社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－ＳＥ－156 (13)
2007－EMB－ 5 (13)
2007／5／29

MDD2006 チャレンジ報告: 新潮流 チーム

斎藤 司 †

本稿では，MDD ロボットチャレンジにおいて 2004 年から 2006 年にかけて，新潮流チームが飛行船の制御システムに対して行ったモデル駆動開発の過程における，高い再利用性，高いメンテナンス性並びに高い移植性のモデル実現への挑戦の結果を報告する．

Report on MDD Robot Challenge 2006: Team Shinchoryu

TSUKASA SAITO †

This paper reported about a result of challenge about making highly reusable ,highly easy to maintenance and highly portable model for blimp system that was made by team Shinchoryu using the way of model driven development in “MDD Robot challenge” form 2004 to 2006.

1. チームの概要

1．1 チーム構成員

4 名．

1．2 チームメンバー紹介

表1 チームメンバー

氏名	所属	所在地
池 広厚	アルゼ株式会社	東京都江東区
北内 信次	産業技術総合研究所	茨城県つくば市
斎藤 司	ニューウェイブシステムズ株式会社	東京都墨田区
関本 渡	ニューウェイブシステムズ株式会社	東京都墨田区

チームメンバーは基本として，ＥＴロボコン２００６の新潮流チームと同じであるが，一部のメンバーが入れ替わっている．池と斎藤は，組み込みシステム開発１０年超のベテランである．斎藤はここ数年オブジェクト指向手法による組み込みシステム開発と技術コンサルテーションをおこなっている．北内，関本はソフトウェア開発歴１年未満の新人である．

1．3 担当

２００５年，２００６年ともモデリングは斎藤一人で行った．池は飛行船の動作テストを主に担当した．実装は斎藤，北内，関本で分担しておこなった．

1．4 進行スケジュールと計画

2004 年から 2006 年にかけて，新潮流チームは三回にわけて，反復型の開発をおこなってきた．毎年仕様が若干変わっているが，その都度その差分を開発し直すことで，対応してきた．表2にこの3年間の取り組み内容を記載する．

表2 3年間の取り組み

年 度	取り組み工程	説 明
2004 年	要求，分析，実装	時間不足で設計モデルは作成できなかった．C言語で実装．
2005 年	要求，分析，設計，実装	2004 年の分析モデルをもとに仕様の差分を反映してモデルを洗練し，設計モデルを追加．
2006 年	要求，分析，設計，実装，テスト	2006 年はさらに仕様の差分をモデルへ反映し，C++で再度実装し直し，テストまで行った．同時に，モデルのメンテナンス性等を評価．

表3 2006 年開発活動の概要

年 度	日 程	予 定	実 績
2006年	9月 16 日～9月 18 日	分析・設計モデリング	同左
	9月１８日～30 日	実装	同左
	10月1日～当日	動作テスト	実装・テスト

表3に2006年の開発活動の概要を示す．2006年においては，プログラムの搭載先を飛行船から基地局に移動させた．一部分をこれでまでのC言語のソー

†ニューウェイブシステムズ株式会社 東京事務所(本社:新潟市)
New Wave System Co, Ltd. Tokyo branch (Head office: Niigata City).

スを捨て，新にC＋＋で再度実装を行った．

2006年は2004年，2005年に比して実装を担当するメンバーが増えたことで，C＋＋で再実装をおこなったが，かなりのレベルで仕上げることができた．しかし，後一歩というところで，間に合わせることができず飛行は，できたところまでの機能での飛行になった．

2．2006年度の報告の対象

2．1 動機

一般的に組み込みシステム開発において業務で行なった取り組みは企業機密になるため，一般に公開されることはれであり，そこで適用されている具体的なノウハウを公開することは困難である．

当初より当チームリーダの斎藤が日頃の業務で取り組んでいる，オブジェクト指向を使ったモデル駆動な開発を，飛行船の航行制御システムに適用してそのノウハウを公開することを目指してきた．

