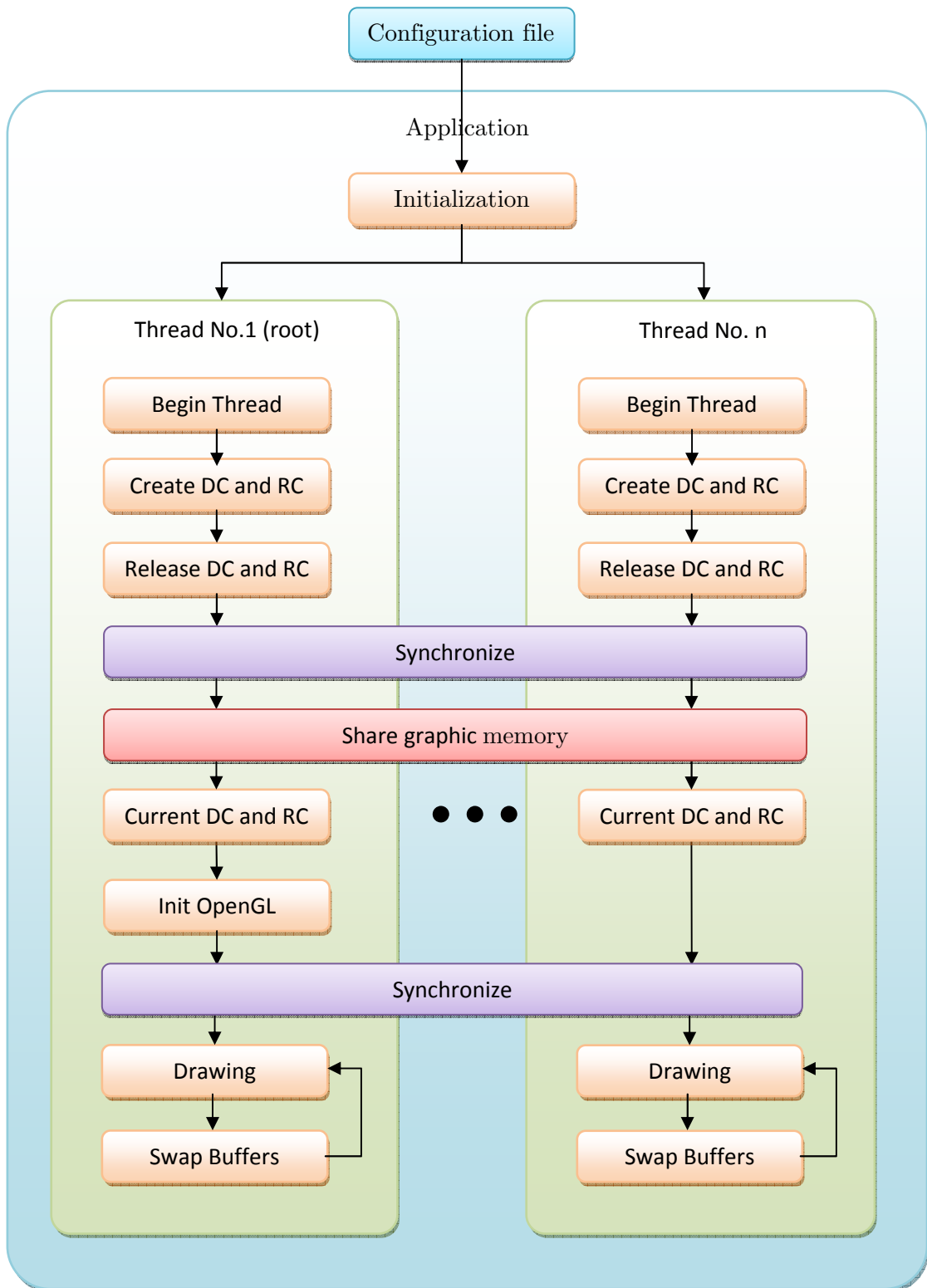
Obrázek 4.7 Malé písmo včetně obrysu

4.4 Implementace připojení více monitorů

V kapitole zabývající se návrhem jsme se rozhodli, že pro náš simulátor využijeme pro každý monitor separátní okno, do kterého se bude renderovat obsah vlastní nezávislou smyčkou.

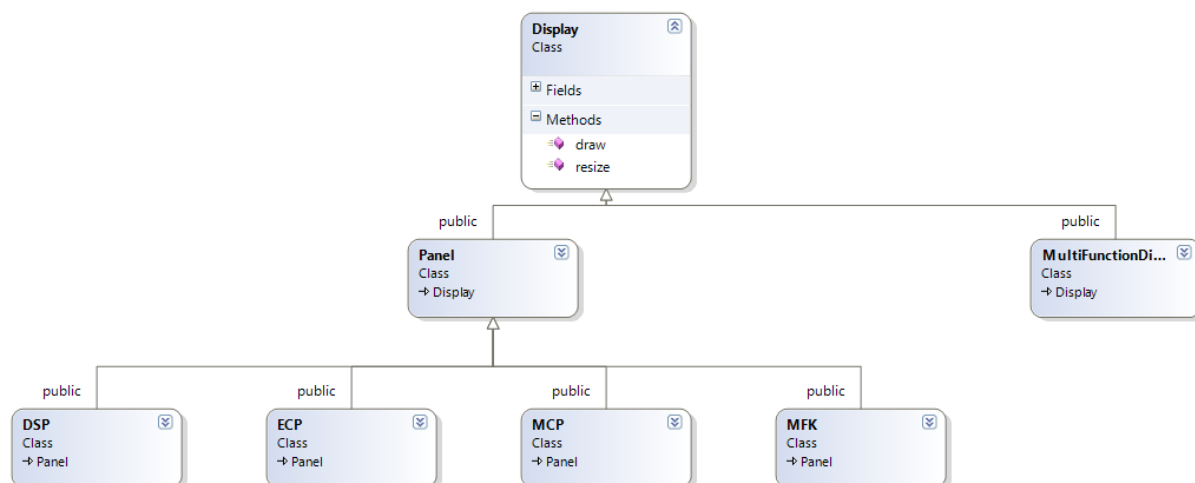
4.4.1 Vykreslovací smyčka

Každá tato smyčka bude fungovat ve svém vlákně, nestačí tedy pouze použít stejný Rendering Context RC, ale je potřeba zajistit sdílení grafických dat napříč všemi RC pomocí funkce *wglShareLists*. Toto sdílení umožní, aby se o načítání dat do grafické paměti staralo pouze jedno vlákno. Taktéž je třeba zajistit synchronizaci mezi vlákny. Výsledná implementace vykreslování je zobrazena na (Obrázek 4.8).



Obrázek 4.8 Implementace vykreslování

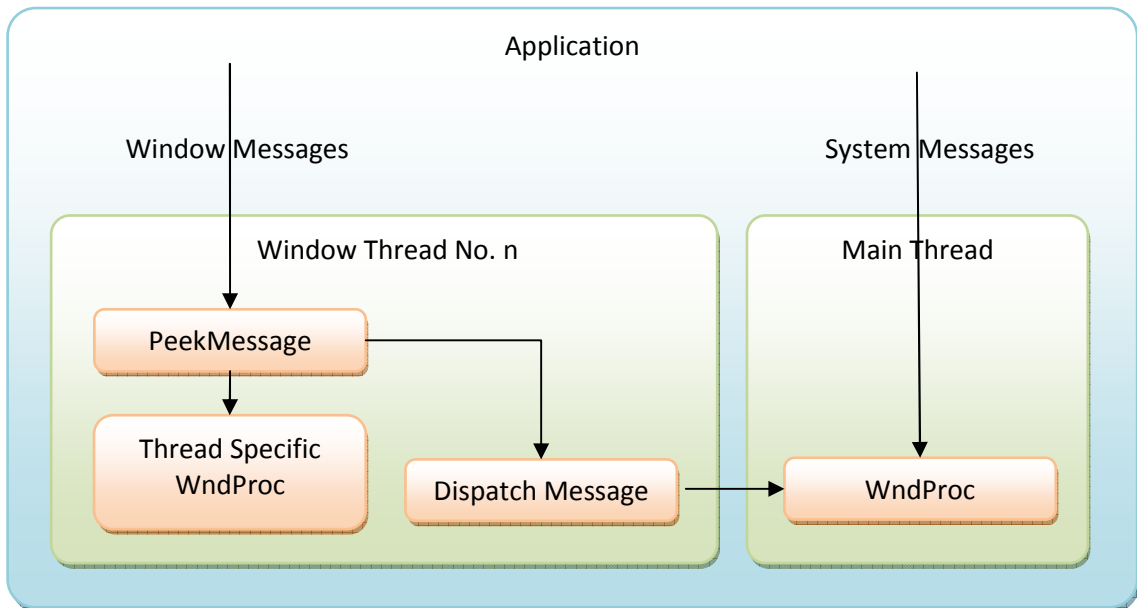
Požadavek na variabilitu a využití konfiguračního souboru lze snadno uspokojit pomocí využití seznamu objektů s parametry, který bude obsahovat konfigurační soubor pro každé okno. Při vykreslování potom pouze stačí, aby všechny vykreslované objekty obsahovaly zděděnou funkci pro vykreslení (Obrázek 4.9).



