

平成12年度産業科学技術研究開発制度
新エネルギー・産業技術総合開発機構委託事業

ヒューマンメディアの研究開発
石油プラント高度情報化システム技術研究開発
成果報告書

平成13年3月

NEDO 図書・資料室 財団法人イメージ情報科学研究所



010017532-2

「ヒューマンメディアの研究開発
石油プラント高度情報化システム技術研究開発」

財団法人イメージ情報科学研究所

平成13年3月 全193頁

研究開発目的

本研究開発は、人間と一体感のある情報環境の実現を目指したヒューマンメディア技術の研究開発により、石油プラントの安全操業および効率的運転を支援する情報基盤システムを研究開発することを目的とする。

商標

本報告書に掲載されている製品名、システム名、会社名は、一般に各開発メーカーの商標や登録商標であり、特に本文中には、TM、コピーライト表示は明記していません。

まえがき

ヒューマンメディアの研究開発は、平成6年度、7年度の調査研究を踏まえて、平成8年度にスタートした。次世代プラント用ヒューマンインタフェースの研究は、大規模で複雑なプラントシステムの安全操業と効率的運転に資するため人間中心の情報環境を構築することを目指した。我が国の人口動態は急速に高齢化しており、プラント運転員の場合も高齢化により、プラントの建設段階から立ち会ってきたベテラン運転員がリタイアする中で、安全が確保され自動化されて運転されているプラントシステムに接する若年運転員には、異常時の経験がほとんどない状態で、運転ノウハウも少ない。このような時代の趨勢に対処するにはシステム側の機能を高めて人間の情報処理に近づける技術が必要になる。人間と一体感のある情報環境を実現するヒューマンメディアの研究開発は以上のような社会的経済的な問題と密接に関係する研究課題である。

本年度は5年間の研究開発の最終年度であり、実証実験システムを開発し、ユーザサイドからの評価も受けた。今回開発のシステムは、(実プラントとの接続はなかったが)シミュレータからプラント状態の異常を検出するやエージェントと運転員が自由に音声対話し、大画面には必要な情報が優先度に従って時々刻々提示される。その背後には、プラントの構造、振る舞い、機能の知識が体系化モデル化され分散協調処理されているのである。これまでの運転員がDCS(分散コンピュータシステム)画面を見ながら運転ノウハウに基づいて意思決定した運転インタフェースを一新するものである。

本研究開発は石油プラント高度情報化システム技術研究開発の一環として実施され、石油精製プラントをターゲットにした。システム開発にあたってはプラントインタフェースの「ミドルウェア化」を一つの目標に掲げており、極力汎用性を持たせ特殊な技術を用いることはしなかった。そのため本技術は、石油、鉄鋼、電力、LNG、セメントなど多種多様なプラント運転インタフェースに適用できることが期待できる。本成果が次世代プラント運転ヒューマンインタフェースの情報基盤システムとして実用化され安全性の向上と経済産業の発展につながることを願って止まない。

目次

まえがき

1.	研究開発の概要.....	1
1.1.	研究開発の目的	1
1.2.	研究開発の内容	1
1.2.1.	インタフェースエージェント	1
1.2.2.	仮想プラント表示インタフェース	1
1.2.3.	分散協調処理技術	1
1.2.4.	プラントモデルとインタフェースシステム	2
1.2.5.	プラントオントロジー	3
1.2.6.	意味表示インタフェース	3
1.2.7.	大画面統合表示技術.....	4
1.3.	研究開発の成果	4
1.3.1.	インタフェースエージェント	4
1.3.2.	仮想プラント表示インタフェース.....	4
1.3.3.	分散協調処理技術	5
1.3.4.	プラントモデルとインタフェースシステム	5
1.3.5.	プラントオントロジー	7
1.3.6.	意味表示インタフェース	7
1.3.7.	大画面統合表示技術.....	8
1.4.	体制.....	9
1.5.	Summary of FY2000 Human Media Technology R&D:Advanced Information System Technology R&D for Oil Refinery Plant	12
1.5.1.	Overview of the R&D.....	12
1.5.2.	The background of the art of the plant operator support.....	14
1.5.3.	System organization.....	17
1.5.4.	Contents and Results of the FY2000 R&D	21
2.	成果の詳細.....	25
2.1.	全体システム	25
2.1.1.	Co-operator モデルとプラント問題解決プロセス	25
2.1.2.	システム構成.....	28
2.1.3.	サブシステムの機能.....	30
2.1.4.	異常時における自律処理	31
2.1.5.	情報統合提示.....	33
2.1.6.	結果および考察	40

2.2.	インタフェースエージェントの研究	42
2.2.1.	本年度研究の概要	42
2.2.2.	実用性向上	43
2.2.3.	ミドルウェア化への対応	46
2.2.4.	DCS エージェント	51
2.2.5.	残された問題点と今後の課題	53
2.3.	仮想プラント表示インタフェースの研究	54
2.3.1.	自律化機能の充実	54
2.3.2.	ミドルウェア化	66
2.3.3.	ユーザ評価	77
2.4.	分散協調処理技術の研究	79
2.4.1.	機能、性能の評価	79
2.4.2.	分散協調処理機構の改良	83
2.4.3.	融合システム開発とそのフィードバック	90
2.5.	プラントモデルとインタフェースシステムの研究	91
2.5.1.	本年度の概要	91
2.5.2.	異常シナリオの内容充実とシミュレータでの検証	91
2.5.3.	各サブシステムの検証	91
2.5.4.	類似システム調査	92
2.5.5.	ユーザー評価	92
2.6.	プラントオントロジーの研究	108
2.6.1.	オントロジーサーバー	108
2.6.2.	プラントモデルの改善と構造検索	122
2.7.	意味表示インタフェースの研究	145
2.7.1.	異常原因推定手法の開発	145
2.7.2.	運転知識と高速シミュレータによる対応操作の選定手法の開発	148
2.7.3.	表示意図に基づく意味表示の実装	154
2.7.4.	最終プロトタイプシステムの開発	155
2.7.5.	Java 版プラントモデル構築ツールの整備	156
2.8.	大画面統合表示システム構築のためのガイドライン作成	158
2.8.1.	プラント運転における各種異常シナリオの調査・分析	158
2.8.2.	大画面統合表示パターンのロジックの策定	160
2.8.3.	大画面統合表示インタフェースのコンセプトモデルの策定	166
2.8.4.	大画面統合表示システム構築のためのガイドライン	172
3.	目標に照らした達成状況	175
3.1.	全体システム	175

3.2.	インタフェースエージェントの研究	176
3.2.1.	目標の達成度とその要因	176
3.2.2.	成果が与えるインパクト	177
3.3.	仮想プラント表示インタフェースの研究	178
3.3.1.	実用化までの道のり	178
3.3.2.	プロジェクト終了後の成果の各社での活用計画	178
3.4.	分散協調処理技術の研究	179
3.4.1.	パフォーマンスの改善	179
3.4.2.	安定性の向上	180
3.4.3.	ミドルウェアの整備	180
3.5.	プラントモデルとインタフェースシステムの研究	182
3.5.1.	シナリオ作成・検討	182
3.5.2.	意味表示インタフェース用データ収集・評価	182
3.5.3.	プラントオントロジー用データの収集・評価	183
3.5.4.	VR 用データの収集・評価	183
3.5.5.	運転用インタフェースエージェント用データの収集・評価	183
3.5.6.	大画面の設計・評価	183
3.6.	プラントオントロジーの研究	184
3.6.1.	成功の要因	184
3.6.2.	成果が与えるインパクト	185
3.7.	意味表示インタフェースの研究	186
3.8.	大画面統合表示システム構築のためのガイドライン作成	188
4.	海外の技術動向調査	189
4.1.	ISRA'2000	189
5.	研究発表・講演、文献、特許等の状況	191
5.1.	研究発表・講演	191
5.2.	文献	193
5.3.	特許等	193
5.4.	その他の公表（プレス発表等）	193

1. 研究開発の概要

1.1. 研究開発の目的

石油プラントの安全操業を目的として、計器情報や画像情報の知的なモニタリング技術、運転員によるプラントの運転状況把握を容易にするための表示・ガイダンス技術、過去のプラントデータを効率的に管理し現在のプラントの運転状況の判断に有効に利用する技術、上記システムを実現するためのプラントのモデル化技術、自律分散協調処理技術などの研究開発を行う。

1.2. 研究開発の内容

本年度は、平成11年度までに研究開発した成果であるところの計器情報や画像情報の知的なモニタリング技術、運転員によるプラントの運転状況把握を容易にするための表示・ガイダンス技術、過去のプラントデータを効率的に管理し現在のプラントの運転状況の判断に有効に利用する技術とこれら技術を融合したプロトタイプシステムの個々の要素技術の改良と汎用化のための研究開発ならびに、上記システムのみドルウェア化のための整備と改良を行う。

1.2.1. インタフェースエージェント

融合化システムによる実用性の評価を目的として、複数異常シナリオにシステムを対応させ、プラント監視機能の実用性評価を行う。また、音声対話システムを運用評価し、性能改善向上を行う。本研究成果のみドルウェア化を目的として、事例ベースエディタ開発、WebブラウザXMLを用いた事例ベース推論システムのエンドユーザ用GUIツール開発を行う。

1.2.2. 仮想プラント表示インタフェース

プラント異常の複数シナリオを対象とし、実用性評価と性能の改善・向上を行う。そして、仮想プラント表示の自律機能の拡張を行う。仮想プラント表示のみドルウェア化対応を進めるために、ヒューマンインタフェースに特化した汎用VRアプリケーションの開発とともに可視化プロセスデータの拡張を実施する。

1.2.3. 分散協調処理技術

平成11年度に開発した分散協調処理機構の機能・性能について、融合システム上で評価を行い、改良項目を検討する。具体的には、協調処理機構の速度

性能の評価、協調処理機構の自律性の評価と改良項目の検討、サブシステムの同期メカニズムの検討などを行い、その結果をもとに、分散協調処理機能のアーキテクチャの見直しを行って改良項目を決定する。

上記の検討結果に基づいて分散協調処理機構の改良、およびミドルウェア化を行い、その評価を実施する。

1.2.4. プラントモデルとインタフェースシステム

本研究の最終年度であり、各々の要素技術及びその分散協調による総合的なプラントヒューマンメディアとしての評価を、下記の視点で行う。

- ・運転員から見たシステムの使い易さ、情報提示のタイミングの適切度
- ・プラントの設備改造などの変更に対する個別要素技術のメンテナンス性
- ・今回対象とした装置以外の操業形態の異なる装置（例：出荷設備、動力設備）への適用の可能性

①最終評価

ユーザーの立場から、このシステムが実用に向けて有効性の評価を行う。
また、実用化に向けての問題点など、指摘、提言を行なう。

②シナリオの充実

シミュレータで再現したシナリオはまだ2件である。オペレータが気づきにくい異常、各要素技術間で多くのデータのやりとりが必要な異常、1原因で複数の異常が発生するものやシナリオ想定外での異常対応など、より複雑なシナリオを作成し、このシステムが多様な異常に対応可能か検討する。

③簡易プラントモデル構築のサポート

多様な異常に対応するには、更に経験式を折り込む必要がある。シナリオに応じて、モデルの挙動に対する経験式を提供する。

④大画面設計

オペレータには大きな画面を提供し、多くの必要な情報を、必要とする形で提供する。そのため、大画面の有効な使用方法を検討する。

⑤プラントオントロジー言語登録・出力機能評価

プラントを構成する機器ならびに運転に関わるデータや語彙の充実をはかるべく情報提供を行うとともに、継続して語彙を充実できるなど実用面からの評価を行うため、語彙登録サポートを行う。

⑥運転用インタフェースエージェント用データの収集・評価

シミュレータと接続を行うインタフェースエージェントに対し、実用性評価のためにプラント履歴データなどの情報提供を行い、実プラント運転面からの機能、性能評価を行う。

⑦VR用データの収集・評価

プラント情報の可視化など、仮想プラント表示構築に必要な機器データの提供を行い完成度を高める。また、個別要素技術の開発内容の評価を行う。

1.2.5. プラントオントロジー

①協調プロトコルとオントロジーの洗練

平成11年度に開発した運転タスクオントロジーを拡張して、さらに高度な協調を可能にする協調方式を実現するために必要なタスクオントロジーと協調方式を開発する。

②融合化の更なる促進と有用性の検証

対象とするシナリオの数を増やして、より厳しい状況のもとでの動作の評価と、そこでのオントロジーサーバの有用性の検証を行う。特に、想定外の状況における動作検証に重点をおく。

③ミドルウェアとしての整備

特別な訓練を必要としないで、専門知識さえあれば容易にオントロジーの拡張と利用法（メッセージ生成を含む）の改良が可能になるような開発環境を整備する。

1.2.6. 意味表示インタフェース

次世代のプラントの運転を支援するため、異常時に異常原因の絞込、対応操作候補の導出や対応操作の効果の評価といった診断情報を機能、挙動、構造の観点から表示するためのヒューマンマシンインタフェースを開発する。

①表示意図に基づく意味表示の実装

平成11年度検討した表示意図に基づく意味表示手法を機能一目標レイヤーの可変詳細度表示として実装する。

②高速シミュレータによる対応操作の選定手法の開発

対応操作の効果の検討のための高速簡易予測シミュレータを完成し、定性推論と定量推論を組み合わせた対応操作候補評価手法を開発する。

③Java版プラントモデル構築ツールの整備

意味表示インタフェースのベースとなるプラントモデルを構築するためのツールのJava版を開発する。

④最終プロトタイプシステムの開発

大画面表示や追加シナリオに対応した最終プロトタイプシステムを開発する。

1.2.7.大画面統合表示技術

各種シナリオに対する大画面統合表示パターンを検討し、大画面統合表示に関する合意形成のアルゴリズムを導出すると共に、大画面統合表示のためのガイドラインを作成し、次世代プラント用ヒューマンインタフェースのシステム構築に資する。

1.3.研究開発の成果

1.3.1.インタフェースエージェント

本年度は前年度融合化研究に続き、前年度成果を実用性とミドルウェアの観点から評価改良を行った。インタフェースエージェントの機能（プラント監視機能、音声対話コマンド機能、プラントデータ提供機能）を分離させ、それぞれプラント監視エージェント、音声対話モジュール、DCSエージェントの一部としてのプラントデータ提供機能に独立させた。こうすることによって、将来一部の機能のみを利用することが容易になる。

プラント監視エージェントでは、前年度二つであった複数異常への対応を、現在シミュレーション可能な異常のすべてである七つの異常に対応拡張した。これにより事例に基づくプラント監視方法が実用化に一步近づいた。

音声対話モジュールでは、前年度開発した音声対話機能をC++言語を用いて実用性を高め、本年度開発の全体システム（フェーズ3）に組み込むことができた。これによりObjectStoreを用いなくても、音声対話コマンド機能のみを任意の他システムに組み込むことが可能になり、具体的な実用に供することが可能になった。