しかしこれから述べることならびに，取り組みの結果は，組み込みシステムをオブジェクト指向で開発をしている現場で通常に取り組まれていることであり，それほど目新しいことではない．組み込みシステム開発においてそれらが，具体例をともなって一般に公開されにくくノウハウが普及しにくいということからこのような形で公開にすることに意味があると考えて取り組んで来た．

2．2 オブジェクト指向開発のメリット

モデル駆動な開発のメリットは，対象ドメインをモデルで分析設計し開発対象を見えるようにして，その複雑さを低減して管理することにある．オブジェクト指向によるメリットはそれに加えてモデルのメンテナンス性や再利用性並びに移植性を向上し総合的な開発効率を上げることにあると考えられる．構造化手法においてはモデルのメンテナス性は低く，モデルの移植性，再利用性はオブジェクト指向を使った方法ほど高くない．

この間の取り組みではモデルにおける高い再利用性，高いメンテナンス性並びに高い移植性を実際に実現して，その具体例を公開する事に挑戦してきた．そして，この3年間の取り組みのなかで，その一部の成果が確認することができた．

2．3 MDDの各工程とのマッピング

オブジェクト指向を使った開発においても要求分析→分析モデリング→設計モデリング→実装→テスト　の各ステップをへて，システムを構築するが，要求分析をＭＤＤにおけるＣＩＭ（Computation-independent model）工程に，分析モデリングをＭＤＤにおけるＰＩＭ（Platform-independent model）工程に，設計モデリングをＭＤＤにおけるＰＳＭ（Platform-specific model）工程と位置づけて取り組みを行なってきている．

2．4 成果物

この間作成した成果物を表4に示す．

表4　2006年の成果物

工程	成果物	備考
要　求 分析	コンテキストダイアグラム ユースケース図 ユースケース記述 概念モデルクラス図	CIM
分析	パッケージ図(ドメインチャート) 分析モデルクラス図 シーケンス図 状態図	PIM
設計	設計モデルクラス図 シーケンス図 状態図	PSM
実装	ソースコード	

2．5 モデリング環境

2．5．1 適用モデリング手法

日ごろから使い慣れているＵＭＬ１．４[1]でモデリングをおこなった．

2．5．2 使用CASEツール

CASEツールはRose Professional C++ Edition 2001.A.04.00[2]を使用したが，実装担当者の実装スキルの向上の為あえてスケルトンの自動生成機能は使用せず手で実装をおこなった．

3 モデリング，実装の方針

3．1 モデルの再利用性とは

モデルの再利用性とは，同じモデルを異なったアプリケーションに対してどれだけ少ない努力で適用できるかということととらえることができる．極論をいうと，ＭＤＤロボットチャレンジにおける飛行船の制御で抽出されたモデルが，ＥＴソフトウェアデザインロボットコンテストのパスファインダーのモデルで使用することができたら，極めて高い再利用性があることになる．ここではそこまでの事はできていない．ただし，当チームの取り組みでは，ＭＤＤロボットチャレンジと，ＥＴソフトウェアデザインロボットコンテストにおいて，非常に近いモデルの共通の構造がパターンとして抽出できている．ＭＤＤロボット

チャレンジ２００５の参加報告[6]において若干ふれさせていただいている．

3．2 移植性とは

移植性とは，あるドメインで作成されたモデルが，その搭載先のプラットホームを選ばずに適用できるかという事を指していると言える．今回の取り組みで具体的には，ＯＳのない１６ビットマイコン上に搭載したモデルとソースコードがどれくらい修正を加えずに，ＷｉｎｄｏｗｓのＰＣ上へ搭載先を変えることができるかということである．

3．3 モデリングの方針

3．3．1 ドメイン分割

ＣＩＭで作成した概念モデルクラス図をもとにドメイン分割するときに，関連の深い概念をまとめて一つのドメインとするようにドメインを分割する．そのことで，ドメイン間で概念の重複がなく，ドメイン間の関連が疎になり，ドメインの独立性（凝集性）がよくなる．概念の重複は，修正時にそれぞれに対して同時に修正が発生し，メンテナンス性を下げることになる．また他に依存するために，一つのドメイン単位で再利用ができず，依存するドメインもあわせて再利用することになりがちであり，ドメイン単位での再利用性を下げることになる．同じ概念が分散するとそれぞれの整合性をとるために，無駄に通信が発生することにもなる．