Obrázek 4.9: Class diagram jednotlivých vykreslovaných částí

4.4.2 Zasílání zpráv

Kromě vykreslovací smyčky obsahuje každá aplikace v systému Windows i smyčku pro přijímání zpráv, pomocí kterých reaguje na vnější události, jako jsou systémové zprávy, informace o stavu klávesnice, myši a další. Tuto smyčku musí obsahovat každé vytvořené okno. Jelikož všechna naše okna fungují jako jeden celkový program, je potřeba tomuto přizpůsobit přijímání systémových zpráv. Hlavní rozdíl je v tom, že je potřeba rozdělit zprávy na ty, které se týkají pouze jednoho okna, jako může být například změna jeho velikosti, minimalizace nebo maximalizace. Dále je pak potřeba oddělit ty zprávy, které mají vliv na aplikaci jako celek, což jsou vstupy z klávesnice, nebo například informace o uzavření okna. Zpracování zpráv je znázorněno na (Obrázek 4.10).



Obrázek 4.10 Zpracování zpráv

4.4.3 Thread Local Storage

Využití více nezávislých oken sebou přineslo i nutnost oddělení stavových proměnných pro jednotlivá okna. Toto se zejména týká informací o pohybu a událostech myši. Aby bylo možné mít tyto údaje uložené na stejném místě nezávisle na okně, pro které dané údaje platí, je zde tedy využito Thread Local Storage (TLS) [43]. TLS umožňuje, aby globální proměnné měly specifické hodnoty v rámci jednotlivých vláken. Kupříkladu v MS Visual Studiu, ve kterém je náš EFIS implementován, je TLS deklarováno klíčovým slovem `__declspec(thread)` [44].

Při použití TLS vykreslovací funkce nepotřebují mít informace, do kterého okna se kreslí a všechny potřebné údaje jsou pro ně k dispozici na stejném místě, avšak v každém vlákně s korektními hodnotami.

4.5 Implementace modulů

Jádro programu, které se stará o simulaci jednotlivých systémů a jejich vykreslování je integrováno jako jeden primární modul. Ostatní části programu, jako je modul pro komunikaci s FSX, nebo modul pro práci s periferními zařízeními, jsou implementovány jako DLL knihovny. Každý z těchto modulů má přístup do objektu Aircraft, kde může přistupovat k veškerým členským proměnným a podle potřeby i upravovat jejich hodnoty, a tak interagovat se simulací.

4 IMPLEMENTACE

5 Testování

K ověření toho zdali náš program odpovídá požadavkům, které jsme si stanovili v kapitole 3, je nutné provést jeho otestování. Dále je také zapotřebí provést testování jednotlivých částí programu po funkční a výkonnostní stránce.

5.1 Profiling

Pro otestování efektivity využívání výpočetních prostředků naším programem a případného nalezení úzkých hrdel implementace bylo použito metody profilingu za použití integrovaného profileru v MS Visual Studiu 2008. Profiling byl proveden za běžných provozních podmínek. Samotný test byl spuštěn po dobu jedné minuty. Během testování jsme se zaměřili zejména na funkce, které vykonávají nejvíce práce. (Obrázek 5.1).

Name	Time (msec)	%
Sleep	65 629,91	33,22
glCallLists	33 998,13	17,21
glColor3f	7 865,34	3,98
glVertex2d	7 410,21	3,75
drawFilledCircle(double,double,double)	6 730,06	3,41

Obrázek 5.1 Funkce vykonávající nejvíce práce

Také jsme se během testování zaměřili na nejčastěji volané funkce (Obrázek 5.2).

Name	Number of Calls	%
glVertex2d	11 565 845	38,55
glVertex2f	1 871 956	6,24
glPopMatrix	1 340 179	4,47
glPushMatrix	1 340 179	4,47
??A?\$basic_string@DU?\$char_traits@D@std@@@V?\$allocator@D@2@@@std@@@...	1 152 764	3,84

Obrázek 5.2 Nejčastěji volané funkce

Podrobnější výsledky provedeního testu jsou obsaženy v příloze A.

Z provedených testů vyplynulo, že na výkon naší aplikace mají největší vliv funkce na vykreslování písem. Konkrétně u kreslení písem jde o funkci *glCallLists*, starající se o vykreslení display listů obsahující uloženou geometrii písem. Dalším faktorem, který snižuje výkon aplikace, je vykreslování geometrických primitiv. V našem případě se jedná o používání funkce *glVerte2d*.

Pro zvýšení výkonu naší aplikace se tedy nabízejí následující řešení.

- Pro urychlení vykreslování písem lze snížit jejich geometrickou náročnost. Toho lze dosáhnout změnou parametru *deviation* u funkce *wglUseFontOutlines*, který určuje maximální odchylku od skutečné geometrie daného písma.

5 TESTOVÁNÍ

- Vykreslování grafických primitiv lze urychlit omezením zátěže na komunikaci mezi CPU a GPU. Nejjednodušší metodou pro toto urychlení je nahrazení vykreslování geometrie pomocí funkce *glVertex2d*, například využitím Display Listů nebo Vertex Array, případně Vertex Buffer Objektů (VBO).

Z výsledků je rovněž zřejmé, že funkce *Sleep* je aktivní značnou část běhu programu. Tento fakt však není způsoben neefektivností jejího využití, nýbrž tím, že v některých částech programu je použito k synchronizaci mezi vlákny právě této funkce. Její běh tedy nemá zásadní dopad na celkový výkon. V budoucnu by však bylo vhodné funkci nahradit jiným synchronizačním prostředkem, například semaforem.

5.2 Unit testing

Další formou testu je takzvaný Unit testing, který se používá k otestování funkčnosti jednotlivých částí programu. Testování probíhá tak, že pro testovanou část programu je vytvořena testovací třída, která ověří, že výstupy testované části programu odpovídají předpokladům pro dané vstupy.

V našem programu jsme testovali následující součásti:

- Práce navigační databáze – testovali jsme, zdali po vložení hodnot do navigační databáze jsou opětovně načtené hodnoty správné.
- Funkce pro práci s vlákny – testovali jsme, zda hodnoty v jednotlivých vláknech nejsou při běhu navzájem ovlivňovány a zda jsou specifické hodnoty v místech, kdy jsme využili TLS, rovněž správné.
- Spojení s FSX – testovali jsme, zdali po nastavení hodnot simulace v FSX pomocí rozhraní mezi EFISem a FSX a jejich opětovným načtením pomocí tohoto rozhraní nedochází k chybám. Tímto jsme ověřili funkčnost vytvořeného rozhraní.

U všech provedených testů odpovídaly výstupy předpokládaným hodnotám. Při testování komunikace mezi EFISem a FSX se hodnoty mírně lišily. Tento fakt byl zapříčiněn tím, že i když během simulace v FSX nedochází k pohybu letounu, dochází k mírným změnám hodnot jednotlivých proměnných.

5.3 Testování uživatelského rozhraní

Jelikož se v případě našeho systému jedná o značně specifický produkt, a proto k jeho používání a rovněž testování je zapotřebí, aby testující uživatelé disponovali alespoň základními znalostmi z oboru letectví. Proto bylo testování uskutečněno formou expertního review s uživateli, kteří jsou obeznámeni s funkcí systémů EFIS v letadlech. Samotného testování se zúčastnili jak uživatelé, kteří většinu svých zkušeností se systémy EFIS získali na různých simulátorech, tak lidé mající zkušenosti s funkcí reálných systémů v letadlech. Takto bylo docíleno, že jsme získali zpětnou vazbu od uživatelů, kteří mají různý úhel pohledu na simulátor. Zároveň jsme tak postihli všechny zamýšlené cílové kategorie uživatelů.