本年度新しくDCSエージェントを分離独立させ、これにプラントデータ取得・蓄積・検索・提供機能を実現した。これらは従来インタフェースエージェントが不十分ながら行っていたものであるが、新しくリレーショナルデータベースを導入し、機能として十分なものにした。さらに、DCSエージェントに種々のデータ監視機能も追加し、ミドルウェアとしての強化を図った。

ミドルウェア開発の一環として、事例データベースの蓄積内容を通常のWEBブラウザから閲覧できる機能を開発した。また、事例データベースの内容を編集できるGUIエディターを開発した。

1.3.2.仮想プラント表示インタフェース

本年度は、前年度の自律化機能をさらに拡充させ、仮想プラント表示インタフェースについても異常状態監視機能を行った。この機能は、他エージェントが直接、監視を行っていない製品性状について監視、異常状態通知を行うもの

であり、想定外異常についてその有効性を確認することができた。

運転員にとって、プラント状態の直感的な理解を助けるため、また、全ての機器の異常に対応するために、プロセスデータを可視化表示した。プラントとして考える全てのプロセスデータに対して、ミドルウェアを意識して汎用的に表示できる可視化方法を考案した。

前年度までは、市販の VR ツールによって仮想環境構築を行っていたが、本年度は VR ビューアを開発することにより、可搬性のある仮想環境を構築できた。これにより、仮想プラント表示インタフェースのミドルウェア化が実現でき、所期の計画目標を十分達成することができた。

仮想プラント表示インタフェースによるプラントヒューマンインタフェースの改良は、ユーザ評価により、一定の水準に達していることが確認された。

1.3.3.分散協調処理技術

平成 11 年度に開発した分散協調処理機構の機能・性能について、融合システム上で評価を行い、改良項目を検討した。具体的には、・協調処理機構の速度性能の評価・サブシステムから見た不足機能の洗い出しと追加項目の検討・(上記に関連して、特に必要と思われた)サブシステム間同期メカニズムの検討などを行った。その結果、特に 1 回の協調処理依頼に対して平均 12 個程度のスレッドを新規に起動する点が、実用化に際して応答速度やメモリ性能上致命的な問題である点を重視し、アーキテクチャの見直しを行うことを決定した。

上記の検討結果に基づいて分散協調処理機構の改良を実施した。また、改良範囲が広いため、作業効率面から分散オブジェクト機能に既存技術を部分適用することを決め、Java の強力な分散オブジェクト管理機能である Jini に協調処理機能を実装した。以上の改良版の分散協調処理機構については、性能評価用のベンチマーク環境を作成し、評価を実施した。評価の結果、機能面、性能面で当初の目標に達していることを確認できた。また、サブシステム数が更に増大した場合の応答速度改善、各サブシステム固有の通信アドレスやインタフェース仕様、タスク情報などの設定・動作確認ツールの必要性、などの課題も明確になった。

1.3.4.プラントモデルとインタフェースシステム

①最終評価

ユーザーの立場から、本システムが実用化に向けて有効であるか、評価基準について、大阪大学、岡山大学、石川島播磨重工業との調整や他の類似研究の調査などを通じて、検討を行なった。最終評価を行なうため判断基準となる項目を抽出し、アンケートを作成した。作成したアンケ

ートを、石連を通じて石油他社に対して行なった。今後当社の技術評価と合わせ総合的な評価となる様結果について解析を行なった。

②シナリオの充実

昨年度まで用意した9件のシナリオ内の因果関係・挙動をさらに詳細に分析し、各要素技術に運転ノウハウを提供した。さらに、評価用シナリオを5件に絞り、プラント情報を充実させた。評価用シナリオの内デモンストレーション用シナリオをさらに3件絞り込み精度向上のための運転ノウハウを提供した。シナリオ以外の想定外のトラブルについても検討し、システムの一般性の評価についてもあわせてを行なった。

③簡易プラントモデル構築のサポート

多様な異常に対応できる様、プラント挙動についての物理・化学計算式をベースとした経験モデル式をシナリオに応じて提供した。合わせてモデルの計算結果の検証、評価を行った。ユーザーとしてシステム使いやすさなど総合的に評価した。

④大画面設計

オペレータ向け大画面を開発することで、多くの情報を必要とする形で提供できるよう、実在のオペレータの行動様式についての情報と望むべく機能について情報の提供を行なった。システム結合テストにより、状況に応じたメッセージ、グラフ、警報が提供されるようになりユーザーの立場としてプラントヒューマンメディアとしての情報提供手法について評価を行なった。

⑤プラントオントロジー言語登録・出力機能評価

オントロジーが出力する言語や文章が実在のオペレータに理解し易いか、具体的なメッセージをベースに評価し、アドバイスをを行なった。システム結合テストにより、状況に応じたメッセージ提供されるようになり、言語や文章について評価するとともにシステムの使いやすさなどもユーザーの立場から評価した。

⑥運転用インタフェースエージェント用データの収集・評価

提供した異常シナリオをシミュレータにて検証し、実データの提供ならびにシナリオのブラッシュアップを行なった。

⑦VR用データの収集・評価

プラント情報の可視化など、仮想プラント表示に必要な機器データの提供とともに、実際の製油所における機器及び配管の設置状況に関する情報を提供した。

実運転の場でVRをより有効に活用するための視点についてユーザーの立場から総合的に評価する予定。また原油、製品の配管ルートの詳細

情報を提供した。

1.3.5. プラントオントロジー

① 協調プロトコルとオントロジーの洗練

計画当初は平成 11 年度に開発したタスクオントロジーを拡張しつつ、より高度な協調を可能にするための協調プロトコルを開発する予定であったが、研究が進むにつれて協調に関しては現状の能力にとどめ、余力をミドルウェア化の推進に当てることになったので、協調プロトコルは別のドメインでの予備的な開発を行った段階でペンディングとした。

② 融合化の更なる促進と有用性の検証

想定している全シナリオにおけるメッセージ生成、及びオントロジーサーバの動作確認は終了している。また、想定外の状況における動作は、メッセージが想定しているテンプレート内であれば対応できることも確認できている。しかし、全く新たな文体のメッセージを生成する必要が生じた場合には対処できないことは採用した方式に固有の限界であることは既に分かっている。

③ ミドルウェアとしての整備

プラントオントロジーとそれを用いたプラントモデル構築を支援するツールの開発は終了している。短時間のトレーニングで専門知識なしにツールを使うことは可能である。また、対象プラントを変更した際にメッセージ生成機能が修正なしにある程度追従することが確認できているので、十分な拡張性が確保されている。

1.3.6. 意味表示インタフェース

① 表示意図に基づく意味表示の実装

平成 11 年度検討した表示意図に基づく意味表示手法を機能一目標レイヤーの可変詳細度表示として実装した。すなわち、正常時には原油から各製品成分が蒸留される質量の流れ部分のみを詳細表示としその他は簡略表示とするが、異常発生時や異常原因推定時には関連部分を自動的に詳細表示するようにした。

② 高速シミュレータによる対応操作の選定手法の開発

対応操作の効果を検討するための高速簡易予測シミュレータを完成し、定性推論と定量推論を組み合わせた対応操作候補評価手法を開発した。簡易予測シミュレータの計算精度については、日石三菱殿の所有するダイナミックシミュレータによる計算結果と比較することにより検証した。

③Java 版プラントモデル構築ツールの整備

意味表示インタフェースにおける意味表示や対応操作候補導出手法のベースとなるプラントモデルの構築を支援するためのツールの Java 版を開発した。

④最終プロトタイプシステムの開発

大画面表示や追加シナリオに対応した最終プロトタイプシステムを開発し、分散協調環境下の実装した。

⑤異常原因推定手法の開発

ユーザサイドの強い要望により、異常原因推定手法を新たに追加開発し、最終プロトタイプシステムに組み込んだ。本手法では、プラント構成機器や系統に対する部分的な機能モデルに基づき故障モードを定義し、故障モードの発生可能性を判定するルールにより、異常状態が検知された場所から機器接続情報を利用して探索範囲を拡げながら異常原因の推定を行うことが可能となる。

1.3.7. 大画面統合表示技術

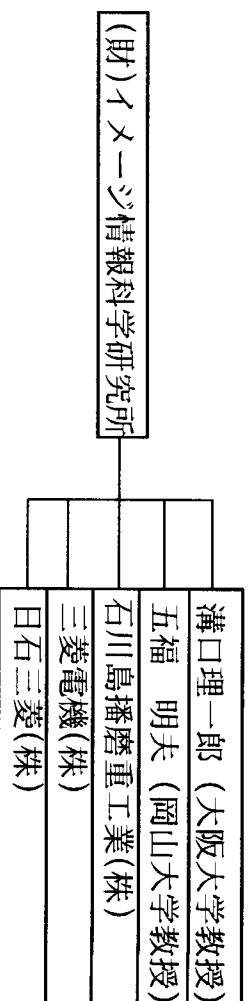
各種シナリオに対する大画面統合表示パターンを検討し、大画面統合表示に関する合意形成のアルゴリズムを導出すると共に、大画面統合表示のためのガイドラインを作成した。

1.4.体制

以下に平成12年度の研究開発体制を示す。

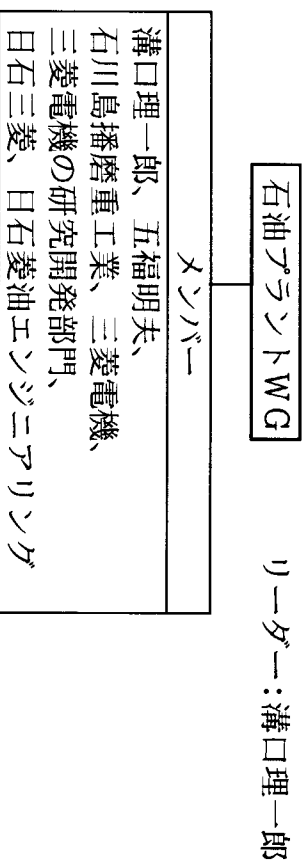
1. 研究体制

<研究開発体制>



<運用体制>

上記メンバー等を中心に以下のWGとその下に検討部会を編成し、研究開発の効率的な運用を行う。



2. 研究開発の実施場所

- 1) 石川島播磨重工業株式会社 メカトロ総合開発センター
東京都江東区豊洲3-1-15
- 2) 三菱電機株式会社 情報技術総合研究所
神奈川県鎌倉市大船5-1-1
- 3) 日石菱油エンジニアリング株式会社
神奈川県横浜市中区桜木町1-1-8 日石横浜ビル
- 4) 大阪大学産業科学研究所
大阪府茨木市美穂ヶ丘8-1
- 5) 岡山大学工学部
岡山市津島中3-1-1
- 6) 財団法人イメージ情報科学研究所 ヒューマンメディア業務部

東京都文京区本郷５－２６－４ 東京クリスタルビル９階

３．研究担当者氏名

<石川島播磨重工業（株）>（９名）

氏 名	所属・役職(職名)	研究項目
山内 進吾	プロジェクト計画部 次長	インタフェースエージェント、仮想プラント表示インタフェース
松浦 由次*1	制御システム開発部 部長	インタフェースエージェント、仮想プラント表示インタフェース
小出 誠二	制御システム開発部 部長代理	インタフェースエージェント
遠藤 政利	制御システム開発部 スタッフ	仮想プラント表示インタフェース
鈴木 俊太郎	制御システム開発部 スタッフ	インタフェースエージェント
御園 昇平	制御システム開発部 スタッフ	仮想プラント表示インタフェース
速水 努	制御システム開発部 スタッフ	インタフェースエージェント
山本 嘉久*1	情報システム部技術グループ スタッフ	インタフェースエージェント
入江 徹*2	情報システム部技術グループ 課長	インタフェースエージェント

* １ 転籍のため 平成１２年７月１日より削除

* ２ 異動に伴い 平成１２年７月１日より追加

<三菱電機（株）>（６名）

氏 名	所属・役職（職名）	研究項目
三石 彰純	情報技術総合研究所 電子システム部 主席研究員	分散協調処理技術
古市 昌一	情報技術総合研究所 電子システム部 主席研究員	
尾崎 敦夫	情報技術総合研究所 電子システム部 主任研究員	
二俣 隆	情報技術総合研究所 電子システム部 主任研究員	
渡部 修介	情報技術総合研究所 電子システム部 研究員	
徳本 修一	情報技術総合研究所 電子システム部 研究員	

<日石三菱(株)> (8名)

氏 名	所属・役職(職名)	研究項目
徳永 幹雄	技術部 工務技術グループ 参事	プラントモデルとインタフェースシステム
吉成 正博	技術部 工務技術グループ 参事	
福地 正	技術部 工務技術グループ 参事	
井上 久一	技術部 工務技術グループ 主事	
小川 芳尊	技術部 工務技術グループ 主事	
梅田 浩和	技術部 工務技術グループ 主事	
牧野 浩	技術部 工務技術グループ 主任	
渡邊 康太郎	技術部 工務技術グループ 主任	

<大阪大学> (4名)

氏 名	所属・役職(職名)	研究項目
溝口理一郎	産業科学研究所知識システム研究分野担当 教授	プラントオートロジー
池田 満	産業科学研究所知識システム研究分野 助教授	
来村 徳信	産業科学研究所知識システム研究分野 助手	
瀬田 和久	産業科学研究所知識システム研究分野 助手	

<岡山大学> (2名)

氏 名	所属・役職(職名)	研究項目
五福 明夫	工学部システム工学科 教授	意味表示インタフェース
小山 真二郎	工学部システム工学科 技官	

<(財)イメージ情報科学研究所> (3名)

氏 名	所属・役職(職名)	研究項目
安原 宏	ヒューマンメディア業務部 技術調査役	大画面統合表示技術
松本 美浩	ヒューマンメディア業務部 部長	
深山みゆき	ヒューマンメディア業務部	

1.5. Summary of FY2000 Human Media Technology R&D:Advanced Information System Technology R&D for Oil Refinery Plant

1.5.1. Overview of the R&D

Human Media project is a METI (Ministry of Economy, Trade and Industry; the successor to MITI:Ministry of International Trade and Industry)-funded national project started in FY 1996. This is an overview of a project on “Development of a human interface for the next generation plant operation” running as a subproject of the Human Media Project. It is METI’s intention that research on concrete topics to realize the idea of Human Media technology should be conducted to make the results visible and easily understandable rather than basic research about Human Media in general. The FY2000 is the end of the first project term.

The philosophy underlying the project is summarized as the coined term “Human Media” which stands for the advanced user-friendly media technology using three representative media such as Knowledge (AI) media, Virtual media and “Kansei” media. “Kansei” is a Japanese term whose rough meaning is the sixth or seventh sense for satisfaction, comfort, beauty, softness, etc. Technologically, the heart of Human Media is an integration of the above three kinds of advanced information technology to come up with an innovative human-centered interface.