3．3．2 分析と設計の分離

分析モデルでは何をしたいのか（ＷＨＡＴ）の分析に着目する一方で，実現手段（ＨＯＷ）を排除して取り組んだ．そのことで，実現手段や，プラットフォームに対して独立したモデル構築することに努めた．一方設計フェーズでは，実際に動かすために必要な実現手段や，プラットフォームに依存する部分をモデルとして追加する形でモデルを構築した．しかしこれは，すでにＭＤＤにおける，ＰＩＭと，ＰＳＭの分離で実現されていることでもある．

3．3．3 実現手段単位でのパッケージ化

本来であれば，ハードウェアになどに依存するところは，ハードウェアが変わるとそのために，改造を余儀なくされるが，その変更ができるかぎり局所化されて，すでにあるドメインに対して影響を与えないようにすることが望ましい．そのような，実現手段としてのハードウェア依存部をそのハードウェアの単位でパッケージとして分離しておくことで，その部分を置き換え可能にすることを目指した．同時に抽象化したインターフェースを定義することで，置き換えが容易になるようした．具体的には，飛行船と基地局の通信手段に関わる部分と，座標の計算に関わる座標データを取得する部分をパッケージとして分離している．

3．3．4GoFのデザインパターン[3]の適用

ファザードパターンを推進メカニズムドメインに適用した．このことで，航行制御からみた推進メカニズムのインターフェースが１つにしぼられることになり，推進メカニズムのドメインと，航行制御のドメインの置き換えが容易になるようにした．

4．仕様の相違

図に２００５年と２００６年の仕様の相違点を表５に示す．

表５　仕様の相違点概要

項目	２００５年	２００６年
座標取得	上部カメラ画像から主に取得，場合に応じて超音波ソナーのデータからも取得	超音波ソナーデータから取得
方位取得	上部のカメラ画像から取得	飛行船内部のジャイロセンサーにより取得
通過点	２点	３点
推進力	前後，上下，回転，左右	前後，上下，回転
制御プログラム搭載先	飛行船側マイコン	基地局ＰＣ
使用したゴンドラ	自前	大会事務局が用意したもの

4．1 座標と方位の取得方式

２００５年と，２００６年で大きく異なる点は，座標の取得方式と，方位の取得方式が変わった点である．２００５年は，上部にカメラが設置されその映像による画像認識により，飛行船の座標と方位を取得する事ができた．ＬＡＮ経由で，座標と方位の計算されたデータを取得する事ができ

た．また，同時に，超音波ソナーのある領域では，超音波ソナーによる伝搬時間差のデータを取得することができ，それを加工して飛行船の座標をもとめることもできた．２００６年は，上部カメラによる画像がなく，座標データは超音波ソナーからのデータにより求めることになった．

方位は，２００５年上部画像カメラからの座標の両方を使って求めたが，２００６年は飛行船に搭載されたジャイロセンサーで計測して求めることに成った．２００６年は，事務局の飛行船ハードウェアを使用しため，方位の取得と高度の取得は飛行船側のマイコン側で計測したデータを使用することになった．

4．2　推進力

推進力としてのプロペラは，２００５年は，上下方向，左右，前後方向すべて各２つのプロペラがあった．回転は，前後方向の左右２つのプロペラを反対方向に回転させることで実現した．

２００６年は，大会事務局側のゴンドラを使用したので，上下方向は１つのプロペラで，前後，回転用に左右２つのプロペラがあるのみで，左右方向のプロペラはない．

4．3　プログラムの搭載先

２００５年はプログラムを飛行船側に搭載したが，２００６年は事務局側のゴンドラを使用した関係で，プログラムは基地局ＰＣ側に搭載した．

4．4　開発対象範囲

２００５年は，飛行船の航行制御と基地局ＰＣのシステムを対象とした．航行制御，飛行船側プロペラ制御，高度取得，超音波によるＸＹ座標取得を対象範囲とした．無線通信，画像による座標計算，ソナーによるデータ取得は対象外とした．