Pro samotné testování byl v rámci možností sestaven systém pro simulaci skládající se ze dvou PC. Na prvním byl spuštěn Microsoft Flight Simulator X, zobrazení pak obstarával jeden monitor. Na druhém PC byla spuštěna vytvořená simulace EFISu. Pro její zobrazení bylo využito dvou monitorů. Rozmístění jednotlivých prvků testovacího systému je zobrazeno na digramu (Obrázek 5.3). Ovládání EFISu bylo realizováno pomocí myši a ovládání FSX bylo realizováno ovladačem Saitek X52 Pro včetně pedálů.



Obrázek 5.3 Rozložení systému během testování

Samotné testování bylo rozděleno do několika částí:

- Představení projektu
- Představení EFISu, jeho možností a omezení včetně způsobu ovládání
- Pre-test dotazník - Uživateli je vyplněn dotazník, pomocí kterého jsou získány základní informace o jejich zkušenostech s jinými systémy a o jejich kvalifikaci:

Otázky pre-test dotazníku:

1. Věk?
 2. Zkušenosti s létáním?
 3. Kvalifikace?
 4. Počet nalétaných hodin?
 5. Předchozí zkušenosti s EFISem?
- Samotné testování - Samotný test je rovněž rozdělen do několika fází, kde každá má za cíl ověřit některé části systému. Celý test pak má za cíl otestovat v minimálním čase co možná nejvíce funkcí systému:

5 TESTOVÁNÍ

1. Vzlet
 - I. Noční vzlet z LOWI
 - II. Konfigurace FMC pomocí CDU pro vzlet (Vyplnění povinných údajů na stránkách IDENT, POS INIT, ROUTE, PERF INIT, THRUST LIM, TAKEOFF REF)
 - III. Nastavení konfigurace pro vzlet (Klapky, Stabilizátor,...)
 - IV. Samotný vzlet a stoupaní do letové hladiny (využití VSD)
 2. Let
 - I. Konfigurace autopilota na MCP
 - II. Test systému TCAS
 - III. (Let VOR to VOR)
 3. Přiblížení
 - I. Přiblížení k LKPR pomocí autopilota v módu approach
 - II. Nastavení stánky APPROACH REF na CDU
 - III. Přistání pomocí ILS (Noční přistání, případně zhoršené povětrnostní podmínky)
 - IV. Odpojení autopilota 200 stop nad povrchem
 4. Pojždění
 - I. Pojždění pomocí mapy letiště na ND
- Závěr zhodnocení - Na závěr proběhne semi-strukturovaný rozhovor s testujícími uživateli, během kterého bude hodnocen průběh testu.
- Uživatelům budou zejména kladeny následující dotazy:
1. Jaké je vaše hodnocení EFISu?
 2. Jaké vidíte možné využití systému?
 3. Kde spatřujete největší slabinu implementace?
 4. Kde spatřujete největší benefit systému?
 5. Objevily se během testu systémy, které nefungovaly korektně (případně které)?
 6. Jaké změny by měly být v systému provedeny?

Provedené testy s uživateli budou po dokončení vyhodnoceny a budou z nich vyvozeny důsledky a na jejich základě budou navrženy změny našeho systému EFIS.

5.3.1 Testování s uživatelem A

Výsledky pre-test dotazníku jsou v (Tabulka 5.1).

Otázka	Odpověď
Věk?	25 let
Zkušenosti s létáním?	15 let se simulátory, 5 let se skutečnými stroji
Kvalifikace?	-
Počet nalétaných hodin?	25 h
Předchozí zkušenosti s EFISem?	Ze simulátorů

Tabulka 5.1 Pre-test dotazník s uživatelem A

Poznatky, které vyplynuly během testování s uživatelem, jsou v (Tabulka 5.2).

Poznatek	Popis
Nefunkční AP disengage	Přepínač AP disengage na odpojení autopilota na MCP nefungoval.
Nefunkční navigace VOR to VOR	Nebylo možné naladit frekvenci pro navigaci pomocí VOR, nebyly k dispozici ani ukazatele na PFD a ND
Nefunkční WXR	Nelze aktivovat zobrazení terénního radaru pomocí spínače na ECP
Do CDU nelze zadat všechny údaje	Do CDU nešlo zadat všechny požadované údaje, konkrétně číslo letu a informace o vzletové runwayi
Nastavení letové výšky pro AP	Po nastavení požadované letové výšky pro autopilota v MCP nedošlo k požadované odezvě od systému autopilota v FSX
Nefunkční přiblížení pomocí ILS	Nebylo možné naladit frekvenci ILS a ani nebyly k dispozici potřebné ukazatele na PFD
Indikace pozice klapek	Po nastavení ovládní klapek do vysunuté pozice neodpovídalo zobrazení na EICAS panelu skutečné pozici klapek
Údaje na mapě letiště	Na mapě letiště na ND se nezobrazovaly0 identifikátory jednotlivých drah
Zobrazení trasy letu	Na ND se nezobrazovala trasa zadaná do FMC pomocí CDU
Panel rádia	Nebyl k dispozici panel pro možnosti naladění jednotlivých radionavigačních prvků
Nestabilita	Během chodu programu došlo k chybě, která zapříčinila pád simulace
Rozsah ND	Rozsah na ND neodpovídal hodnotám zobrazeným na měřítku a nastaveným na ECP
Přepínač módu ND	Přepínač módu ND na ECP nefungoval
Přepínání módů na ND	Nebylo možné přepnout ND do módu Plan pomocí tlačítka v ND
Výpočty V1, VR, V2	FMC neprovádělo automaticky výpočty rychlostí V1, VR a V2 a bylo nutno je zadat ručně pomocí MFK
Indikátory na EICAS panelu	Zobrazení některých indikátorů na EICAS displeji neodpovídalo

5 TESTOVÁNÍ

	zcela skutečnosti, nebyly k dispozici zejména ukazatele mezních hodnot
Údaje na minimapě	Na minimapě v PFD nebyly zobrazeny všechny údaje jako ve skutečném letadle

Tabulka 5.2 Výsledky testu s uživatelem A

5.3.2 Testování s uživatelem B

Výsledky pre-test dotazníku jsou v (Tabulka 5.3).

Otázka	Odpověď
Věk?	28 let
Zkušenosti s létáním?	Airbus A320
Kvalifikace?	Dopravní pilot, FO Airbus A320
Počet nalétaných hodin?	3500 h
Předchozí zkušenosti s EFISem?	Z letounu Airbus A320

Tabulka 5.3 Pre-test dotazník s uživatelem B

Poznátky, které vyplynuly během testování s uživatelem, jsou v (Tabulka 5.4).

Poznatek	Popis
Rozložení MFK a CDU	Umístění CDU a MFK vůči sobě nebylo vhodné, jelikož se vyskytovaly na opačných stranách a zadávání hodnot tak nebylo komfortní
Umístění PFD	PFD jako primární displej by měl být umístěn přímo proti pilotovi
Zadání redukce tahu	V CDU nebylo možné nastavit hodnoty pro redukce tahu v jednotlivých fázích letu
Výpočty V1, VR, V2	FMC neprovádělo automaticky výpočty rychlostí V1, VR a V2 a bylo nutno je zadat ručně pomocí MFK, rovněž nebyl k dispozici automatický výpočet pro váhy
Indikace maximálního úhlu stoupání	Nebyl k dispozici automatický výpočet pro maximální úhel stoupání a indikátor na PFD neukazoval správné hodnoty
Hodnoty pro přiblížení	Hodnoty pro přiblížení zobrazené na stránkách CDU neodpovídaly parametrům zadaným při předletové přípravě
Ovládání podvozku	Ovládání podvozku by mělo být realizováno jako přepínač nikoli

	pouze spínač
Test konfigurace pro vzlet	Nebyl k dispozici kompletní systém pro testování vzletové konfigurace (Takeoff configuration test)
Zobrazení minimálních rychlostí	Indikátor minimální rychlosti nebyl zobrazen ve správných fázích letu, rovněž i výpočet jeho hodnoty nebyl zcela správný
Flight director	Ukazatel flight directoru neukazoval správné hodnoty
Varování dosažení maximální rychlosti	Akustické varování o dosažení maximální provozní rychlosti (overspeed) nebylo aktivní při vysunutých klapkách
Varování dosažení pádové rychlosti	Nebylo k dispozici akustické varování při dosažení pádové rychlosti (stall)
Indikátor TCAS	Při indikaci varování od systému TCAS na PFD bylo zobrazeno na variometru pouze červené a nikoli i zelené pole
Odpojení autopilota	Autopilota bylo možné odpojit pomocí ovládání na MCP, nikoli i na joysticku, jak je tomu ve skutečném letadle
Ovládání prvků GUI	Ovládání přepínačů prvků GUI na MCP nebylo zcela intuitivní. Pozice, ve které bylo třeba kliknout myší, byla málo zvýrazněna