We see four major trends of paradigm shifts in computer technology:

- (1) From Computer-centered to Human-centered
- (2) From Processing-centered to Information-centered
- (3) From Form-oriented to Content-oriented
- (4) From Centralized control to Distributed control

The first two are based on deep reflection on the long history of computer-centered and processing-centered research which have never been good in human-computer interaction aspects, since such technology forces a human to approach machines/systems or allow, at best, addition of an ad-hoc interface on top of each system. We need to change paradigmatically to come up with an innovative and essentially better human interface technology. The concepts of Human-centered and Information-centered technologies are key concepts of such an enterprise. The true man-machine systems, which are what we need in the coming information age, do require an open architecture involving humans who need computers to help them facilitate their daily activities. The heart of Human Media project is compliant with

these paradigm shifts, more strongly, it is an enterprise to promote such shifts by providing concrete technological ideas through the integration of Knowledge media, Virtual media and Kansei media.

The third is a bit different from the first two in that it is related to artificial intelligence (AI) research where so-called form-oriented basic research has been extensively conducted. It has been trivial from the beginning that no intelligent system can function without a reasonable amount of knowledge. Nevertheless, form-oriented research has dominated AI research. Content-related activities are mainly knowledge base construction. Although huge amount of such activities have been conducted to date, they cannot be said to be "Research", since they are ad hoc, heavily domain-specific and hardly accumulatable. We do need content-oriented "Research" to make an essential contribution to intelligent system building. This is related to Knowledge media technology.

The fourth is related to system architecture issues. It is an infrastructure of building large-scale robust systems, which are often difficult to build and maintain. Typical distributed control systems include a multi-agent system in which agents collaborate with each other without a priori specification of interaction between them unlike the conventional centralized control systems. This paradigm of system design makes it easier to build a large-scale system provided a powerful negotiation protocol.

On the basis of the background described above, we decided to employ the following four major policies to conduct the project:

(a) Open-architecture including human operators rather than closed one where a system tries to solve everything by itself:

This policy is the most straightforward decision made according to the project mission of building a sophisticated human interface for plant operators. The human operators who are responsible for every decision should take the initiative. The role of systems in support of the plant operation has to be not "passive" but "active". By active support, we mean information is given to the operators not only by their demands but also system's spontaneous decision on what information is necessary and when it should be provided. Such support systems should have potentials of coping with the difficulties and such capabilities enables them to work with human operators who require an intelligent partner which could augment their capabilities. Such systems apparently have to provide human operators with

useful and timely information about the plant including suspected causes with operation recommendations with justifications. For that purpose, we employ an Interface Agent who monitors the dynamic states of the plant and notifies operators of noteworthy events as well as conventional messages to operators. It complies with the paradigm shifts (1) and (2).

(b) Introduction of ontological engineering to avoid ad hoc knowledge representation and utilization: Conventional AI research has concentrated on formal issues such as logic, knowledge representation, formal reasoning, etc. However, the real world problems require sophisticated treatment of knowledge, that is, to cope with the issues of how to make the knowledge base construction principled and how to make the knowledge bases reusable. Ontological engineering is expected to give a solution to this. It complies with the paradigm shift (3).

(c) Multi-agent architecture rather than centralized one: Such a system that we plan to build will necessarily become a large-scale system, which is hard to develop, control and maintain. The multi-agent architecture with distributed control has many merits provided we come up with powerful collaboration protocol and vocabulary or concepts which are two of the major research topics of the project. It complies with (4)

(d) Case-based and model-based approach rather than heuristics-based one: Unlike the conventional expert systems, we decided to employ Case-Based Reasoning: CBR for monitoring the plant and Model-Based Reasoning: MBR for understanding the plant state with prediction of plant behavior. These policies make the system unique and different from conventional systems of similar purposes. It is not an expert system in conventional sense. The next section discusses motivation and background of the project together with project organization.

1.5.2. The background of the art of the plant operator support

We adopted an oil refinery plant as a task domain. After general investigation and several interviews with domain experts and plant engineers, the following current situations of plant operation were revealed:

(a) Operators have little chances to acquire know-how of plant operation because of the increase of reliability of the plant.

(b) A plant has become a large black box to operators because of its growth in size and advancement of automation.

(c) The human interface available is at best a multiwindow system of DCS: Distributed Control Systems, which is totally a passive information provider with an uncontrolled alarm system.

(d) The way of information provision is monotone rather than adaptive to operators and situations.

(e) Many kinds of information are available upon request, but all given is unrelated and disordered. Lack of integration.

(f) All the operator support systems are built from scratch and hence expensive.

(g) Maintenance of the support systems has become more and more difficult according to the growth of the plant in scale.

The goal of our project is apparently to resolve these difficulties. A plant should be transparent to the operators. The interface cannot be a simple thing which exists between the system and operators, rather, it should be a mediator which assists operators in making appropriate decisions by giving necessary and sufficient information at appropriate time based on a deep understanding of the plant itself and characteristics of human perception and cognition. Information provision should be not passive but active. Information should be integrated in the sense that all pieces of information should be controlled by a principled manner according to an explicit goal of the provision. In order to enable the interface to understand the plant, it should have a model of the plant and hence it should employ model-based reasoning. Such an interface necessarily becomes a huge system. But, it should not be difficult to build. The knowledge bases should be reusable for building similar systems later on. It requires an ontology engineering to build reusable knowledge bases. Such a system should be flexible in the cases of building and execution, which suggests multi-agent architecture with negotiation functionality.

(1) Interface Agent: IA

It is located between the human operators and the plant to monitor the plant states for them, triggers the collaborative actions among other agents and gives messages to operators in both text and speech. It is an “agent” in the following two meanings: An agent who performs monitoring job for human operators and an agent who can autonomously work with other agents. It consists of two major modules: case-based reasoning for plant monitoring and spoken language processing for human interface. The former stores cases of time series of various sensor data of the plant together with labels (normal, abnormal, etc.), diagnostic information and operations done for the situation. All the information is indexed for retrieval and later-use for plant monitoring, diagnosis and operation. Techniques used in the spoken language processing include Direct Memory Access Parser: DMAP and Memory Organization Packet: MOP. Speech input and output is carried out using commercially available speech boards. It employs no heuristics of the operators to make it free from the difficulties that conventional expert systems had. It can learn through its experience by increasing the case base.

(2) Virtual Plant Agent: VPA

VP is responsible for realizing a 3-D virtual plant for easy access to any portion of the plant. It allows operators to inspect any sensor data for his decision making as well as virtual operation on devices. It makes the plant physically transparent by visualization. The virtual plant has been carefully designed to imitate the real plant in a good fidelity so that plant operators can enjoy highly situated cognitive activities.

(3) Semantic Information Presentation Agent: SIA

The major functionality of SIA includes providing semantic information of the plant based on multi-layered logical model of plant rather than raw data. Given candidates of causes of a fault from IA, it identifies the most plausible one with restoration operation for it. IA and SIA are complementary to each other in that SIA reasons about the plant using model-based methodology, while IA employs a CBR approach based only on the past experiences. When IA finds a very similar case, the decisions drawn from it might be highly reliable. On the other hand, if it cannot, CBR cannot say anything useful. Model-based approach is generally robust as far as the plant states exist

within the coverage of the model built. Thus, the two compensate for each other.

(4) Ontology server: OS

Each agent has their own specialty, which differs a lot from each other. It is a must for them to share minimum commitment on understanding of the domain to collaborate with each other. A solution to this is ontology. Ontology is a shared theory/system of concepts of the target world to specify its conceptualization. The concepts and vocabulary in ontology are used as building blocks of the operational model of the world. The ontology server, OS stores an ontology developed for the target oil-refinery plant and serves it with agents. All agents necessarily commit to the shared understanding by using the ontology and hence they can maintain fluent communication. Further, OS has a function of message generation in terms of appropriate words. OS is always asked to generate a message by IA who wants to say something to the operators. It sends OS stylized meaning representation, which should be converted into natural language, Japanese in our case. The word selection is done consulting the model of the target plant and definitions of terms in the ontology.

(5) Distributed Collaboration Infrastructure: DCI

The infrastructure performs the functions of a communication channel and a negotiation facilitator for the communication among agents. Needless to say, DC1 and OS are compliant with the above policies. They make the whole system robust in runtime and easy to maintain thanks to its sophisticated negotiation protocol. An agent only has to know what jobs they want to get done and send a message using the local terms which have been told OS in advance without knowing who could do the job. The unique point of DC1 is that it introduces Proxy agents (in our system they are called Task agents) which do the negotiation among them for the agents they represent. Therefore, addition and modification of any agent is relatively easy.

(6) Plant Simulator

The system requires real time plant operation data. At an actual plant, DCS (Distributed Control System) is installed and manipulates the plant data every second. In this project, a real time dynamic simulator instead of a real plant is installed, using the mathematical model representing a Topper (Crude Distillation Unit). This model consists of the following major equipment and instrumentation.

Desalter Preheat Train (19 Heat Exchangers)

Preflush Drum

Crude Furnace

Crude Tower (42 Trays)

3 Side Strippers

37 PID Controllers

18 Pumps

132 Analog Indicators and Monitors

93 Field Valves

A set of tuning parameters such as PID and alarm range can be changed arbitrarily. This model can produce emergency operations such as pump degradation and failure, tube leakage, valve stuck and controller failure etc. After the set values of controller are changed or malfunctions are set, the operating condition will be changed. This model can calculate the changing data in every second and send the data to Interface Agent in every 2 seconds.

1.5.4. Contents and Results of the FY2000 R&D

In FY2000, the prototype system of FY1999 was evaluated and updated. Further in order to use for the general purpose, middleware researches were approached by each sub-system.

① Interface Agent

(R&D Contents)

In order to evaluate from practical point view,

To evaluate plant monitoring functions through multiple scenarios

To evaluate voice dialogue system

In order to approach to the plant operation middleware,

To develop case base editor, end user GUI of Case Base Reasoning System using XML.

(Results)

Plant monitoring function was updated from 2 anomaly scenarios to 7 anomaly scenarios.

Voice dialogue system was improved by C++.

The function of acquisition, store, retrieve, supply and monitoring of plant data was provided to DCS agent.

WEB browsing and GUI editing of case base contents were developed.

② Virtual Plant Agent

(R&D Contents)

To evaluate and improve VP

To upgrade VP autonomy function

In order to approach to the plant operation middleware,

To develop general purpose VR application focused on human interface

To improve visualization of process data

(Results)

Products state monitoring function and its communication function were developed.

Process data was visualized.

VR viewer program was developed.

VP was evaluated from user viewpoint.

Operational level was gained.

③ Semantic Information Presentation Agent:

(R&D Contents)

To implement diagnostic information based on display intention with variable levels of detail

To develop fast and simple forecast simulation combined with qualitative and quantum reasons

To re-write modeling tool by Java

(Results)

Variable LOD was implemented by fluid of mass in detail at normal and related part in detail at anomaly happening or identification of anomaly cause.

Evaluation method of counter actions was developed using simple forecast simulator.

Plant modeling tool was developed by Java.

New identification method of anomaly cause from strong user requirement was developed based on localized functional model of plant component and system.

④ Ontology Server

(R&D Contents)

To refine operation task ontology

To evaluate effectiveness of ontology server especially out of supposed scenarios

In order to approach to the plant operation middleware,

To consolidate ontology development environment

(Results)

Protocol development for advanced cooperation was tried in different domain. But main focus was shifted to middleware.

In case of out of supposed scenarios, message was successfully generated if message would be applied in template, but the message was not applied if it was quite different sentence pattern.

Plant ontology development tool was developed., of which short time training and message generation facility were applied even if domain was changed.

⑤ Distributed Collaboration Infrastructure

(R&D Contents)

To evaluate and improve distributed collaboration infrastructure, in particular, processing speed, autonomy of collaboration mechanism and synchronizing mechanism

(Results)

12 occurrences of threads at 1 collaboration request were found and architecture was improved by using Jini.

The goal was attained by the improvement but the further increase of subsystems would bring another issues.

⑥ Plant Simulator

(R&D Contents)

To evaluate total system by the following view points,

Easiness of plant system operation and adequacy of display timing from operator point of view

Maintenance of subsystems in case of system upgrade

Applicability to other operation work

(Results)

User questionnaire was made and the answers were analyzed.

Scenarios were improved.

Simple plant model construction was supported.

Information for large-scale display was supplied.

Ontology terms and sentences were evaluated.

Scenario was improved by simulation.

VR data was supplied.

⑦ Large Scale Display

(R&D Contents)

To derive display integration algorithm

To make guideline for large-scale display

(Results)

Scenario based on display algorithm was implemented.

Cooperator model of large-scale display was presented.

2. 成果の詳細

2.1. 全体システム

前年度に開発したフェーズ2プロトタイプシステムでは、システムアーキテクチャとしてマルチエージェント方式を採用し、オペレータへ自動的に情報提示するシステムを構築した。そして、それらのエージェントが生成する情報を統合して、大画面に表示する大画面統合表示システムを開発したが、そのシステムは、予め用意しておいたJPGファイルを切替えて、表示するだけの機能だったため、決められたシナリオしか動作することができなかった。そこで、今年度では、実際に各サブシステムから転送されてくる情報を用いて、動的に画面生成を行う大画面統合表示システムを開発し、プロトタイプシステムへ統合した。フェーズ2プロトタイプシステムと今年度開発したフェーズ3プロトタイプシステムの違いを表 2.1-1に示す。

表 2.1-1フェーズ2プロトタイプシステムとフェーズ3プロトタイプシステムの違い

	フェーズ2プロトタイプシステム	フェーズ3プロトタイプシステム
大画面統合表示システム	JPGファイルの画面切替方式	エージェントから生成された情報を用いて画面表示する方式
インターフェースエージェント	音声入出力機能、事例ベースプラント監視機能、プラントシミュレータとのI/F機能を1つのサブシステムに実装	事例ベース監視エージェント、DCSエージェントを新規に作成し、インターフェースエージェントから事例ベースプラント監視機能、プラントシミュレータとのI/F機能を分離
意味表示エージェント	定性推論機能、定量情報表示機能を1つのサブシステムに実装	意味表示エージェント1、意味表示エージェント2を新規に作成し、定性推論機能と定量情報表示機能を分離
システムの実装レベル	2つのシナリオしか動作できない	限られたシナリオだけでなく、全ての異常に対して動作し、情報提示することができる
音声対話	音声対話機能は実装されていなかった	聞き返しにより、音声コマンドを補完していく、音声対話機能を組み込んだ

2.1.1.Co-operator モデルとプラント問題解決プロセス

大画面統合表示システムを開発するにあたり、基本コンセプトとして、Co-operator モデルを提案した。Co-operator の役割は、Operator に代わって、センサ等の計測値を監視し、異常に気づくと Operator に教えてあげたり、Operator に言われたタスクを即座に実行し、Operator に伝えることである。また、異常等の様々な状況において、その時々に応じた有用な情報を、Operator のタスクを邪魔することなく、伝えることが、有能な Co-operator の役割ともいえる。Operator と Co-operator との関係を図 2.1-1に示す。Operator と

Co-operatorの間には、複雑な情報を伝達し、協調しながら問題を解決するというCPS(Collaborative Problem Solving)場が存在する。このCPS場では、Operatorの意思決定を支援するために、円滑な情報伝達が重要となる。また、最終的な判断はOperatorが行う。Co-operatorが最終的な判断まで下してしまうと、権威勾配が逆転し、本来の役割分担が失われて、大きな事故を起こす可能性があるからである。