２００６年は基地局ＰＣ上の航行制御機能を対象とした．無線通信，方位取得，高度計測，プロペラへの出力制御，ソナーによるデータ取得は対象外とした．

4．5　航行制御方式

4．5．1　2005年の航行制御方式

図1に2005年における航行実現方式を示す．2005年において，経過点を直線で結ぶ経路を航行しようとした．目標地点の上空までは，水平方向の座標として会場上部に設置された画像によるやや精度の低い座標を使って移動し，目標地点の上空でより精度の高いソナーによる座標を使って位置合せを行う部分がある．

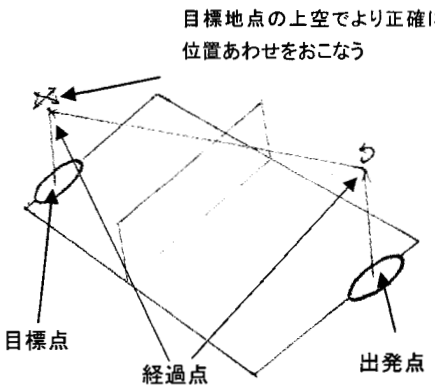

図1：2005年航行実現方式

4．5．2　2006年の航行制御方式

図2に2006年における航行実現方式を示す．2005年同様に，航行は垂直，回転，水平移動の運動で航行を実現する．座標の取得は，上部画像カメラがない関係で，ソナーからの座標データによる．目標点の上部での位置合わせは，無くなっている．

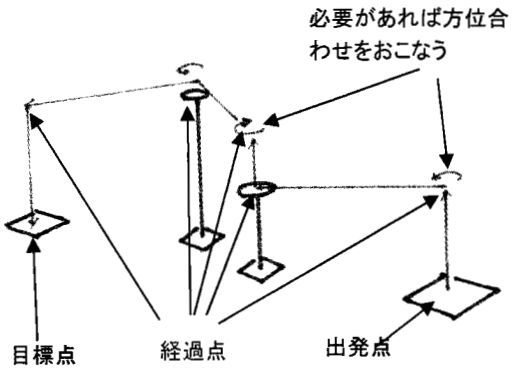

図2：2006年航行実現方式

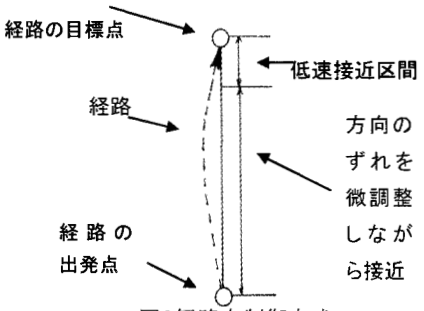

図3経路内制御方式

4．5．3　経路内制御方式

図3に経路内の制御方式を示す．経過点間を結ぶ経路内では目標に向けて回転と直進で制御するものとする．次に一連の手順を述べる．こ

れは2005年，2006年変っていない．

① 経路の出発点から目標点に向けてまず方位を合わせる回転を行う．

②次にフル出力で前進を行い，途中目標地点からのずれを前進しつつ角度を微調整して目標点まで接近する．

③目標点に一定の距離に近づいた時点で出力を下げてゆっくりと接近する．

④上昇，下降の経路の場合は方位合わせと，方位微調整がないだけで後は水平の経路と同じ．

5. モデリング結果

5. 1 CIM

要求モデルとしてユースケースを作成したが，紙面の関係でユースケースは割愛させていただく．ユースケース記から抽出された概念を整理して関連づけて作成された2006年の概念モデルクラス図を図４に示す．ここでは，分析モデリングでのドメイン分割の目安として，関連の深い概念をグループ化してドメイン候補を抽出している．