Tabulka 5.4 Výsledky testu s uživatelem B

5.3.3 Zhodnocení testování s uživateli

Vzhledem k již zmíněnému faktu, že pro naše expertní review jsme potřebovali uživatele s určitými znalostmi z oboru letectví a náš čas pro provedení testování byl značně omezen, podařilo se uskutečnit test pouze se dvěma uživateli. Toto sice zmenšilo množství získaných výsledků, ale i navzdory tomuto faktu jsme od uživatelů získali množství cenných podnětů. Proběhlé testy měly od uživatelů vesměs kladné reakce, avšak vyskytly se určité faktory, na které průběh testování upozornil. Tyto by se daly rozdělit do několika skupin.

- Problémy s hardware – z důvodu omezených hardwarových možností byla simulace EFISu spuštěna pouze na dvou monitorech. Nebylo tak možné realizovat simulaci se stejným rozložením komponent, jako je ve skutečném letadle. Některé komponenty byly tedy přesunuty z důvodu úspory místa. Tyto přesuny však zapříčiňovaly různé problémy, jako je například nevhodná

5 TESTOVÁNÍ

pozice MFK vůči CDU, nebo fakt že PFD nebylo umístěno přímo proti testujícímu uživateli jako ve skutečném letadle.

- Problémy zapříčiněné neúplnou implementací – z důvodu nedostatku času ve finální fázi vývoje aplikace a v některých případech i kvůli nedostatku informací o funkci reálného systému, již nebylo možné dokončit implementace některých částí. Jedná se zejména o výpočty prováděné FMC a s tím spojené údaje zobrazované na CDU. Proto docházelo v průběhu testování k potížím, kdy nebylo možno provést v těchto částech věrnou simulaci průběhu letu.
- Problémy zapříčiněné chybami v implementaci – na rozdíl od předchozích dvou kategorií, chyby zapříčiněné implementací lze snadno opravit. Nejzávažnější z těchto chyb byly tedy opraveny, některé i mezi testováními, jelikož se jednalo o markantní závady, které omezovaly funkčnost systému. Rovněž tyto opravy umožnily se při dalším testování zaměřovat na jiné aspekty systému, a tím pokrýt větší testovaný rozsah.

Celkově lze testy s uživateli hodnotit jako úspěšné, jelikož se v průběhu testování vyskytovaly zejména chyby související s neúplnou implementací některých systémů, zejména na FMC, nebo byly způsobeny nedostupností hardware. Soupis jednotlivých opravených i neopravených chyb je v (Tabulka 5.5).

Chyba	Oprava
Nefunkční AP disengage	Opraveno
Nefunkční navigace VOR to VOR	Implementována indikace pro navigaci pomocí VOR, nicméně není implementován panel rádia
Nefunkční WXR	Neimplementováno z důvodu nedostatku informací získaných z FSX
Do CDU nelze zadat všechny údaje	Implementace CDU není kompletní
Nastavení letové výšky pro AP	Opraveno
Nefunkční přiblížení pomocí ILS	Opraveno
Indikace pozice klapek	Opraveno
Údaje na mapě letišť	Implementace není kompletní
Zobrazení trasy letu	Částečně implementováno
Panel rádia	Neimplementováno
Nestabilita	Neopraveno
Rozsah ND	Opraveno
Přepínač módu ND	Opraveno
Přepínání modů na ND	Opraveno
Výpočty V1, VR, V2	Neimplementováno z důvodu nedostatku informací
Indikátory na EICAS panelu	Opraveno
Údaje na minimapě	Částečně implementováno
Zadání redukce tahu	Neimplementováno z důvodu

			nedostatku informací
Indikace stoupání	maximálního úhlu		Neimplementováno z důvodu nedostatku informací
Hodnoty pro přiblížení			Neimplementováno z důvodu nedostatku informací
Test konfigurace pro vzlet			Neimplementováno z důvodu nedostatku informací
Zobrazení minimálních rychlostí			Neopraveno z důvodu nedostatku informací
Flight director			Neopraveno
Varování rychlosti	dosažení maximální		Neopraveno z důvodu nedostatku informací
Varování rychlosti	dosažení pádové		Neopraveno z důvodu nedostatku informací
Indikátor TCAS			Neopraveno z důvodu nedostatku informací

Tabulka 5.5 Soupis chyb

6 Závěr

V této kapitole zhodnotíme výsledky, kterých se podařilo dosáhnout. Zhodnotíme zde jak splnění zadání projektu, tak vyhovění stanoveným funkčním a nefunkčním požadavkům. Také zhodnotíme výsledky dosažené během testování a budeme uvažovat možný další rozvoj našeho projektu a jeho uplatnění.

6.1 Výsledky implementace

Naše implementace systému EFIS letounu Boeing 787 je funkční simulací reálného systému. Podařilo se tedy navrhnout a implementovat kvalitní systém pro simulaci. Je implementována většina funkcí reálného EFISu. Z důvodu nedostatku času a rovněž nedostatku podrobných informací o funkci reálného systému nebylo možné některé části simulovat s absolutní věrohodností. Je implementována většina funkcí pro samotné vykreslování displejů a panelů. Dále je částečně implementována funkce FMC včetně navigační databáze. Avšak některé části FMC nebylo ze zmíněných důvodů možno implementovat zcela detailně.

6.2 Splnění požadavků

Nyní se zaměříme konkrétně na vyhodnocení splnění funkčních a nefunkčních požadavků, které jsme si stanovili v kapitole návrh.

Vyhodnocení funkčních požadavků:

1. Simulovat displeje systému EFIS, zejména PFD,ND, CDU – Simulace displejů EFISu je funkční a až na minoritní funkce kompletní. Největší nedostatky jsou u simulace CDU, kde nebylo možno z již zmíněných důvodů implementaci dokončit.
2. Simulovat panely kokpitu potřebné pro interakci se simulátorem při nedostupnosti odpovídajícího hardware – Jsou kompletně simulovány všechny panely potřebné pro základní ovládání EFISu.
3. Simulovat důležité systémy v letadle – TCAS, EGPWS – systémy jsou implementovány a v průběhu testování byla jejich funkčnost úspěšně ověřena.
4. Podporovat zobrazení na více monitorech – Je implementováno multimonitorové zobrazení. Konfigurace je umožněna pomocí konfiguračního souboru.
5. Komunikovat s FSX jako vizuální platformou – Bylo realizováno rozhraní mezi naším programem a FSX. Jeho funkčnost byla ověřena jak unit testingem, tak i při testování s uživateli.
6. Umožnit komunikaci s periferními zařízeními – Program je schopen přijímat signály od připojeného joysticku. Program do budoucna umožňuje i komunikaci s jiným hardwarem pomocí vytváření dalších modulů.

Vyhodnocení nefunkčních požadavků:

6 ZÁVĚR

1. Aplikace bude vyvinuta pro platformu Windows – Aplikace je vytvořena pro běh v systému Windows a využívá funkce z WinAPI.
2. Aplikace bude koncipována modulárně – K programu je možné připojovat moduly ve formě DLL knihoven.
3. V co nejmenší míře používat knihovny třetích stran – Nejsou použity žádné knihovny třetích stran. Jedinou částí, u které je využit kód třetí strany, je část pro dekompilaci BGL souborů.
4. Uchovávat nastavení v konfiguračním souboru – Nastavení je uloženo v souboru *conf.txt*, jeho popis je popsán v příloze D.