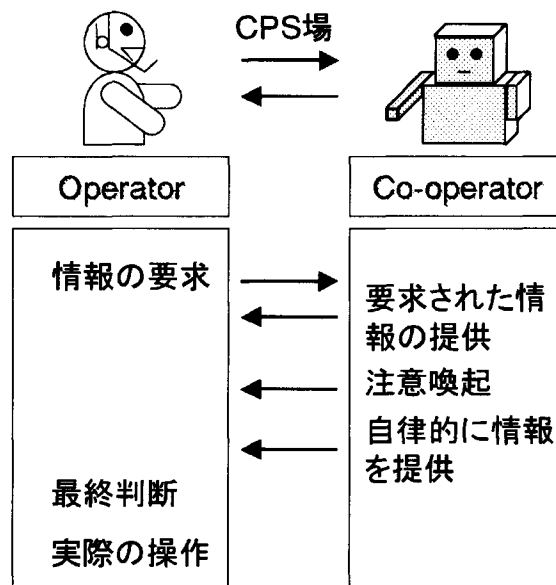


図 2.1-1 Operator-Co-operator モデル

Operator と Co-operator 間の権威勾配が逆転することなく、円滑に情報伝達し、協調しながらタスクを実行するためには、次のインタフェースが必要である。

- オペレータの問題解決プロセスを支援するインタフェース
- オペレータのメンタルモデルを考慮したインタフェース
- オペレータに不安を起こさせないインタフェース
- ヒューマンエラーを考慮したインタフェース
- 石油プラントの環境条件を考慮したインタフェース

ここで、問題解決プロセスとは、「異常が発生する」→「異常に気づく」→「現在状態を把握する」→「異常回避のための対処法を考える」→「具体的な操作を考える」→「対応操作を実施する」→「定常状態に戻る」という異常が発生した時に運転員が踏むステップを示す。プラントの場合の問題

解決プロセスを図 2.1-2に示す。異常が発生すると異常通知をして、システムに注目してもらう。次に現在のプラント状態と近未来予測したプラント状態を提示することにより、異常箇所と異常レベルを運転員に把握してもらう。さらに、過去の異常原因、過去の対応操作、異常原因推定結果、対応操作導出結果を提示することにより、異常回避法を考える際の支援情報を提供する。そして、運転員が具体的な操作を決定したところで、操作手順を提示し、それに従った操作を実施する。

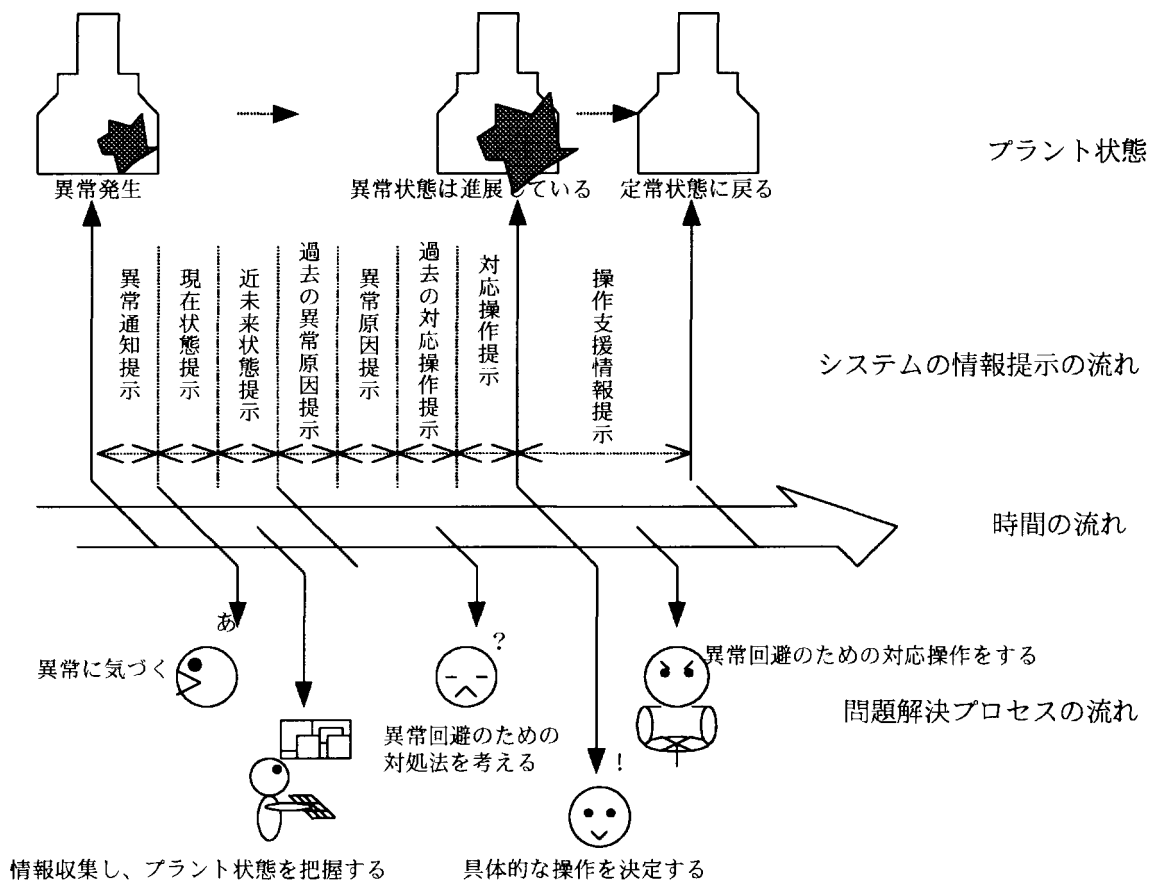


図 2.1-2 プラント問題解決プロセス

2.1.2. システム構成

大画面統合表示システムを組み込んだフェーズ3プロトタイプシステムの構成を図 2.1-3に示す。フェーズ2プロトタイプシステムでは、インタフェースエージェントが、オペレータとのインタフェース機能、事例ベースプラント監視機能、プラントシミュレータとのインタフェース機能を持っていたが、フェーズ3では、事例ベースプラント監視機能、プラントシミュレータとのインタフェース機能を分離し、事例ベース監視エージェントおよびDCSエージェントを新規に開発した。また、フェーズ2の意味表示エージェントでは、異常原因推定や対応操作導出といった定性推論機能と、グラフ表示や近未来予測といった定量情報表示機能が1つのサブシステムに融合されていたが、定性推論機能と定量情報表示機能を分離し、意味表示エージェント1および意味表示エージェント2を新規に開発した。これにより、各サブシステムは、次の4つのカテゴリに分類され、全部で10サブシステムにより構成される。

- ①人とシステムとのパイプ役をするエージェント
 - ・インタフェースエージェント (I A 1)
 - ・大画面統合表示システム (LD)
- ②自律的に情報生成するエージェント
 - ・事例ベース監視エージェント (I A 2)
 - ・意味表示エージェント1 (S I A 1) —— 定性推論処理部
 - ・意味表示エージェント2 (S I A 2) —— 定量情報表示部
 - ・仮想プラント表示エージェント (V P A)
- ③システムとプラントとのパイプ役をするエージェント
 - ・DCSエージェント (I A 3)
- ④インフラストラクチャ
 - ・オントロロジーサーバ (OS)
 - ・分散協調通信サーバ (DC)

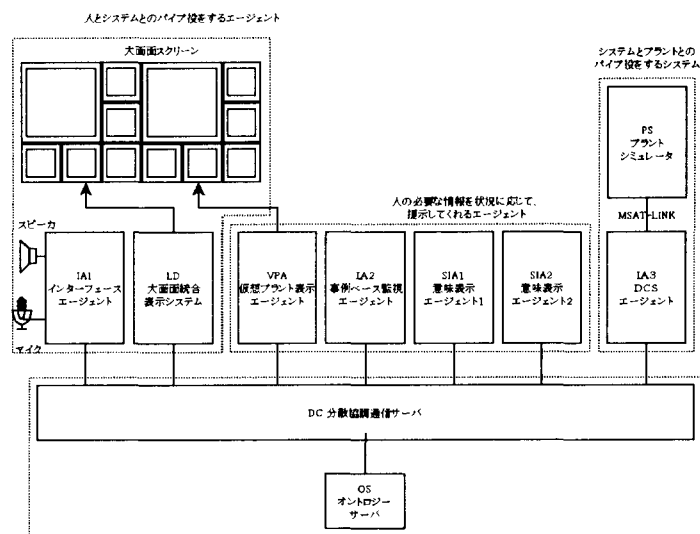


図 2.1-3 システム構成

2.1.3. サブシステムの機能

フェーズ3プロトタイプシステムの各サブシステムで実装されている機能を表 2.1-2に示す。

表 2.1-2サブシステム機能

番号	サブシステム	機能
1	インターフェースエージェント(IA1)	音声認識機能
		音声合成出力機能
		音声対話機能
		情報提示タイミング制御機能
2	事例ベース監視エージェント(IA2)	事例に基づくプラント状態監視
		セットポイント操作の事例登録機能
		異常通知機能
		過去の異常原因検索&表示機能
		過去の対応操作検索&表示機能
3	DCSエージェント(IA3)	プラントデータ提供機能
		セットポイント変更機能
		プラントデータ監視機能
		異常通知機能
4	意味表示エージェント1(SIA1)	異常機器表示機能
		異常原因推定&表示機能
		対応操作導出&表示機能
5	意味表示エージェント2(SIA2)	トレンドグラフ表示機能
		近未来予測機能
		近未来グラフ表示機能
6	仮想プラント表示エージェント(VPA)	異常機器VR表示機能
		プロセスデータ可視化表示機能
		機器操作機能
		操作手順検索&表示機能
7	大画面統合表示システム(LD)	各種情報(テキスト&グラフィック)表示機能
8	オントロジーサーバ(OS)	ウィンドウタイリング制御機能
		プラントモデル情報検索機能
		日本語メッセージ作成機能

フェーズ3プロトタイプシステムの大画面表示において、新規に開発した主な機能は、インタフェースエージェントの情報提示タイミング制御と大画面統合表示システムのウィンドウタイリング制御である。各エージェントは提示すべき情報が生成された時点で情報提示をするが、各エージェントが勝手に情報を表示してしまうと、短時間にしかもばらばらのタイミングで情報提示されることになり、運転員は混乱してしまう。また、大画面といえども、表示リソースには、限界があり、全ての情報を一画面に配置することができない問題がある。そこで、各エージェントから生成された情報はいったんインタフェースエージェントにバッファリングされ、必要なタイミングで、運転員へ情報提供す

る情報提示タイミング制御を考案した。さらに、重要な情報はより大きく表示し、時間の経過した情報については小さく表示するというダイナミックタイリング制御についても考案した。情報提示タイミング制御およびダイナミックタイリング制御については、2.1.5項で説明する。

2.1.4. 異常時における自律処理

プラントの監視は、事例ベース監視エージェント、DCSエージェントが行うが、異常を検知すると、全サブシステムへ Broadcast して、異常の通知をする。各エージェントは異常の通知を受信すると、現在状態表示、近未来状態表示、異常原因表示、対応操作表示を自律的に実行する。以下では、事例ベース監視エージェント、意味表示エージェント 1、意味表示エージェント 2、仮想プラント表示エージェントが異常時において自律表示する処理内容について記す。

(1) 事例ベース監視エージェントの自律処理

異常が通知されると、事例ベース監視エージェントは、図 2.1-4の処理を実行する。

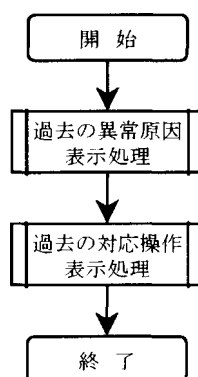


図 2.1-4 事例ベース監視エージェントにおける自律処理

事例ベース監視エージェントでは、過去の異常原因や過去の対応操作が事例として登録されており、異常発生機器および異常原因機器から、それらに関連する過去の異常原因や過去の対応操作を検索し、表示する。過去の異常原因検索では、異常通知で送信されてくる異常発生状況テンプレートをキーとして、過去に何が原因で、同じ異常が発生したかを表示する。また、過去の対応操作検索では、意味表示エージェント 1 で推定された異常原因をキーとして、セットポイントをどのように変更したかを表示する。これらを表示することにより、運転員が異常回避するための意思決定を支援することができる。

(2) 意味表示エージェント 1 (定性推論処理部) の自律処理

異常が通知されると、意味表示エージェント 1 は図 2.1-5の処理を実行する。

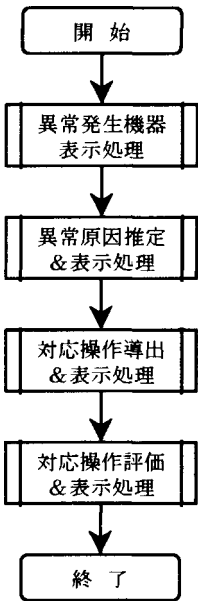


図 2.1-5 意味表示エージェント 1 における自律処理

意味表示エージェント 1 ではプラントモデルを持っており、異常通知を受信すると、プラント系統図に異常発生機器を黄色で表示するとともに、モデルベースによる異常原因推定処理、対応操作導出処理、対応操作評価処理を実行する。異常原因推定結果、対応操作導出結果、対応操作評価結果を表示することにより、運転員が異常回避するための意思決定を支援する。

(3) 意味表示エージェント 2 の自律処理

異常が通知されると、意味表示エージェント 2 は図 2.1-6の処理を実行する。

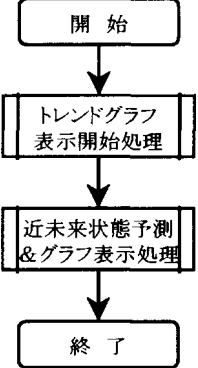


図 2.1-6 意味表示エージェント 2 における自律処理

意味表示エージェント 2 では、グラフ表示モジュールを持っており、異常通

知を受信すると、異常検知したセンサのトレンドグラフ表示および近未来トレンドグラフ表示を実行する。また、異常状態になるまでの時間も計算して、運転員へ通知する。これにより、異常状態を把握することができるとともに、異常の限界に達するまでの時間がわかり、早急に対応しなくてはいけないのかあるいは、まだ余裕があるのかを判断することができる。

(4) 仮想プラント表示エージェントの自律処理

異常が通知されると、仮想プラント表示エージェントは図 2.1-7の処理を実行する。

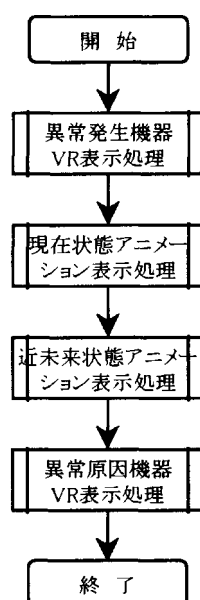


図 2.1-7 仮想プラント表示エージェントにおける自律処理

仮想プラント表示エージェントは異常通知を受信すると、VR画面上に異常発生機器を黄色で表示するとともに、異常発生機器の現在状態および近未来状態をアニメーションを用いて表示する。また、異常原因機器VR表示では、意味表示エージェント1で推定された異常原因機器を取得して、VR画面上に橙色で表示する。これらを表示することにより、異常発生箇所およびその状態を直感的に把握することができる。