5. 2 PIM(分析モデル)

ＰＩＭであるところの分析モデルはここでは，先に述べた方針ように，概念モデルクラス図に対して関連の深い概念のまとまりで，ドメインを分割しており，その結果各ドメインの凝集性を高くすると同時に同じ概念がドメイン間で重複することを避けている．2005年，2006年の分析モデルクラス図を図５，図６にそれぞれ示す．

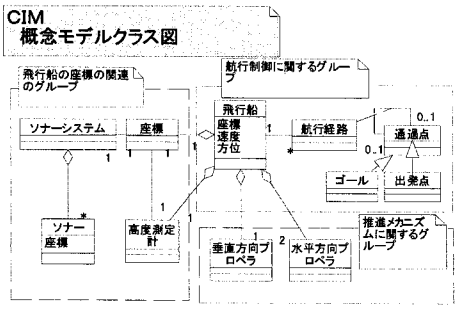

図4:2006 年概念モデル

5. 2. 1 モデルの比較とメンテナンス性

2005年の分析モデルと2006年の分析モデルを次に比較しそのメンテナンス性を検証する．

(1)航行制御ドメイン

航行制御ドメインは，左右方向のプロペラに関する機能が無くなったことから，「目標上空位置合わせナビゲーション」，「目標上空位置合わせ経路」の２つのクラスがなくなっただけで他は変更無し．これらを削除しても，これらはポリモリフィックに抽象化された親クラスを継承している関係で当ドメインのほかのクラスへの影響はなく，基本的なクラス構造は変化していない．

(2)推進メカニズムドメイン

推進メカニズムドメインは，「左右方向推進器」クラスがなくなっただけで「推進器クラス」を除く他のクラスには影響を与えていない．推進器クラスのインターフェースから，左右方向に関するメッセージが削除されている．