6.3 Budoucí rozvoj a uplatnění

Jak jsme si ověřili během uživatelských testů, program může být dobře využíván jako simulátor. Díky své nízké hardwarové náročnosti může být dostupný pro široké spektrum uživatelů. Program by pro jeho do značné míry univerzální strukturu bylo možné v případě zájmu uzpůsobit i pro simulaci jiných strojů než je Boeing 787. Program by mohl být rovněž využit i jako postupový simulátor pro výcvik posádek letadel. Nicméně tento způsob využití by však vyžadoval dokončení všech funkcí simulátoru a jeho certifikaci. V případě České republiky u Úřadu pro civilní letectví.

Pokud budeme uvažovat budoucí rozšíření našeho simulátoru, tak kromě dokončení nekompletních funkcí by bylo možné simulátor rozšířit i o funkcionality nové. Jedná se zejména o doplnění našeho systému o funkční hardware simulující reálnou funkci. Jelikož, pokud je simulace celého kokpitu provedena zcela softwarově, jak je tomu v našem případě, není možné, aby simulace byla zcela věrohodná. Tento fakt se projevil i při testování s uživateli, kdy emulovaná MFK nebyla příliš vhodná pro komfortní ovládání a uživatelé navrhovali její nahrazení klávesnicí hardwarovou nebo alespoň dotykovou obrazovkou. Rovněž pokud bychom chtěli pro náš simulátor získat certifikaci pro skutečný letový výcvik, byl by vývoj hardwarové části nezbytný, jelikož certifikace je vždy udělována kompletnímu systému, tedy softwaru + hardwaru.

Dalším směrem vývoje by bylo, po uvedení nového simulátoru od firmy Microsoft [20], upravení komunikačního rozhraní i pro tuto novou verzi, protože je po grafické stránce znatelně pokročilejší než FSX (Obrázek 6.1) a simulace by se stala opět o kus reálnější.



Obrázek 6.1 Porovnání FSX a MS Flight [19]

6.4 Porovnání se skutečným systémem

Jelikož nebylo možné porovnat náš systém přímo se skutečným systémem v reálném letadle nebo v simulátoru vyšší kategorie, a také nebylo možné k testování přizvat uživatele se zkušenostmi přímo z letounu Boeing 787, porovnávali jsme tak obecnou funkčnost jednotlivých systémů s jinými stroji, se kterými měli testující uživatelé zkušenosti. Toto nám umožnilo otestovat systém po funkční stránce, jelikož funkce základních systémů se u různých letadel diametrálně neliší. Avšak porovnání s konkrétním systémem tak bylo možné získat pouze z dostupných zdrojů, jako jsou fotografie či videa. Na (Obrázek 6.2) a (Obrázek 6.3) je vizuální porovnání našeho systému a fotografií skutečného systému.

6 ZÁVĚR



Obrázek 6.2 Porovnání vytvořeného a skutečného EFISu (nahore vytvořený EFIS, dole fotografie ze simulátoru úrovně Level D)) [26]



Obrázek 6.3 Porovnání vytvořeného a skutečného EFISu (nahore vytvořený EFIS, dole fotografie ze simulátoru úrovně Level D)) [26]

Literatura

- [1] Tim Nelson, "787 Systems and Performance," Flight Operations Engineering, Boeing Commercial Airplanes, 2005.
- [2] Burkhard Domke, "Boeing 787 Lessons Learnt," Head of Engineering Intelligence, Future Projects Office, Airbus S.A.S., 2008.
- [3] Lori Gunter. (2011) Boeing Begins Final Phase of 787 Flight Testing. [Online]. <http://boeing.mediaroom.com/index.php?s=43&item=1820>
- [4] (2011) All things 787. [Online]. <http://nyc787.blogspot.com>
- [5] International Civil Aviation Organization. (2011) Doc8643 - ICAO Aircraft Type Designators. [Online]. <http://www.icao.int/anb/ais/8643/>
- [6] The Boeing Company. (2011) Orders and Deliveries. [Online]. <http://active.boeing.com/commercial/orders/>
- [7] Alan R. Jacobsen, "787 Flight Deck Displays," Boeing Commercial Airplanes, 2006.
- [8] The Boeing Company, "787-8 Flight Crew Operations Manual," 2010.
- [9] Mike Carriker, "Boeing 787 Dreamliner Flight Deck Safety, Comfort, Efficiency," Chief Pilot, 787 Program, Boeing Commercial Airplanes, 2006.
- [10] Letecké Simulátory.cz. (2011) Letecký тренаžér 737NG. [Online]. <http://www.letecke-simulatory.cz/media/fotogalerie/>
- [11] US Federal Aviation Administration, "14 CFR Part 60," 2008.
- [12] Joint Aviation Authorities Training Organisation, "JAR-FSTD A - Aeroplane Flight Simulation Training Devices," 2008.
- [13] Úřad pro civilní letectví, "JAR-STD 1A - Letové simulátory pro letouny," 2004.
- [14] Úřad pro civilní letectví, "JAR-STD 2A - Letová výcviková zařízení pro letouny," 2004.
- [15] Úřad pro civilní letectví, "JAR-STD 3A - Trenažéry letových a navigačních postupů pro letouny," 2004.
- [16] Úřad pro civilní letectví, "JAR-STD 4A - Základní přístrojová výcviková zařízení," 2004.
- [17] Czech Airlines, a. s. (2011) Výcvik posádek - Základní informace. [Online]. http://www.czechairlines.com/cs/posadky/pos_letove_simulatory/pos_let_simulatory_info.htm
- [18] Mattias Edlund, "Microsoft Flight Simulator 2004, Product Information," Microsoft Northern Europe, Microsoft PC Games, 2003.

LITERATURA

- [19] (2011) AVSIM Forum. [Online]. <http://forum.avsim.net/>
- [20] Microsoft Corporation. Microsoft Flight. [Online]. <http://www.microsoft.com/games/flight/>
- [21] Lockheed Martin Corporation. (2011) Lockheed Martin - Prepar3D. [Online].
<http://www.prepar3d.com/>
- [22] Dana Casey. (2011) Lockheed Martin - Prepar3D. [Online].
http://www.lockheedmartin.com/news/press_releases/2011/061311Prepar3D-NFA.html
- [23] Christian Stock. (2007) Sim Pilot Magazine. [Online].
http://www.simpilotnet.com/index.php?option=com_content&task=category&id=16&Itemid=&Itemid=3
- [24] Tim Capps. (2009) AVSIM Commercial Aircraft Review. [Online].
<http://www.avsim.com/pages/0909/Abacus/787.htm>
- [25] Project Magenta. (2011) Project Magenta - High End Flight Simulation Software. [Online].
<http://www.projectmagenta.com/>
- [26] Jason Paur. (2011) Try Before You Fly: How Dreamliner Pilots Train Without Lifting Off. [Online]. <http://www.wired.com/autopia/2011/09/787-flight-training/>
- [27] Microsoft Corporation. (2008) SimConnect SDK Reference. [Online].
<http://msdn.microsoft.com/en-us/library/cc526983.aspx>
- [28] OpenGL.org. (2011) GLUT - The OpenGL Utility Toolkit. [Online].
<http://www.opengl.org/resources/libraries/glut/>
- [29] Microsoft Corporation. (2011) The Virtual Screen. [Online]. [http://msdn.microsoft.com/en-us/library/dd145136\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/dd145136(v=vs.85).aspx)
- [30] Microsoft Corporation. (2008) Simulation Variables. [Online]. <http://msdn.microsoft.com/en-us/library/cc526981.aspx>
- [31] Peter Dowson. (2011) The Official Peter Dowson Page. [Online].
<http://www.schiratti.com/dowson.html>
- [32] Peter Dowson, "FSUIPC4 User Guide," 2011.
- [33] Peter Dowson, "WideFS User Guide," 2011.
- [34] Microsoft Corporation. (2008) Terrain and Scenery. [Online]. <http://msdn.microsoft.com/en-us/library/cc707102.aspx>
- [35] Winfried Orthmann, "FS X BGL File structure," 2006.
- [36] Jeppesen. (2011) Jeppesen. [Online]. <http://www.jeppesen.com>