2.1.5. 情報統合提示

異常が通知されると、各エージェントは、並列的に前節で記した処理を実行する。各エージェントは、情報が生成された時点で、インタフェースエージェントに情報提示を要求するが、各エージェントから要求がきたタイミングで情

報提示してしまうと、短時間にしかもばらばらにたくさんの情報が提示されてしまい、オペレータは混乱してしまう。そこで、インタフェースエージェントは、複数のエージェントから送信されてくる情報提示要求を統合整理し、運転員は現在どのような情報が必要であるかを考え、提示する情報の順番およびタイミングを制御する。情報提示する場合、音声出力はインタフェースエージェントが実行するが、大画面表示は大画面統合表示システムに要求するので、画面表示と音声出力を同期させる必要がある。

そして、大画面統合表示システムでは、インタフェースエージェントの表示要求に従って、大画面切替を実行するが、大画面といえども表示リソースには限界があり、全ての情報を表示することはできない。そこで、大画面統合表示システムは、表示する情報の中で、どの情報が重要であるかを考え、重要な情報を大きく、重要性の低い情報を小さく表示するように、各ウィンドウの位置・サイズを計算し、ダイナミックな画面切替を実行する。

以下では、インタフェースエージェントが実行する情報提示タイミング制御と大画面統合表示システムが実行するダイナミックウィンドウタイリング制御について記す。

(1) 情報提示タイミング制御

インタフェースエージェントにおける情報提示タイミング制御処理は、複数のエージェントから送信されてくる情報提示要求を受信する通信部と、それらの情報提示要求を一時的に保存しておく提示データキューイング処理部と、現在キューイングされている情報提示データの中からどの情報を提示するかを決める情報提示要求選択処理部、画面表示と音声出力を同期して提示する情報提示タイミング管理部から構成される。情報提示タイミング処理の機能ブロック図を図 2.1-8に示す。また、提示情報選択処理部の処理フローを図 2.1-9に示す。

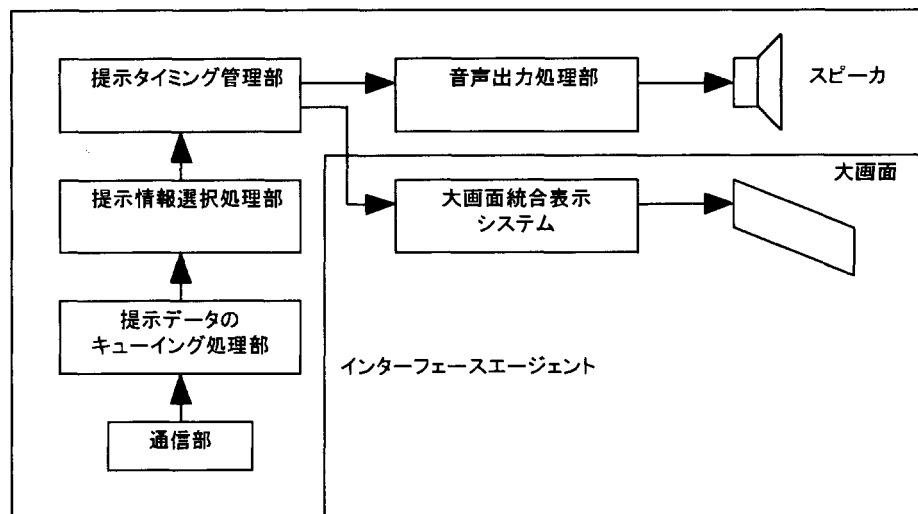


図 2.1-8 情報提示タイミング処理のブロック図

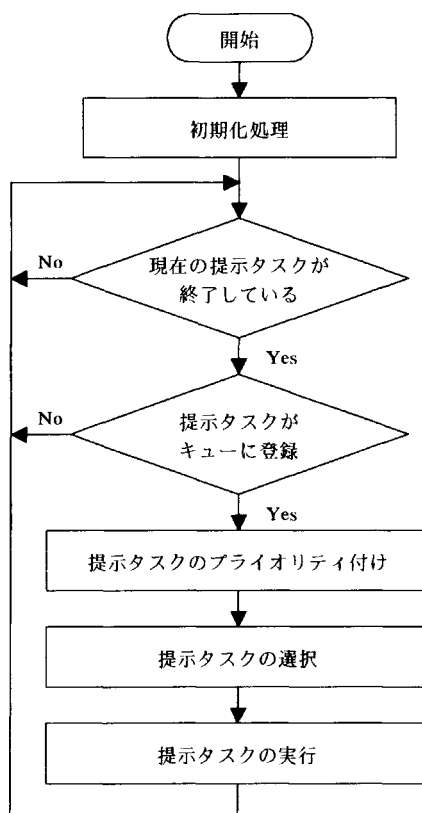


図 2.1-9 情報提示タイミング処理アルゴリズム

情報提示する順番は、2.1.1 項で説明したプラント問題解決プロセスにそって行う。これにより、運転員の現在置かれた状況に従った適切な情報をタイムリーに提供することができる。

(2) ダイナミックウィンドウタイリング制御

大画面統合表示システムにおけるウィンドウタイリング制御処理は、表示要求を受信する通信部と、表示すべきウィンドウの位置・サイズを計算するウィンドウタイリング計算処理部と、計算した結果が表示リソースに収まるかどうかを判定する表示可能判定処理部と、計算結果に従ってウィンドウを表示するウィンドウ表示処理部と、オペレータのマウス操作を管理するポインティングデバイス操作管理部から構成される。ダイナミックウィンドウタイリング処理の機能ブロック図を図 2.1-10に、処理フローを図 2.1-11に示す。

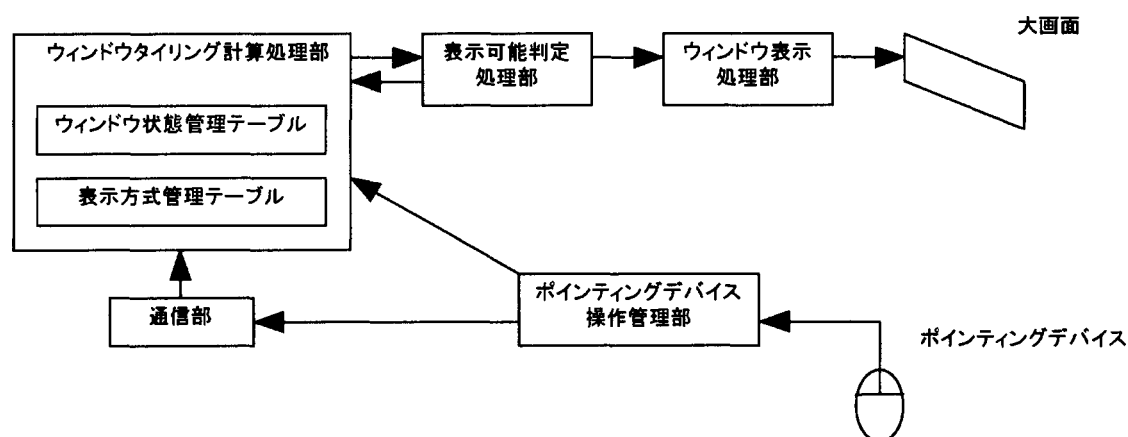


図 2.1-10 ウィンドウタイリング処理のブロック図

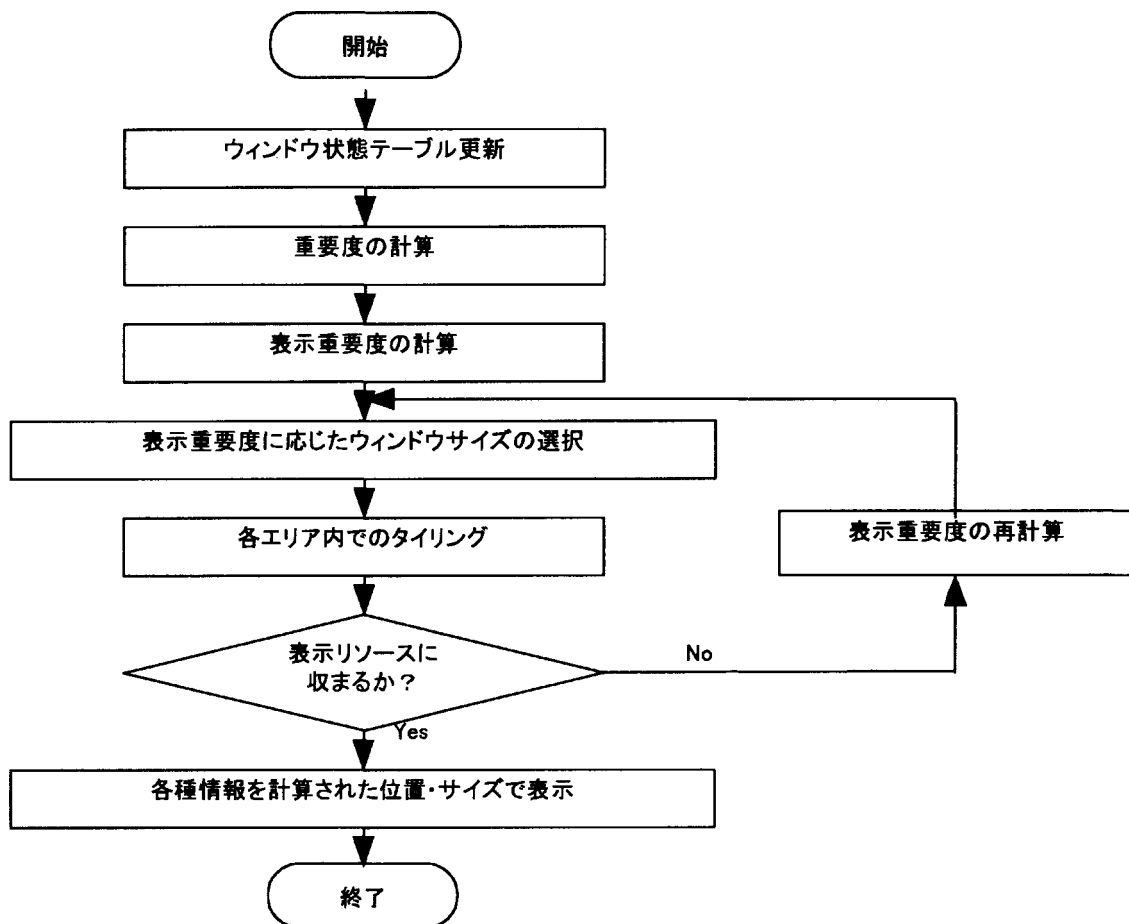


図 2.1-11 ウィンドウタイリング処理アルゴリズム

大画面統合表示システムにおける情報ウィンドウタイリング処理は、インタフェースエージェントからの画面表示要求に従い、ウィンドウ状態管理テーブルの更新をするステップと、ウィンドウ状態管理テーブルと表示方式管理テーブルから各ウィンドウの位置およびサイズを計算するステップと、表示リソース内にウィンドウが収まるかどうかを判断するステップと、算出された位置およびサイズで実際に画面表示をするステップで構成される。大画面の表示リソースは、図 2.1-12に示すように、「プラント状態表示エリア」「意思決定支援情報表示エリア」「操作エリア」の3つエリアで構成される。プラント状態表示エリアには、異常通知提示情報、現在状態提示情報、近未来状態提示情報を表示する。意思決定支援情報表示エリアには、過去の異常原因、過去の対応操作、異常原因推定結果、対応操作導出結果を表示する。操作エリアには、操作手順情報、機器操作パネルを表示する。大画面統合表示システムにおけるタイリング処理では、問題解決プロセスに従って、各エリアのサイズがダイナミックに変化する。すなわち、最初は、プラント状態表示エリアが大きく表示され、異

常の発生した箇所とその状態を運転員に見てもらう。次に、過去の異常原因、過去の対応操作、異常原因推定結果、対応操作導出結果が表示されると、意思決定支援情報表示エリアが大きく表示され、対応操作を考えるための支援をする。運転員が対応操作を決定すると、操作エリアが大きく表示され、直接操作をするための支援をする。

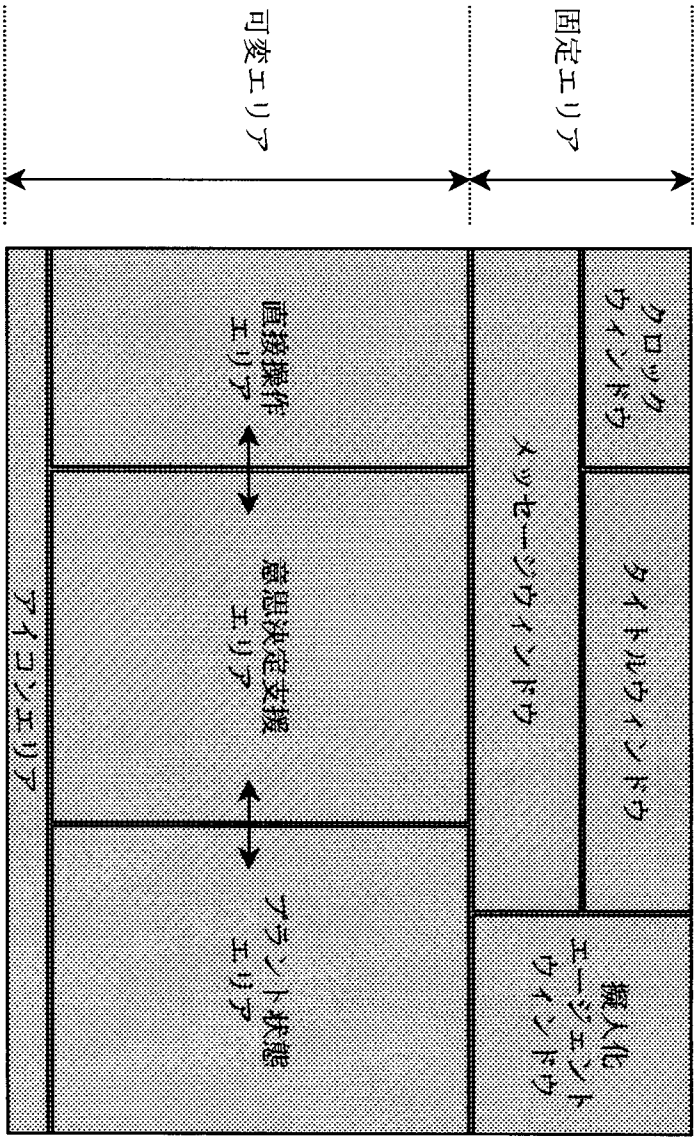


図 2.1-12 大画面統合表示システムの画面構成

インタフェースエージェントから、情報表示要求が送信されてくると、大画面統合表示システムでは、現在の状態と要求された情報を統合した表示を行うため、同じ情報表示要求であっても、以前の状態に応じて、表示方法も変わってくる。その状況毎に応じた表示をするために、大画面統合表示システムでは、各ウインドウの状態テーブルを持つ。初期状態は、プラント系統図のみ表示されている状態であるが、インタフェースエージェントから送信されてくる情報の順番に応じて、状態テーブルをダイナミックに更新する。ウインドウ状態テーブルから各ウインドウの重要度を計算する。この重要度は、各ウインドウ毎に別な値が設定されるが、インタフェースエージェントから情報表示が送信される時に、1回だけ計算される。次に表示重要度を計算するが、最初は重要度の値をそのまま設定しておく。そして、表示重要度に応じて、ウインドウサイズを決定する。ウインドウサイズが決定したところで、ウインドウの配置処理を