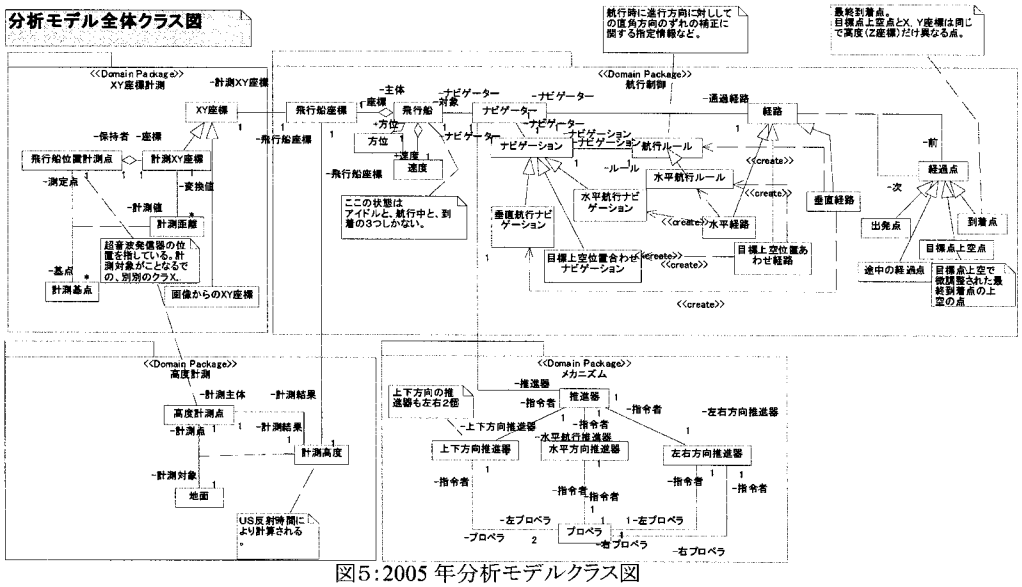

図5:2005 年分析モデルクラス図

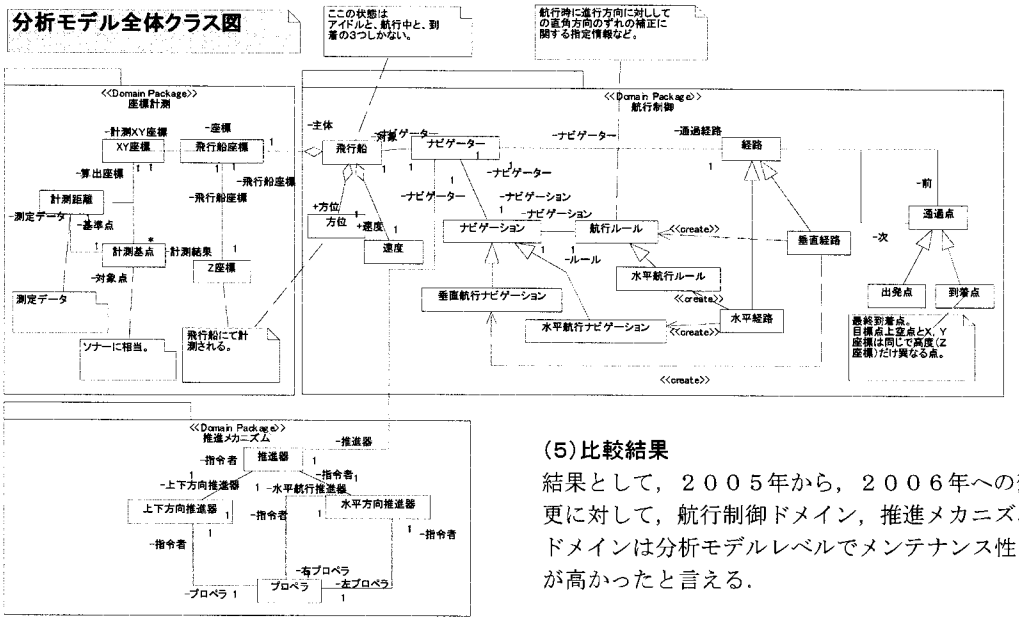

図6:2006 年分析モデルクラス図

(3)座標計測関連ドメイン

座標計測ドメインは一部のクラス構造が変更されている．２００５年では，航行制御ドメインに所属していた，「飛行船座標」クラスを移動して，ＸＹ座標を求めドメインから，飛行船の全ての座標を求めるドメインに再構成している．画像による，ＸＹ座標がなくなったことと，高度の測定を飛行船からの情報にゆだねることになったため，再度構成をし直した．２００５年飛行船の高度の測定も対象であったため，高度計測ドメインがあったが，２００６年は対象外になったため，無くなっている．

これらの変更は座標に関するドメインに局所化されており，座標に関する変更は航行制御ドメイン，推進メカニズムドメインに影響をあたえていない．

(4)動的な側面

モデルの動的な側面として図７に水平ナビゲーションクラスの状態図を示す．これは，２００５年，２００６年とも変化はしていない．ほかの共通クラスの動的な側面もほぼ変更が必要なかった．相互作用図はここでは紙面の関係で割愛させていただく．

(5)比較結果

結果として，２００５年から，２００６年への変更に対して，航行制御ドメイン，推進メカニズムドメインは分析モデルレベルでメンテナンス性が高かったと言える．

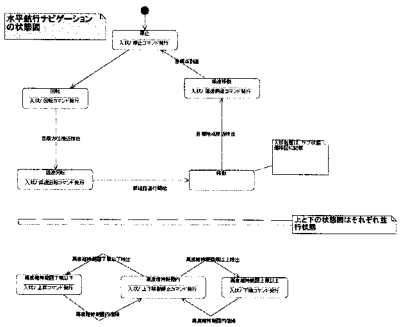

図7:水平ナビゲーションクラス状態図

5．2．2モデルの移植性

航行制御ドメインと，推進メカニズムは，プラットフォームに依存していなかったため，飛行船側のマイコンから，基地局ＰＣへ搭載先が変更されても，移植による影響はなかった．