- [37] Navigraph. (2011) Navigraph. [Online]. <http://www.navigraph.com>
- [38] Vlastimil Havran, "Proximity Search and its Applications," Czech Technical University in Prague - Faculty of Electrical Engineering - Department of Computer Graphics and Interaction, 2008.
- [39] Jessica Zorich, "Monitoring AI Objects," 2008.
- [40] Alessandro G. Antonini, "BGL to XML converter," Central Park Informatica, 2004.
- [41] Microsoft Corporation. (2011) Microsoft Visual Studio. [Online]. <http://www.microsoft.com/cze/msdn/vstudio/2010/>
- [42] (2011) ARB_multisample. [Online]. <http://www.opengl.org/registry/specs/ARB/multisample.txt>
- [43] Microsoft Corporation. (2011) Thread Local Storage. [Online]. [http://msdn.microsoft.com/en-us/library/windows/desktop/ms686749\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/ms686749(v=vs.85).aspx)
- [44] Microsoft Corporation. (2011) Compiler Implementation for TLS. [Online]. <http://msdn.microsoft.com/en-us/library/6yh4a9k1.aspx>
- [45] Sparx Systems. (2011) Enterprise Architect. [Online]. www.sparxsystems.com
- [46] The Boeing Company, "777/787 Flight Crew Training Manual," 2010.
- [47] The Boeing Company, "787 Flight Crew Training Manual," 2010.
- [48] The Boeing Company, "787 Quick Reference Handbook," 2010.
- [49] Abacus Software, Inc. (2011) Abacus. [Online]. <http://www.abacuspublisher.com>

LITERATURA

Příloha A – Výsledky profilingu

Funkce seřazené podle množství času, které pro svůj běh požadují, jsou na (Tabulka A.1) funkce seřazené podle počtu volání jsou pak na (Tabulka A.2). Kompletní výsledky profilingu jsou pak na příloženém DVD v adresáři results/.

Function Name	Elapsed Exclusive Time %	Elapsed Exclusive Time
Sleep	33,22	65 629,91
glCallLists	17,21	33 998,13
glColor3f	3,98	7 865,34
glVertex2d	3,75	7 410,21
drawFilledCircle(double,double,double)	3,41	6 730,06
drawDashedLine(double,double,double,double,double,double)	3,23	6 372,61
SwapBuffers	3,06	6 036,93
glBegin	3,03	5 976,60
glPrintCentreXY(class Font *,double,double,double,char const *,...)	2,77	5 479,05
CDUDisplay::print(class Font *,double,double,struct CDUDisplayLine)	2,39	4 719,13
drawRadius(double,double,double,double,double)	1,43	2 826,11
glPopAttrib	1,08	2 131,81
glPrintCentreY(class Font *,double,double,double,char const *,...)	1,07	2 113,62
glPushMatrix	0,9	1 771,45
??A?\$basic_string@DU?\$char_traits@D@std@@V?\$allocator@D@2@@@std@@QA EAADI@Z	0,88	1 733,46
drawCircle(double,double,double)	0,87	1 727,20
??5?\$basic_istream@DU?\$char_traits@D@std@@@std@@@QAEAAV01@AAN@Z	0,72	1 420,11
glVertex2f	0,71	1 408,89
glPushAttrib	0,7	1 374,26
vsprintf	0,68	1 348,33
glPopMatrix	0,65	1 282,15
drawFilledRadius(double,double,double,double,double)	0,61	1 213,43
SuspendThread	0,6	1 186,66
glClear	0,6	1 183,57
Panel::drawButton(double,double,double,double,double,class Font *,double,char *,char *)	0,6	1 178,80
Mouse::getMousePosition(double *,double *)	0,55	1 077,51
glLineWidth	0,51	1 003,92
glPrintCentreX(class Font *,double,double,double,char const *,...)	0,5	982,61
glPrint(class Font *,double,double,double,char const *,...)	0,48	956,16
glRotated	0,42	833,39
Mouse::isIn(double,double,double,double)	0,4	791,66
wglUseFontOutlinesA	0,35	682,55
glTranslated	0,33	656,06

Tabulka. A.2 – Funkce seřazené podle jejich doby běhu

PŘÍLOHA A – VÝSLEDKY PROFILINGU

Function Name	Number of Calls
glVertex2d	11 565 845
glVertex2f	1 871 956
glPushMatrix	1 340 179
glPopMatrix	1 340 179
??A?\$basic_string@DU?\$char_traits@D@std@@V?\$allocator@D@2@@std@@QAEADI@Z	1 152 764
glTranslated	1 129 933
glCallLists	988 140
glPopAttrib	988 140
glPushAttrib	988 140
glListBase	988 140
glColor3f	785 245
glLineWidth	691 430
glBegin	663 368
glEnd	663 368
glScaled	557 086
glRotated	501 314
??5?\$basic_istream@DU?\$char_traits@D@std@@@std@@QAEAAV01@AAN@Z	487 736
vsprintf	295 522
glGetDoublev	292 540
??\$?5DU?\$char_traits@D@std@@@std@@YAAAV?\$basic_istream@DU?\$char_traits@D@std@@@0@AAV10@PAD@Z	275 232
glColor(int)	205 393
Mouse::getMousePosition(double *,double *)	146 270
Mouse::isIn(double,double,double,double)	146 270
gluUnProject	146 270
glGetIntegerv	146 270
??1?\$basic_string@DU?\$char_traits@D@std@@V?\$allocator@D@2@@std@@QAE@XZ	128 500
??0?\$basic_string@DU?\$char_traits@D@std@@V?\$allocator@D@2@@std@@QAE@ABV01@@@Z	128 372
glPrintCentreXY(class Font *,double,double,double,char const *,...)	112 353
CDUDisplayLine::CDUDisplayLine(struct CDUDisplayLine const &)	79 926
glPrintCentreY(class Font *,double,double,double,char const *,...)	77 595
??5?\$basic_istream@DU?\$char_traits@D@std@@@std@@QAEAAV01@AAH@Z	74 823
??\$?5DU?\$char_traits@D@std@@@std@@YAAAV?\$basic_istream@DU?\$char_traits@D@std@@@0@AAV10@AAD@Z	74 772
__ftol2_sse	74 714
CDUDisplay::print(class Font *,double,double,struct CDUDisplayLine)	48 440

(Tabulka A.2 – Funkce seřazené podle počtu volání)

Příloha B – Class diagram

Kompletní diagram tříd je kvůli své obsáhlosti na přiloženém DVD v adresáři UML/ ve formátu XML. Soubor byl vytvořen v programu Enterprise Architect [45].

Příloha C – Instalační příručka

Instalaci je možné provést dvěma různými způsoby. První je instalace Microsoft Flight Simulatoru X a EFISu na jeden PC. Výhodou této varianty je jednodušší konfigurace oproti tomu druhá varianta tedy instalace na 2 PC umožňuje rozdělit zátěž, a tím i snižuje výkonnostní požadavky na hardware jednotlivých PC.

C.1 Instalace na 1 PC

Požadavky na hardware:

- Windows XP SP2, Windows Vista, Windows 7
- Dvujádrový procesor
- 2 GB RAM
- Grafická karta s podporou DirectX 9 a OpenGL
- 15GB volného místa

Postup instalace:

1. Nainstalovat poslední dostupnou verzi DirectX
2. Nainstalovat FSX
3. Nainstalovat Service Pack 2 pro FSX nebo datadisk FSX Acceleration
4. Nainstalovat Visual C++ 2008 Redistributable Package
5. Nahrát do PC složku s EFISem (např. *efis/*)
6. Do složky s *efis/scenery/001/scenery/* nahrát soubory „*APX*.bgl*“ z instalace FSX
7. Nainstalovat písma *787C.ttf* a *LCD.ttf*
8. Do složek *efis/navData1/* a *efis/navData2/* nahrát aktuální verze navigační databáze

C.2 Instalace na 2 PC

Požadavky na hardware PC1:

- Windows XP SP2, Windows Vista, Windows 7
- Dvujádrový procesor
- 1 GB RAM
- Grafická karta s podporou DirectX 9
- 14GB volného místa

Postup instalace PC1:

1. Nainstalovat poslední dostupnou verzi DirectX
2. Nainstalovat FSX
3. Nainstalovat Service Pack 2 pro FSX nebo datadisk FSX Acceleration