実行する。配置処理は、「プラント状態表示エリア」「意思決定支援情報表示エリア」「操作エリア」の各エリア毎に行う。各ウィンドウ毎にタイリングして、それらを統合した結果、1つの表示リソースに収まれば、計算された表示位置およびサイズで表示処理を実行する。ここで、表示リソースに収まらない場合は、表示重要度を1ランク下げて、タイリングし直す。そのため、表示リソース内に情報が表示しきれなくなるまで、表示重要度は更新されることになる。

ウィンドウ状態テーブルでは、各ウィンドウ毎に次のパラメータを持つ。

- ・優先度パラメータ P_{PRIORITY}
- ・新鮮度パラメータ P_{FRESH}
- ・調整用パラメータ P_{ADJUST}
- ・信頼度パラメータ P_{RELIANCE}
- ・緊急度パラメータ P_{URGENCY}

優先度パラメータは、そのウィンドウを表示する場合は1、表示しない場合は0を設定する。また、個人個人で、必要な情報も異なるため、各ウィンドウ毎に優先度を設定することにより、その人にとって重要なウィンドウは大きく、あまり重要でないウィンドウは小さく表示することができる。すなわち、重要な情報ウィンドウは1に近い数値を、あまり重要でない情報ウィンドウは0に近い数値を設定する。新鮮度パラメータは、情報の新しさを示し、0から1の値を設定する。最新の情報は、1を設定し、状況が変わる度（画面が切り替わる毎）に1ランク（0.2）デクリメントしていく。調整用パラメータは、問題解決プロセスどおりに情報提示している場合は、0を設定しておくが、問題解決プロセスの流れに反した順番で、ある情報を大きく表示してしまう場合に、±0.2の設定をする。信頼度パラメータは情報の信頼性を示し、0から1の値を設定する。同一種類の情報が同時に表示される場合があるが、信頼度の高いウィンドウの方をより大きく表示するために、このパラメータを設定しておく。緊急度パラメータは、特定のウィンドウのみを表示する必要がある場合に使用する。通常1を設定しておくが、表示しないウィンドウを0に設定することにより、1のウィンドウのみを表示することができる。

これらのパラメータを用いて、重要度 $P_{\text{IMPORTANCE}}$ および表示重要度 P_{DISPLAY} は次式によって計算される。

$$P_{\text{IMPORTANCE}} = P_{\text{PRIORITY}} \times (P_{\text{FRESH}} + P_{\text{ADJUST}}) \times P_{\text{RELIANCE}} \times P_{\text{URGENCY}}$$
$$P_{\text{DISPLAY}} = P_{\text{IMPORTANCE}} \cdot 1.0 \times N \text{ (再計算回数)}$$

ウィンドウ状態テーブルの例を図 2.1-13に示す。

	優先度 パラメータ	新鮮度 パラメータ	調整用 パラメータ	信頼度 パラメータ	緊急度 パラメータ	重要度 パラメータ	表示重要度 パラメータ	Window Size
	P _{PRIORITY}	P _{FRESH}	P _{ADJUST}	P _{RELIANCE}	P _{URGENCY}	P _{IMPORTANCE}	P _{DISPLAY}	
プラント系統図	1.0	0.4	0.0	1.0	1.0	0.4	0.4	Icon
トレンドグラフ	1.0	0.6	0.0	1.0	1.0	0.6	0.6	Small
近未来グラフ	1.0	0.8	0.0	1.0	1.0	0.8	0.8	Middle
過去の異常原因	1.0	1.0	0.0	0.9	1.0	0.9	0.9	Large
異常原因推定結果	1.0	1.0	0.0	1.0	1.0	1.0	1.0	Large
過去の対応操作	1.0	0.0	0.0	0.9	0.0	0.0	0.0	None
対応操作導出結果	1.0	0.0	0.0	1.0	0.0	0.0	0.0	None
:	:	:	:	:	:	:	:	:

図 2.1-13 ウィンドウ状態テーブル

2.1.6. 結果および考察

各エージェントが生成する情報を用いて画面表示する大画面統合表示システムを開発し、プロトタイプシステムへの融合化を実現した。

開発したシステムは、マルチエージェント方式を採用しており、自律的に情報提示を実行する。状況に応じて自動的に情報を提供してくれるため、システム操作に不慣れな人でも、必要な情報を適確に収集することができる。また、画面切替等の操作をすることなく、必要な情報が提示されるため、オペレータが自ら操作する手間が省け、オペレータの負荷を軽減することができる。すなわち、異常回避方法を考えることに専念できるので、より適切かつ迅速な異常回避方法を選択することができる。そして、より深刻な状況に陥る前に異常回避できるようになる。

システムはプラント問題解決プロセスにそった情報提示を行うため、オペレータは各ステップ毎に、より適確な状況判断、意思決定を円滑に行うことができる。

全ての情報を大画面に統合表示しようとする、非常に大きな画面を用意するか、モニターをたくさん用意する必要があるが、逆に、情報量が多いとどれを見れば良いかオペレータもわからなくなってしまう問題がある。ダイナミックウィンドウタイリング処理では、必要な情報を大きく表示することにより、どの情報が重要であるかをオペレータは即座に判断することができる。また、重要性の低い情報を小さく表示することにより、表示リソースを有効に活用す

ることができる。

昨年度までは、2つのシナリオにしか対応できなかったが、今年度は、プラントシミュレータで発生する全ての異常に対応できるシステムを構築した。意味表示エージェント1の異常原因推定および対応操作導出については、全ての異常に対応できていないが、モデルを追加すれば、対応することができる。

昨年度までのシステムは、全てのエージェントが揃わないと、動作できないという欠点があったが、今年度は、あるエージェントがいなくとも、そのエージェントが生成する情報が提示されないだけで、現在、起動しているエージェントのみでも動作するシステムを構築した。これにより、システム構成を柔軟に変更することができ、ユーザの必要な機能を持つエージェントのみで、システムを構成し、ユーザへ提供することができるようになった。

2.2. インタフェースエージェントの研究

2.2.1. 本年度研究の概要

2.2.1.1. 評価改良とミドルウェア化

本年度研究では、前年度の融合化研究を受け、実用性の観点からの開発システムの評価改良を行うとともに、成果の今後の普及を目的としてミドルウェア化を実施した。

2.2.1.2. 音声対話・擬人化エージェントとDCSエージェントの分離

前年度プロトタイプシステムでは、インタフェースエージェント、仮想プラント表示エージェント、意味表示インタフェースエージェントのそれぞれが自律的に情報を取得し情報提示する、マルチエージェント方式のシステムを構築した。インタフェースエージェントはプラント監視機能以外に、音声対話機能と各エージェントの要求に応じてプラントデータを提供する機能も、受け持っていた。本年度では、従来インタフェースエージェントが担っていた複数の機能を、実用性とミドルウェア化の観点から分離独立させ、事例に基づくプラント監視エージェント、音声対話・擬人化エージェント、プラントデータを提供するDCSエージェントを構築した。こうすることで、大画面表示における情報提示と音声出力のタイミング同期を可能にし、音声対話機能の先行的実用化を可能にし、プラントデータの蓄積・検索機能を強化することができた。

2.2.1.3. 各分離エージェントの研究概要

プラント監視エージェントでは、前年度二つの異常シナリオに対して対応したが、本年度は実用性向上の観点から、シミュレーション可能な五つすべての異常シナリオについて、正しく対応できるようにした。

音声対話・擬人化エージェントでは、音声対話部を、ObjectStore を用いない純粋な C++ 言語に書き換え、本年度開発のフェーズ3プロトタイプシステムに組み込むとともに、具体的な実用に供することを可能にした。擬人化については前年度に引き続き、女性とロボットの擬人化キャラクターを製作し、アニメ機能により音声出力に連動したいわゆる口パク機能を実現した。

DCSエージェントでは本格的なデータ蓄積検索機能実現を目的に、リレーショナルデータベースを導入し、時々刻々採取するプラントデータを継時的に蓄積するとともに、分散協調システム経由で他の各エージェントからの要求に対して、時刻・時間をキーに任意のプラントデータを検索提供できるようにした。また、従来DCSが行っている警報検出機能を高度化してDCSエージェントに実装し、警報出力するようにした。

2.2.2. 実用性向上

2.2.2.1. プラント監視、複数異常シナリオへの対応

実用性の観点から見て、プラント監視エージェントの最大の課題は実プラントにて生ずる様々な種類の異常パターンへの対応能力である。前年度ではシナリオ1「リフラックスドラム液面上昇」に加えて、シナリオ3「加熱炉出口温度(COT)低下」に対して、データ分解能を変更するだけで対応することができた。本年度はさらに、これまでに日石三菱より提供されてシミュレーション可能なすべてのシナリオ5個に対応することとした。

ここで考えた実現目標は、パラメータ等を何ら変更することなく、全部で5個のシナリオをどういう順番で行っても、システムは事例に基づく監視を実行できて、5個の異常を区別して適切な時期に検出することができ、しかも正常状態としたいところでは異常を検出しない、ということである。この目標を、最初監視に適切なデータを抽出し、次に高速化のためにバージョン空間の簡易化を行うことで実現した。

2.2.2.1.1. 監視データ抽出

プラント監視エージェントでは、プラント状態をその時々 of プラントデータの瞬時値および変化率で表現する。事例に基づくプラント監視が一般的に可能と考える根拠は、プラント変数で構成される多次元空間を考えたとき、類似のプラント状態はその多次元空間の中で互いに近傍に分布するであろうという前提である。本事例ベースシステムはあらかじめ構成しておいたバージョン空間のスケルトンに従って、この多次元空間内の点（プラント状態）を自動的にクラスタリングする。複数の異常状態を正しく識別するとか、正常／異常を混同しないようにするという前述の目標を実現するためには、空間を構成するプラント変数を正しく選択することがカギになる。監視用データの選択を次のような手順で行った。

- 最初一切の事前知識を用いず、各シナリオ毎にプラント正規化データの変動の大きいものから2-3個取り出し、6個を選択したが、専門家の立場からは不安定な変数を扱うことは好ましくないとのコメントがあった。
- そこで全シナリオの関連データ 19 個をまず取り出し、これに変動量の大きい3データを加えて監視対象候補とした。
- 21 個の監視データ中からまず6個を選択し、試行したが一部シナリオで正常／異常の区別ができないことがわかった。
- 監視結果を考察することにより、1個のデータを監視に加え、結局合計5個

のデータで目標を実現した。

監視データを多くすればするほど、多くの異常を分離識別できるようになることは自明である。しかし、監視データ数の増加は計算速度の低下をもたらす。したがって、本方式によるプラント監視を実現するためには、必要十分な精度で最小限の監視用データセットを発見する必要がある。今回の目標実現においては、シミュレータで実際に異常を起こして、ヒューリスティックな手法で監視用データセットを発見したが、実用化のためには監視用データセットを決めるためのもっと組織的な手法が必要と考える。

2.2.2.1.2. バージョン空間簡易化による高速化

上記監視データセットの発見には、あらかじめプラントシミュレータから採取したデータファイルを用いて、オフラインのリスブシステム (Pentium-II300MHz、AllegroLisp5.01)により行ったが、6データでは実行時間に何ら問題なかったものが、7データにしたとたんにイニシャライズ（事例の階層構造作成のためのスケルトンをあらかじめ作成する）に30分、一回のプラントデータを処理するために4分かかるようになった。本事例に基づく監視方法では、入力データについて瞬時値のみならず変化率も計算し、これらのデータのすべての組合わせ集合を作り出して、それをバージョン空間として整理している。原理的にこの方法には組合わせ爆発を招くという欠点があるが、7データにしたとたんに遅くなった理由も組合わせ爆発のためと考えられる。ちなみに、12変数のバージョン(スケルトン)空間のノード数は4,095であるが、14変数のそれは16,383である。

計算の高速化のために、瞬時値と変化率の組合わせ数を減らすようにバージョン空間の簡易化を行うこととした。最終的に採用した方法は、瞬時値のみのバージョン空間と変化率だけのバージョン空間に加えて、同一データの瞬時値と変化率の組み合わせを一つのデータ単位と捉えて、このバージョン空間も考え、これら三種のバージョン空間をマージしたようなものを簡易バージョン空間とするという方法である。その根拠は、データ瞬時値の組合わせや同一データの瞬時値と変化率の組合わせには意味があるが、異なるデータの瞬時値と変化率の組合わせには意味はないだろうというところにある。このような簡易化により、7監視データ(14変数)でイニシャライズに30分かかっていたものが0.5秒になり、一回のプラントデータ処理も問題無い速度となった。ちなみに、この簡易バージョン(スケルトン)空間におけるノード数は864個である。簡易バージョン空間のイメージを図2.2-1に示す。

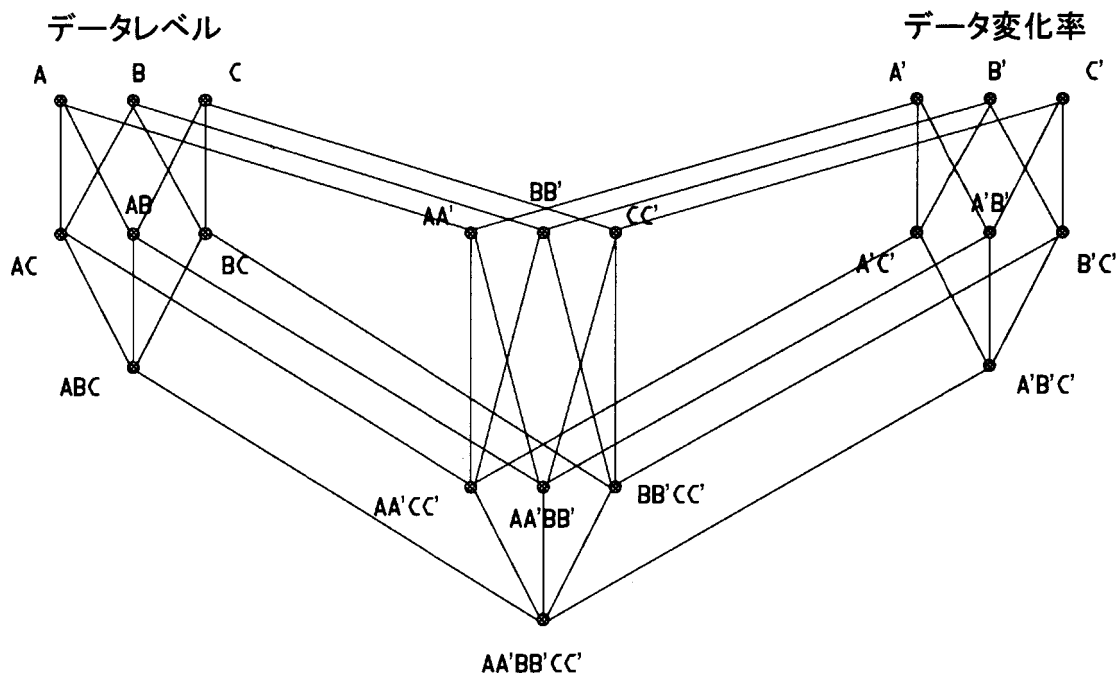


図 2.2-1 簡易バージョン空間のイメージ

上述のアルゴリズム開発はオフラインのリスプシステムで実施したが、開発されたプログラムをオブジェクトデータベース ObjectStore 上に実装された事例ベース推論エンジンに移植し、シミュレータに結合してオンライン化した場合でも、2 秒周期のデータ取り込みに追従して、事例に基づく監視を 5 個すべてのシナリオについて問題無く実施することができた。