また，座標計測に関するドメインは変更が入ったが，それらはプラットフォームによる変更でなく，座標の計測の仕方の違いに起因して発生している．ＰＩＭの目的とするプラットフォームに依存しない分析モデルを達成できていたことにもなる．

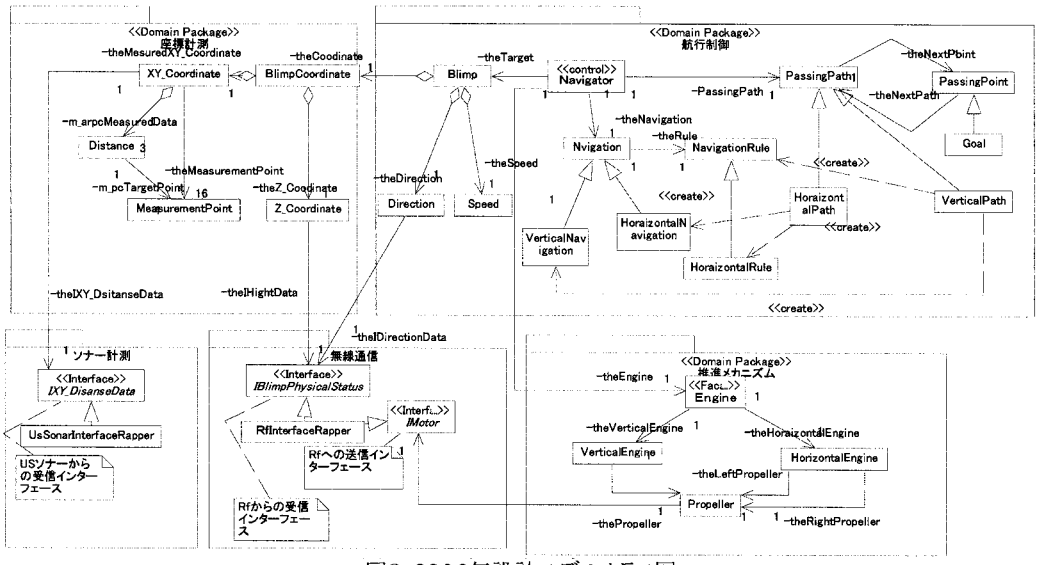

図8:2006年設計モデルクラス図

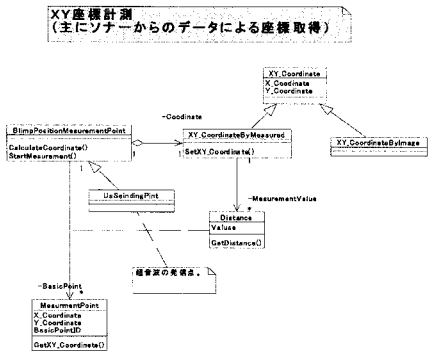

図9:2005年XY座標計測ドメイン設計モデルクラス図

5.3 PSM(設計モデル)

ＰＳＭであるところの２００６年の設計モデルのクラス図を図８に示す．基本として，２００５年と２００６年では分析モデルとほぼ変わらない変更点である．状態図においても同じである．状態図，相互作用図は紙面の関係でここでは割愛させていただく．座標関連ドメインは，分析モデルの変更を反映して，２００５年と２００６年はことなったものと成っている．

5.3.1設計で追加されたパッケージ

プラットフォーム依存部と，無線通信に関連する部分を，あらかじめインターフェースをきめておき，独立したパッケージとして後に追加することで，対象部分の置き換えを可能としている．

今回は，ソナー計測と，無線通信のパッケージが追加されている．

5.3.2 メンテナンス性

メンテナンスに関しては，航行制御ドメイン並びに推進メカニズムドメインは，設計モデルでも分析モデルと同じ内容の変更のみにとどまり，極めて少ない変更内容にとどまっている．

本来同じ言語での実装であれば，ソースコードレベルでも極めてすくない変更であった事と推測される．残念ながら２００６年対象範囲のコードをＣ言語からＣ＋＋での再実装にしたため，コードレベルでのメンテナンス性を数値で示すことができない．