PŘÍLOHA C – INSTALAČNÍ PŘÍRUČKA

4. Nakonfigurovat soubor SimConnex.xml podle postupu na [27]

Požadavky na hardware PC2:

- Windows XP SP2, Windows Vista, Windows 7
- Dvoujádrový procesor
- 1 GB RAM
- Grafická karta s podporou OpenGL
- 2GB volného místa

Postup instalace PC2:

1. Nainstalovat Visual C++ 2008 Redistributable Package
2. Nainstalovat SimConnect Client
3. Nahrát do PC složku s EFISem (např. *efis/*)
4. Do složky s *efis/scenery/001/scenery/* nahrát soubory „*APX*.bgl*“ z instalace FSX
5. Nainstalovat písma *787C.ttf* a *LCD.ttf*
6. Do složek *efis/navData1/* a *efis/navData2/* nahrát aktuální verze navigační databáze
7. Nakonfigurovat soubor SimConnex.cnf podle postupu na [27]

Příloha D – Uživatelská příručka

D.1 Spuštění

Program lze spustit pomocí souboru *787C.exe*. Pro aktivaci spojení mezi EFISem a FSX je třeba spustit FSX a následně EFIS. Pokud dojde ke spojení obou programů je potřeba aktivovat vzájemnou komunikaci. Pro toto je nutné, aby EFIS od FSX obdržel signál o zahájení simulace. Tento signál lze odeslat buďto zahájením letu, nebo při již aktivním letu je nejjednodušším způsobem stisknutí 2x klávesy ESC, což vyšle potřebný signál a zahájí komunikaci.

D.2 Konfigurace

Zobrazení jednotlivých součástí EFISu lze nakonfigurovat pomocí konfiguračního souboru *conf.txt*. Soubor má následující formát:

<počet oken> <fullscreen> <rámec>

<počet částí> <x> <y> <šířka> <výška>

<id části> <x> <y> <šířka> <výška>

Úvodní sekce:

<počet oken> - počet oken, které budou použity pro vykreslování. Optimální počet oken je stejný jako počet monitorů, ale lze použít jakoukoli jinou konfiguraci.

<fullscreen> - určuje zdali budou jednotlivá okna zobrazena ve fullscreen módu. Možné hodnoty jsou 0 nebo 1.

<rámec> - Okolo jednotlivých MFD budou zobrazeny virtuální rámečky. Možné hodnoty jsou 0 nebo 1. Na (Obrázek D.1) je zobrazen MFD bez a se zobrazeným rámečkem.



Obrázek D.1 Rámec okolo MFD

PŘÍLOHA D – UŽIVATELSKÁ PŘÍRUČKA

Po úvodní sekci následují sekce popisující konfiguraci jednotlivých oken, počet těchto sekcí odpovídá hodnotě **<počet oken>**.

<počet částí> - Obsahuje číslo udávající počet panelů a displejů, které se zobrazují v daném okně.

<x> - Horizontální pozice okna na obrazovce

<y> - Vertikální pozice okna na obrazovce

<šířka> - Šířka okna

<výška> - Výška okna

Dále následují části specifikující jednotlivé displeje a panely. Jejich počet je stejný jako hodnota **<počet částí>**.

<id části> - ID specifikující konkrétní displej nebo panel, který bude vykreslen. ID jednotlivých panelů a displejů jsou v (Tabulka D.1).

<x> - Horizontální pozice vzhledem k danému oknu

<y> - Vertikální pozice vzhledem k danému oknu

<šířka> - Šířka okna

<výška> - Výška okna

ID	Význam
101	Kapitánův vnější displej
102	Kapitánův vnitřní displej
103	Vnější displej prvního důstojníka
104	Vnitřní displej prvního důstojníka
105	Spodní displej
200	Prázdný panel
201	Levý EFIS control panel
202	Pravý EFIS control panel
203	Levý display selection panel
204	Pravý display selection panel
205	Levá multifunkční klávesnice
206	Pravá multifunkční klávesnice
207	Mode control panel
208	Levý glareshield panel
209	Pravý glareshield panel
210	Center forward panel

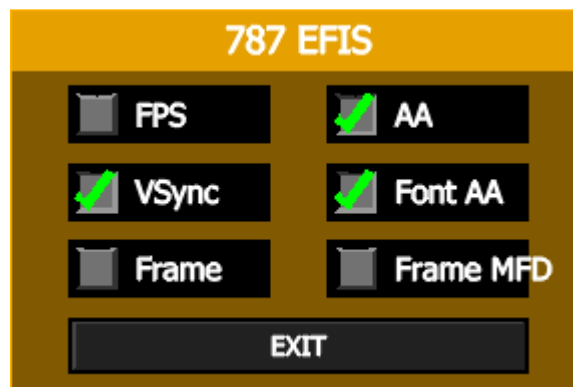
(Tabulka D.1 Identifikátory částí systému)

Ukázka možného obsahu konfiguračního souboru:

2	0	0		
5	1280	-224	1280	1024
101	427	50	853	640
208	65	750	825	200
201	890	750	252	200
203	1142	750	138	200
205	65	150	200	600
3	0	0	1280	1024
102	0	50	853	640
105	853	50	853	640
207	0	750	928	200

D.3 Uživatelské rozhraní

Jednotlivé prvky uživatelského rozhraní lze ovládat pomocí klávesnice a myši. Menu pro nastavení některých částí EFISu lze vyvolat stisknutím klávesy ESC (Obrázek D.2).



(Obrázek D.2 – GUI EFISu)

Zde lze nastavit následující položky:

- FPS – Zobrazí snímkové frekvence pro jednotlivá okna
- VSync – Aktivuje vertikální synchronizaci
- Frame – Zobrazí rám okolo displejů
- Frame MFD – Zobrazí tenký rámeček okolo displejů
- AA – Zapne antialiasing
- Font AA – Zvýší kvalitu vykreslování písem
- EXIT – Ukončí program

D.4 Záznamy letu

Při každé simulaci letu, kdy je EFIS spojen s FSX, vytváří systém záznam letu. Každý tento záznam je ukládán do adresáře s EFISem jako textový soubor s názvem ve tvaru:

PŘÍLOHA D – UŽIVATELSKÁ PŘÍRUČKA

DFDR_<rok>-<měsíc>-<den>_<hodina>-<minuta>-<vteřina>.txt

Pokud je EFIS spuštěn a nedojde ke spojení s FSX program přehraje záznam uložený v souboru *DFDR.txt*.

Příloha E – Stavové proměnné

Ve třídě Variables, která je členem třídy Aircraft je uložena většina stavových proměnných. Tyto proměnné jsou ovlivňovány jednotlivými moduly aplikace (jako je např. modul FSX.dll pro komunikaci s FSX) v tabulce (Tabulka E.1) je uveden jejich seznam včetně významu a jednotek.

Název	Význam	Jednotky
airTemperature	Vnější teplota	Celsius
totalAirTemperature	Total air temperature	Celsius
latitude	Zeměpisná šířka	Radians
longitude	Zeměpisná délka	Radians
altitude	Nadmořská výška	Feet
elevation	Výška povrchu	Feet
aboveGround	Výška nad povrchem	Feet
onGround	Letadlo na zemi	Boolean
bankAngle	Bank angle	Degrees
pitchAngle	Pitch angle	Degrees
groundSpeed	Rychlost vůči povrchu	Knots
indicatedAirpseed	Indikovaná rychlost	Knots
trueAirspeed	Skutečná rychlost	Knots
mach	Mach	Number
verticalSpeed	Vertikální rychlost	Feet per second
magneticHeading	Magnetický kurz	Radians
aircraftWeight	Váha letadla	Pounds
alpha	Úhel náběhu	Radians
beta	Sideslip	Radians
accelerationX	Zrychlení letadla v ose X	Feet per second squared
accelerationY	Zrychlení letadla v ose Y	Feet per second squared
accelerationZ	Zrychlení letadla v ose Z	Feet per second squared
rotationX	Rotace letadla v ose X	Degrees per second
rotationY	Rotace letadla v ose Y	Degrees per second
rotationZ	Rotace letadla v ose Z	Degrees per second
speedX	Rychlost letadla v ose X	Feet per second
speedY	Rychlost letadla v ose Y	Feet per second
speedZ	Rychlost letadla v ose Z	Feet per second
windSpeed	Rychlost větru	Knots
windDirection	Směr větru	Degrees
gearDown	Podvozek vysunut	Boolean
gearNosePosition	Pozice předního podvozku	Number (0-1)
gearMainLeftPosition	Pozice hlavního podvozku L	Number (0-1)
gearMainRightPosition	Pozice hlavního podvozku R	Number (0-1)
gearNoseLocked	Přední podvozek zajištěn	Boolean
gearMainLeftLocked	Hlavní podvozek zajištěn L	Boolean