2.2.2.2. 音声対話エージェントの性能向上

音声対話では、前年度に柔軟な音声コマンド機能実現を目的とした運転員との対話機能を実現しながら、学会発表に止まって全体システムに組み込むことができなかった。本年度は前年度開発したリスプ言語による音声対話コマンドシステムを C++ 化し、分散協調システムに組み込むとともに、改善項目を洗い出し、実用上の性能を向上させた。ちなみに、音声対話コマンドモジュールは、音声認識エンジン(L&H 社、ASR1600)および音声合成エンジン(L&H 社、TTS32)のほかに、事例ベース推論システム MOP、その上で稼動する言語解析システム DMAP、そして音声対話用アプリケーションプログラムから成っている。

前年度からの改修事項は以下のとおり。

- 事例ベース推論システム MOP の純粋 C++化

前年度において、事例ベース推論システムは ObjectStore の C++クラスを用いて C++言語化されていたが、音声対話には ObjectStore は必須ではない。実用化のためには ObjectStore を使用しないほうが望ましい。そこで ObjectStore なしで音声対話システムが動くように、事例ベース推論システムを ObjectStore なしの純粋の C++に書き換えた。

- 言語解析システム DMAP の C++化

音声対話言語解析システム DMAP も前年度まではリスプ版と Java 版しかなかったが、これも ObjectStore なしの純粋 C++版を製作した。

- 音声対話プログラムの C++化と全体システムへの統合

昨年度作成したリスプベースの音声対話アプリケーションプログラムを、上記純粋 C++の MOP と DMAP を用いて C++言語化し、分散協調システム上に結合した。

- 音声認識性能向上

認識語彙の見直し、定義フレーズの見直し、認識信頼度のクラスタリングを用いた上位候補選択により、認識性能向上を図った。

2.2.3. ミドルウェア化への対応

2.2.3.1. 事例ベース推論システム用 G U I エディタの開発

本プロジェクトでは、事例ベース推論システムとして Schank らによって開発された MOP(Memory Organization Package)システムを用いている。この MOP システムは古いリスプでも稼動するようにリスプの基本的な関数だけでプログラムされていたが、本プロジェクト 5 年の研究開発期間中にそれを現代の Common Lisp に書き換え、分散協調システムに対応するように Java に書き換え、大規模事例ベースに対応するようにオブジェクトデータベース ObjectStore に書き換え、さらに音声対話コマンドシステム汎用化のために純粋 C++に書き換えてきた。

リスプを用いてプログラム開発するときの利点は、関数定義、実行、デバッグの一連のプログラム開発が対話型開発環境の中で連続的に実行できることである。プログラミング中に事例ベースの内容をブラウズすることもリスプでは容易にできるが、その他の言語では困難である。ObjectStore を用いた場合、ObjectStore 用エディタを用いれば、データベースの中身を見ることはできるが、それは事例と抽象度が異なり、事例ベースの中身を見るという意味では非常に使いにくい。そこでミドルウェア化の一環として、プログラマーが事例ベースシステムを開発中に、任意の時点で事例の内容を参照したり変更したりすることのできる、事例ベース用グラフィックエディタを開発した。図 2.2-2に ObjectStore とエディタとの関連を示す。

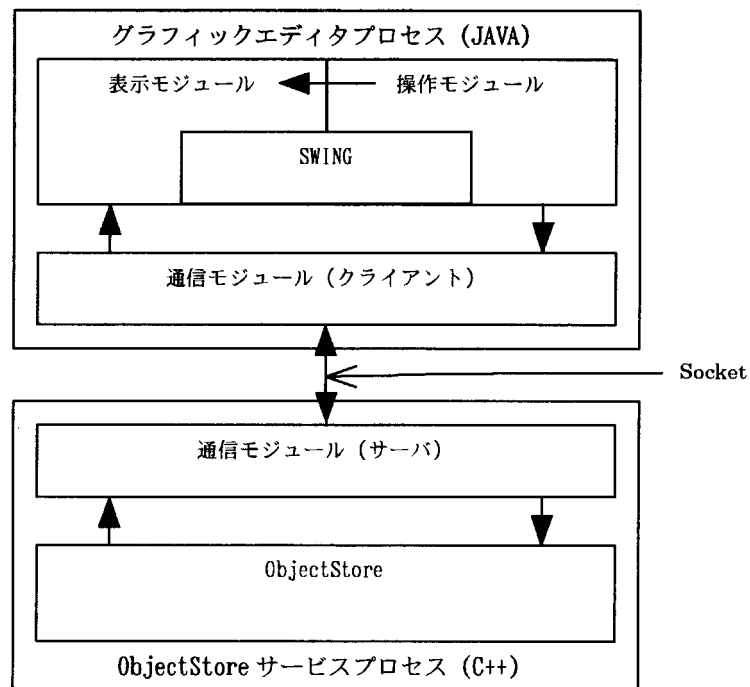


図 2.2-2 事例ベース用グラフィックエディタのシステム構成図

図 2.2-3に事例エディタウィンドウを開いたときの様子を示す。

ユーザは事例ベース中の任意の部分の階層構造をこのようにツリー構造で見ることができ、表示された階層構造をリンクを切ったり付けたりすることで、ObjectStore 中の事例の階層構造を変更することができる。また、各ノード(事例)の中身をまた展開してその内容を変更することもできる。

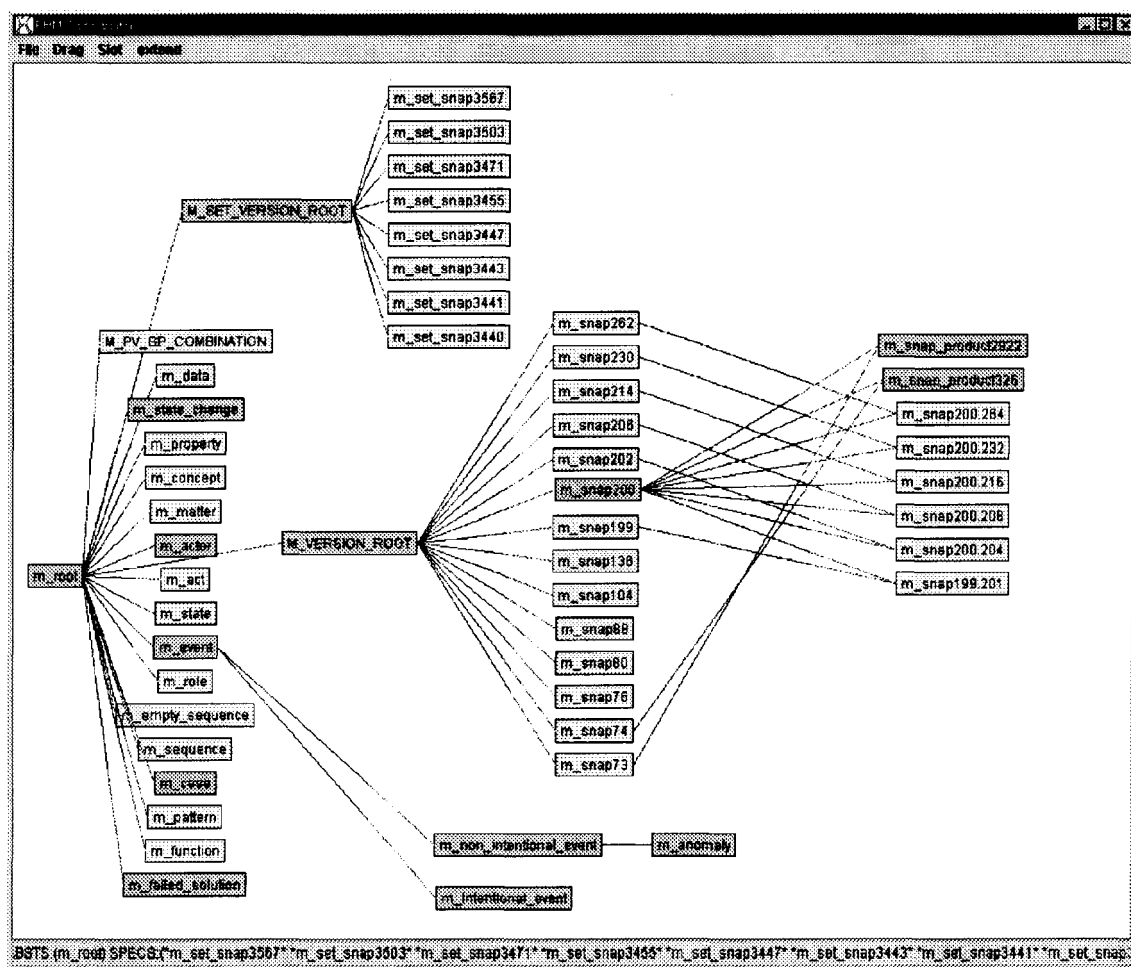


図 2.2-3 事例ベース用グラフィックエディタのウィンドウ表示例

2.2.3.2. 事例ベース推論システム用ブラウザ開発

前述のGUIエディタはプログラマーが使用することを前提としていた。一方、最近発達の著しいIT(Information Technology)を利用して、WEBサーバと事例ベースを結合し、インターネット経由で事例ベースの内容をブラウズする機能を開発すれば、それはミドルウェアとして有効のみならず、将来本プロジェクトで開発した事例推論技術をIT分野に展開することを容易にするであろう。そこで、WEBサーバとしてTOMCATを利用し、事例ベースをインターネットに接続してどこにでもあるインターネット・エクスプローラ(IE5)から事例ベース利用を可能にするWEBサービス技術を開発した。

図 2.2-4に開発したシステム構成を示す。ObjectStore を含む事例サーバは TOMCAT を含む中間ビジネスサービスプロセスと socket 結合され、S 式で通信しあう。S 式というのはリスプで用いられる表現で、括弧の入れ子構造でデータ

表現されるが、これは XML による表現と非常に相性がよく、しかも簡潔に表現できる。

IE5 は TOMCAT に対して URL 出力や POST 要求を行い、TOMCAT 上の Java サブレットがそれを解釈実行し、適切な S 式に変換して事例サーバに要求を出す。事例サーバから S 式で返された結果はやはり Java サブレットによって XML に変換され IE5 に返される。IE5 は XML 文書や XSL 文書を解釈実行できるが、送られる XML 文書の種類ごとに適切な XSL 文書を指定することにより、IE5 は送られたデータを正しく表示する。