一方座標計測関連ドメインは，分析モデルでの相違を反映して，設計以降は作り直しに成っている．

5.3.3 移植性

航行制御ドメイン並びに推進メカニズムドメインは，設計モデルとしては，プラットフォームに依存する部分がふくまれていないため，プラットフォームに関する変更を加えることなく移植できた．もし，同じ言語であれば，ソースレベルでもプラットフォームに依存しないで移植ができたことになる．航行制御ドメインと，推進メカニズムドメインであるが，これらは両方を同時に飛行船から，ＰＣへ動かす事がかならずしも必要

なく，航行制御のみ基地局ＰＣへ移動することが容易に可能になっている．それは，推進メカニズムのEngineクラスと同じ，仕様のインターフェースを定義して，そのインターフェースを実装する無線通信ラッパークラスを作ることで，航行制御と，推進メカニズムクラスの間を無線通信でやりとりさせることが出来るからである．その意味で，航行制御と，推進メカニズムのそれぞれも個別に移植できることになり，独立性が高いといえる．

6. 考察

6.1 モデルの再利用性

ここまでで，今回の取り組みで高いメンテナンス性，移植性を実現することができたことが確認された．残る目標は，モデルの再利用性であるが，現時点では，まだ今回抽出されたモデルが他のアプリケーションに対して具体的に適用できるところまでは確認されていない．

しかし，航行制御ドメインで抽出した，「飛行船」～「ナビゲーター」～「経路」～「通過点」のクラスが構成する部分は一つのフレームワークを形成しており，同じような構造をもつ対象アプリケーションへの適用が可能であることが分かっている．先にものべたが新潮流チームにおいては，ＥＴソフトウェアデザインロボットコンテスト２００５に於けるパスファインダーのモデリングの結果，この飛行船のモデルに大変近いモデルの構造が抽出されている．２００５年の新潮流チームのＭＤＤ参加報告[6]に報告させていただいた．

7. まとめ

昨年に比較して，今回はハードウェアを事務局が提供してくれたものをそのままつかったため，ソフトウェアの開発のみに集中することができた．また開発要員も一部入れ替わる代わりに，実装を担うことができるメンバーが増えた関係でＣ＋＋で再度実装をし直したが，かなりのところまでこぎ着けることができたのは幸いである．当初から，目指していたモデルレベルでの再利用はまだ確認されていないが，高いメンテナンス性，ドメイン単位での高い移植性を実現できたことが確認された．

今回も例年同様プログラムを完成させることができなかったが，今後完全させ目指す動きを実現きるようにしてゆきたい．

今後モデリングの技術として最近注目され始めている，ＤＳＬ（ドメイン特化言語）やソフトウェアファクトリィズ[7]などの新技術を取り込んで，その技術の組み込みシステム開発への適用の可能性を模索して行きたい．

参考文献

[1] 竹政 昭利 著 「はじめて学ぶUML」 ナツメ社, 2004
ISBN4-8163-3411-4

[2] IBM, Inc.: Rational Rose Professional C++ Edition 2001.A.04.00,

[3] エリックガンマ他著「オブジェクト指向における再利用のためのデザインパターン」 ソフトバンククリエイティブ, 1999
ISBN4-7973-1112-6

[4] 渡辺博之他 「組み込み UML - eUML によるオブジェクト指向組み込みシステム開発」 著 翔泳社, 2002
ISBN4-7981-0214-8

[5]斎藤司著 「MDD ロボットチャレンジ 2004 参加報告:New Wave System チーム」 MDD ロボットチャレンジ2004報告集, 2004.
ISBN4-915256-56-1 C3040

[6] 斎藤司著 「MDD ロボットチャレンジ 2005 参加報告:新潮流チーム」 MDD ロボットチャレンジ2005 報告集, 2005.
ISBN4-915256-61-8 C3040

[7] Jack Greenfield,Keith Short 他著 「ソフトウェアファクトリィー」 日経 BP ソフトプレス, 2005.
ISBN4-89100-472-X