PŘÍLOHA E – STAVOVÉ PROMĚNNÉ

gearMainRightLocked	Hlavní podvozek zajištěn R	Boolean
gearNoseDoorPosition	Pozice dveří předního podvozku	Number (0-1)
gearMainLeftDoorPosition	Pozice dveří hlavního podvozku L	Number (0-1)
gearMainRightDoorPosition	Pozice dveří hlavního podvozku R	Number (0-1)
gearLeverPosition	Pozice ovládání podvozku	Number (0-1)
autobrake	Pozice automatické brzdy	Number
flapsPositionLeftLeadingEdge	Úhel vysunutí slotů L	Degrees
flapsPositionLeftTrailingEdge	Úhel vysunutí klapek L	Degrees
flapsPositionRightLeadingEdge	Úhel vysunutí slotů R	Degrees
flapsPositionRightTrailingEdge	Úhel vysunutí klapek R	Degrees
flapsLeverPosition	Pozice ovládání klapek	Number
engineLeverPositionL	Pozice levé přípusti	Number (0-1)
engineLeverPositionR	Pozice pravé přípusti	Number (0-1)
fuelCutoffRunSwitchOnL	Pozice přepínače paliva L	Boolean
fuelCutoffRunSwitchOnR	Pozice přepínače paliva R	Boolean
speedbrakeArmed	Rychlostní brzda odjištěna	Boolean
speedbrakeLeftPosition	Pozice levých spojlerů	Number (0-1)
speedbrakeRightPosition	Pozice pravých spojlerů	Number (0-1)
speedbrakeLeverPosition	Pozice ovladače spojlerů	Number (0-1)
parkingBrakeOn	Parkovací brzda	Boolean
parkingBrakePositionOn	Pozice ovladače parkovací brzdy	Boolean
rudderPosition	Pozice kormidla	Number (0-1)
rudderTrim	Trim kormidla	Number (0-1)
stabilizerPosition	Pozice stabilizátoru	Number (0-1)
stabilizerTrim	Trim stabilizátoru	Number (0-1)
aileronLeftPosition	Pozice křidélek L	Number (0-1)
aileronRightPosition	Pozice křidélek R	Number (0-1)
aileronTrim	Trim křidélek	Number (0-1)
flaperonLeftPosition	Pozice flaperonu L	Number (0-1)
flaperonRightPosition	Pozice flaperonu R	Number (0-1)
flaperonTrim	Trim flaperonu	Number (0-1)

(Tabulka E.1 Stavové proměnné)

Příloha F – Struktura navigační databáze

Navigační databáze jsou uloženy v adresářích *navData1/* a *navData2/*. Struktury dat uložené v těchto složkách jsou shodné. Každá z těchto složek slouží jako samostatná navigační databáze. Databázi, která bude pokládána za aktivní lze vybrat na stránce IDENT na jednom z CDU. V jednotlivých složkách jsou pak soubory se samotnými daty.

Ident.txt – soubor obsahuje identifikátor databáze a její platnost. Ukázka:

```
787-ND-V0.01
DEC01DEC01/10
```

Airports.txt – soubor obsahuje informace o letištích. V souboru jsou uloženy jednotlivé záznamy v následujícím formátu:

```
<ICAOkód> <jméno> <zeměpisná šířka> <zeměpisná délka> <nadmořská
výška>
```

Ukázka:

```
LKPR PRAHA/RUZYNE 50.100833 14.26 1247
```

Nav aids.txt – soubor obsahuje informace o různých navigačních bodech. V souboru jsou uloženy jednotlivé záznamy v následujícím formátu:

```
<identifikátor> <jméno> <typ> <frekvence> <zeměpisná šířka> <zeměpisná
délka> <nadmořská výška> <využití>
```

Jednotlivé typy jsou popsány v (Tabulka F.1) parametr **<využití>** není v současné verzi použit.

ID	Význam
1	VOR
2	VORTAC
3	TACAN
4	VORDME
5	NDB
6	NDBDME
7	ILS
8	APT
9	DME
10	WPT

(Tabulka F.1 Typy navigačních bodů)

PŘÍLOHA F – STRUKTURA NAVIGAČNÍ DATABÁZE

Ukázka:

DOR	WICKEDE	4	108.65	51.525342	7.631061	373	B
-----	---------	---	--------	-----------	----------	-----	---

Waypoints.txt – soubor obsahuje informace o waypointech. V souboru jsou uloženy jednotlivé záznamy v následujícím formátu:

<identifikátor> <jméno> <zeměpisná šířka> <zeměpisná délka>

Ukázka:

AGEDA AGEDA 51.782947 7.021419

Navigační databáze dále obsahuje soubory *transalt.txt*, *STARs.txt*, *SIDs.txt*, *runways.txt* v současné fázi vývoje však tyto soubory neobsahují žádná data.

Příloha G – Seznam použitých zkratek

Zkratka	Význam
ANA	All Nippon Airways
ATC	Air traffic control
AUX	Auxiliary Display
B788	Boeing 787-8
BITD	Basic Instrument Training Device
CAA	Civil Aviation Authority
CDU	Control Display Unit
COMM	Communication
DEM	Digital Elevation Model
DFDR	Digital Flight Data Recorder
EASA	European Aviation Safety Agency
EFB	Electronic Flight Bag
EFIS	Electronic Flight Instrument System
EICAS	Engine Indication and Crew Alerting System
ETOPS	Extended-range Twin-engine Operational Performance Standards
F/O	First Officer
FAA	Federal Aviation Administration
FFS	Full Flight Simulator
FMC	Flight Management Computer
FNPT	Flight and Navigation Procedures Trainer
FS	Flight Simulator
FSUIPC	Flight Simulator Universal Inter-Process Communication
FSX	Microsoft Flight Simulator X
FTD	Flight Training Device
GLUT	OpenGL Utility Toolkit
GPWS/EGPWS	Ground Proximity Warning System
HUD	Head-Up Display
CHKL	Checklist
ICAO	International Civil Aviation Organization
ILS	Instrument Landing System
INFO	Information
ISFD	Integrated Standby Flight Display
MFD	Multifunction Display
MFK	Multifunction Keypad
MIP	Main Instrument Panel
MS2004/FS9	Microsoft Flight Simulator 2004
MSFS	Microsoft Flight Simulator

PŘÍLOHA G – SEZNAM POUŽITÝCH ZKRATEK

ND	Navigation Display
PFD	Primary Flight Display
PWS	Predictive Windshear System
QMID	Quad Mesh Identifier
QTG	Qualification Test Guide
SID	Standard Instrument Departure
STAR	Standard Terminal Arrival Route
SVS	Synthetic Vision System
SYS	System
TCAS	Traffic Alert and Collision Avoidance System
VSD	Vertical Situation Display
WXR	Weather Radar
XWB	Extra Wide Body

Příloha H – Obsah přiloženého DVD

Obsah přiloženého DVD (Obrázek H.1):

—	readme.txt	- jednoduchý popis obsahu DVD
—	install.txt	- instalační informace
—	index.html	- výchozí stránka projektu
—	text/	- adresář s textem DP
	— DP.pdf	- vlastní text DP ve formátu PDF
	— DP.docx	- vlastní text DP ve formátu DOCX
—	src/	- adresář se zdrojovými kódy programu
	— ...	- zdrojové kódy
—	proj/	- adresář s projektem
	— ...	- projekt ve Visual Studiu 2008
—	release/	- adresář se zkompilevaným programem
	— 787C.exe	- zkompilevaný program
	— ...	-
—	UML/	- adresář s UML diagramy
	— class.xml	- kompletní diagram tříd ve formátu XML
—	results/	- adresář s výsledky profilingu
	— ...	- výsledky profilingu ve formátu CSV

Obrázek H.1 Obsah přiloženého DVD

PŘÍLOHA H – OBSAH PŘILOŽENÉHO DVD