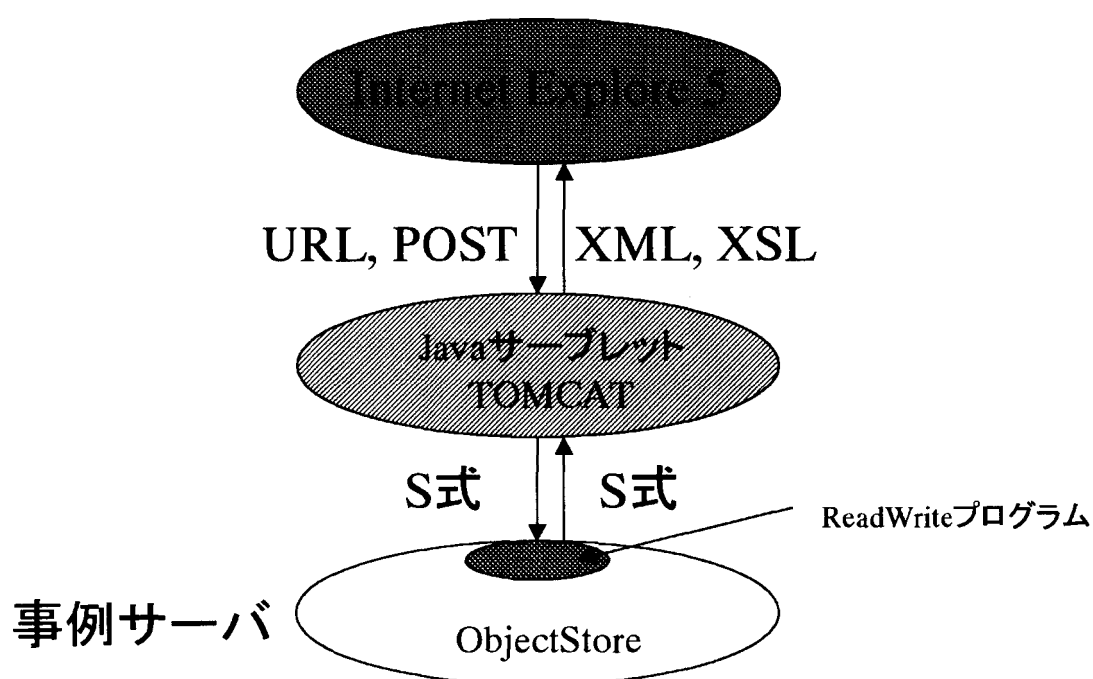


図 2.2-4 事例ベースブラウザのシステム構成図

IE5 に表示された事例ベースの内容の一例を図 2.2-5に示す。

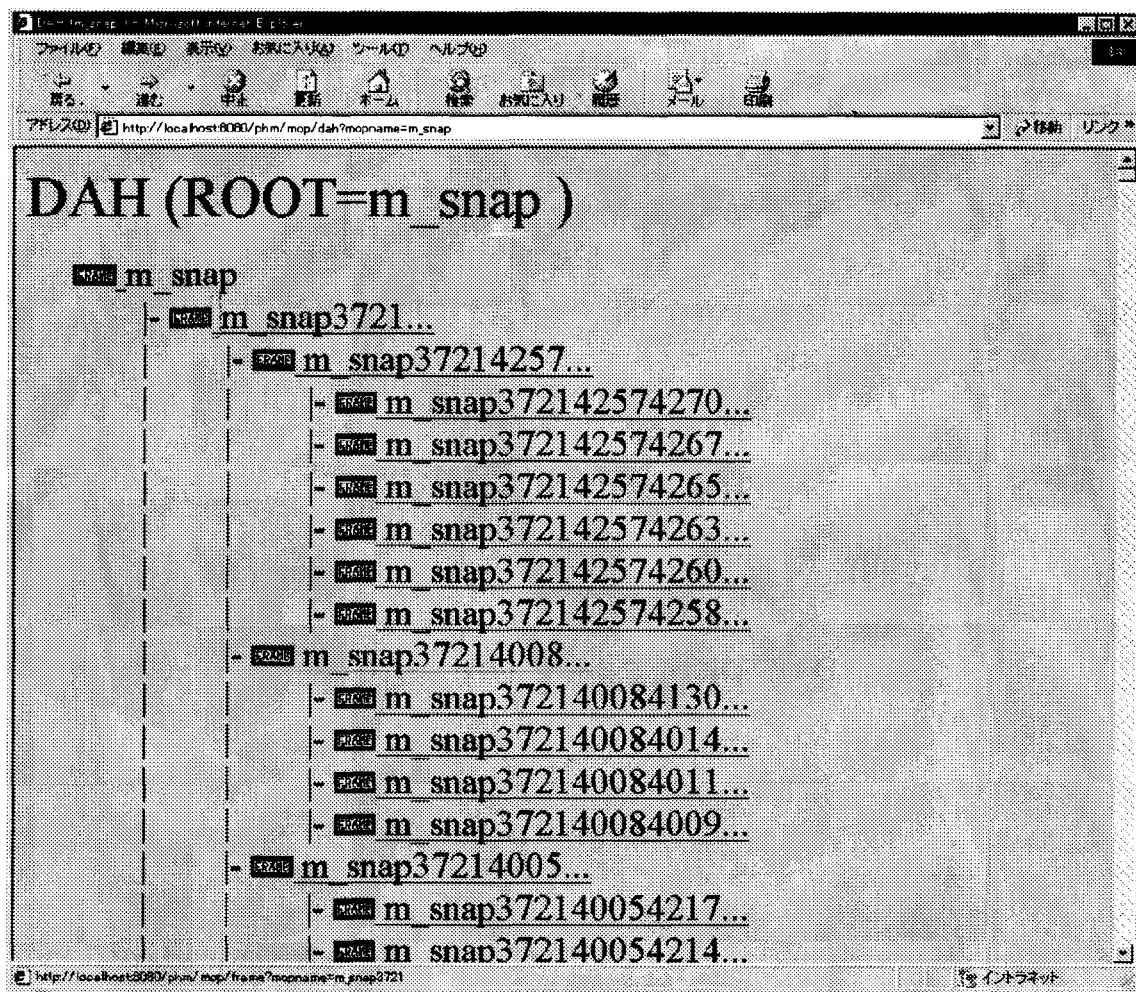


図 2.2-5 インターネットエクスプローラによる事例ベースブラウズ例

図中の **FRAME** をマウスクリックすることで、該当事例の中身をさらに見ることが出来る。

2.2.4.DCS エージェント

DCSエージェントは、従来インタフェースエージェントが担っていた、プラントシミュレータからのデータ収集、サブシステムへのデータ提供、プラントシミュレータのセットポイント変更機能を分離独立させ、さらにDCSに備わっている警報出力機能を高度化させて実装したものである。

前年度システムでは、インタフェースエージェントの事例ベース監視機能による異常検知のほかに、簡単な異常検知は意味表示インタフェースが行っていた。本年度システムの新しい機能として、プラントデータを時々刻々取り込むDCSエージェントにプラントデータ監視機能を担わせることとした。DCSエージェントでは単一のプラントデータを上下限值監視、変化率監視、単調変化監視、MV値監視するのに対して、事例ベース監視エージェントはデータの組み合わせに対して正常／異常判断を行うところが相違点である。

図 2.2-6にDCSエージェントのシステム構成を示す。

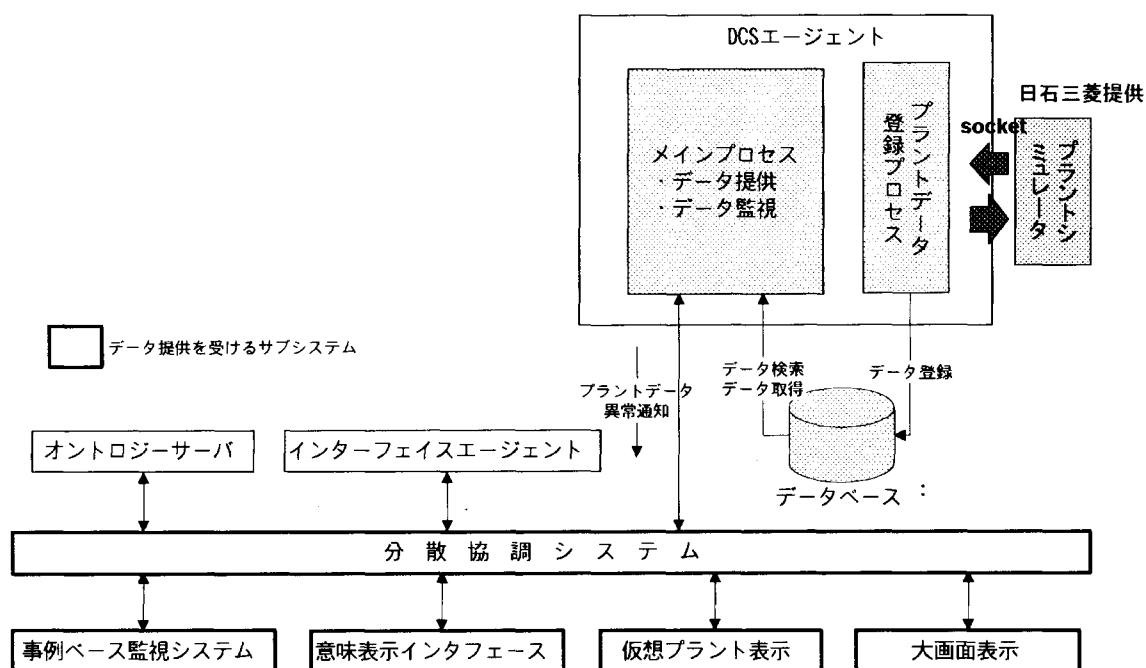


図 2.2-6 DCSエージェントのシステム構成

2.2.4.1. リレーショナルデータベースによるデータ蓄積

プラントシミュレータからプラントデータを収集し、他エージェントに提供する機能は、前年度インタフェースエージェントが行っていたが、そこでは収集したプラントデータをファイルに蓄えていた。本年度はミドルウェア化の観点から、新しくリレーショナルデータベース MSDE を使い、時々刻々取り込むプラントデータをデータベース化し、時刻と時間をキーとして汎用的な S Q L 文を使ったデータ登録・提供機能を実現した。

データベースエンジンを採用した利点として以下が挙げられる。

- S Q L 文によるデータ検索により汎用性が向上した
- システム運転中にデータベースに接続しマルチユーザでデータ処理が可能

2.2.4.2. データ監視機能の高度化

事例ベースデータ監視機能は、データの組み合わせによるデータ監視を行うことができるが、異常の意味を知らず、経験して教えられた異常しか検知できないことが欠点である（未経験状態の検出となる）。一方、現在の D C S ではプラントデータの上下限值監視や変化率の上下限值監視を行っているが、簡単な閾値判別のみで、スタートアップ・シャットダウン、運転条件変更から定常運転まで監視範囲とするために、定常運転状態に対しては緩めの閾値判別となり、異常の早期発見が困難となっている。

D C S エージェントのデータ監視機能は、D C S と同様なデータ単体の監視を行うが、上下限值監視や変化率の上下限值監視に加えて、従来検知困難であった単調変化現象の監視、および M V (maneuvering value) 値監視を実現した

単調変化監視ではデータの変化傾向を最尤法により区分直線近似して、長期間の単調増加単調減少を検出する。図 2.2-7 に単調変化監視の概要を示す。

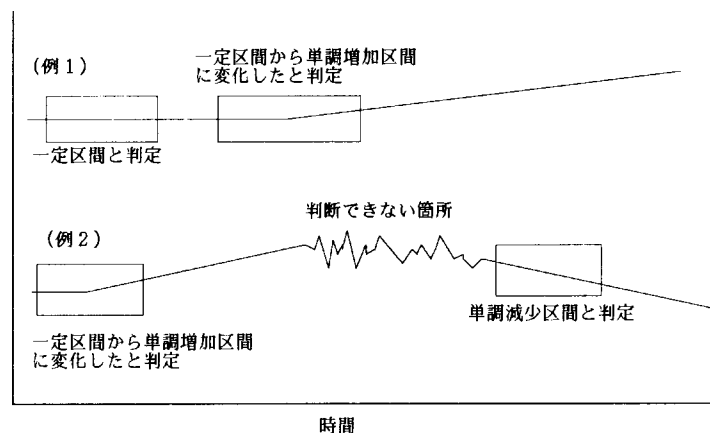


図 2.2-7 単調変化監視の概要

MV値監視では、MV値の大きさだけでなく、時間とPV・SPの乖離した状態も考慮することで、機器に対する制御が有効に行われているかどうかということとを判別する。図 2.2-8にMV値監視の概要を示す。

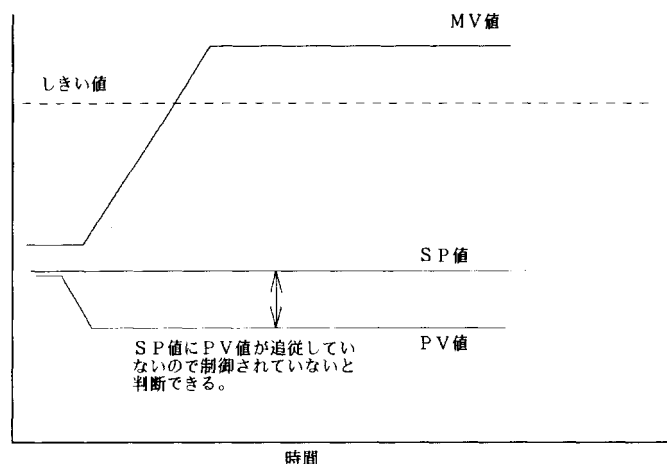


図 2.2-8 MV値監視の概要

2.2.5. 残された問題点と今後の課題

事例に基づくプラント監視では、年度計画ではプラントデータの瞬時値と変化率のみならず、定性的な時系列変化パターンによるプラント状態の分類と類似性判定も実用性の観点から行う予定であったが、できなかった。

一方、本手法によりプラントデータの瞬時値と変化率のみであっても、五つの異常に対して 7 個のデータを監視することで事例に基づくプラント監視が可能であることを示し、実用化に一步近づくことができた。

DCSエージェントではプラントデータの監視に、単調変化監視とMV値(制御量)監視を、従来の上下限值監視、変化率監視と組み合わせることにより、より高機能な監視機能を実現することができた。ここではリレーショナルデータベースを導入し、プラントデータを継時的に収集蓄積しているが、それも無限に蓄積できるわけではない。実用化のためにはデータベースからデータウェア、データハウスへの発展を行い、現在のデータベースを保存し、新しいデータ蓄積を可能にする仕組みや、過去の古いデータを即時利用できるような仕組みが必要である。

2.3. 仮想プラント表示インタフェースの研究

2.3.1. 自律化機能の充実

2.3.1.1. 自律機能の照査

2.3.1.1.1. 自律機能の整理、再定義

VPA の自律機能を整理するとともに、再定義を行った。以下にその内容を示す。

(1) 提示情報種類

VPA がオペレータに提示する情報には、対象プラントの物理的情報と論理的情報の二種類の情報がある。

①物理情報

物理的な情報には、以下のような情報がある。

- －物理的接続情報
- －構造
- －位置
- －ITV による実映像

②論理情報

論理情報には、以下のような情報がある。

- －論理的接続情報
- －語彙、名称
- －系統

仮想プラント表示インタフェースでは、上記の物理情報を知識を利用して論理情報と関連付けてオペレータに表示した。

物理情報と論理情報との関係を以下に示す。

- －物理情報と各機器は機器タグ No. で管理し、機器タグを機器名称に変更するために知識を利用する。
- －論理的接続情報検索：知識を利用する。
- －系統情報：知識を利用する。

(2) 知識の利用

仮想プラント表示インタフェースが利用する知識ベースは、オントロジーサーバの有する知識、すなわち、プラントオントロジーに蓄積された石油プラントにおける各機器間接続情報、機器名等を利用した。

オントロジーサーバとの間は、分散協調を介して通信し、仮想プラント表示

インタフェースの側から、オントロジーサーバに対して必要な情報を要求し、論理的情報を受け取ることとした。

(3) 認知階層

VPA のエージェントとしての自律性について検討する上で、まず、モデル化した人の認知行動を参照した。なんとなれば、エージェントの定義自体が人間の代理としてのふるまいを有するものであるからである。

人の認知行動のモデル化について、最もポピュラーであるのは Jens Rasumussen による「行為の 3 階層モデル」である。(Information Processing and Human-Machine Interaction : An Approach to Cognitive Engineering) すなわち、人間の行為は 3 つの認知的階層に支配されているとしている。この認知階層は、「技能ベース Skill Base」、「規則ベース Rule Base」、「知識ベース Knowledge Base」である。これら階層の簡単な特徴を以下に示す。

Skill Base : 一度その行為を始動させるシグナルが存在すると、意識的な制御のないまま自動的に最終ゴールまで進行する。熟達化の域まで達した日常的な行為、エキスパートの行為などは Skill Base で行われる。

Rule Base : 特定の目的を指向しており、ある目的に達成するために必要な規則を連ねて最終的な行為にまで到達する。Rule Base の行為は if-then で記述することができる。

Knowledge Base : 眼前の事象に対して積極的にモデルを組み立てて関わって行く (model-drives)。このような行為が必要とされるのは眼前の事象があいまいであったり、複雑すぎたり、なじみがなかったりする場合である。

(4) エージェントの自律性

自律性のあるエージェントとは、人間と他のシステムの介入なしに行動することができ、このようなエージェントは自身の内部状態とふるまいの両方を制御することができるエージェントである。そして、設計目的に合致するために柔軟性のある自律行動を行う能力があるエージェントを特に Intelligent Agent と呼んでいる。このような柔軟性のある自律的行動には、以下に示す能力が必要である。

reactivity : 周囲の環境を知覚することができ、Intelligent Agent 自身の設計目的を満足させるために環境内で生じた変化に対して曲がりなりにも時期を逸せず応答することができる。

pro-activeness : Intelligent Agent は本来の設計目的を満足させるために率先して目標方向性のあるふるまい (goal-directed behavior) を提示することができる。

social ability : Intelligent Agent は本来の設計目的を満足するため他のエージェントや人間との Interaction が可能である。

これと人の認知階層とを対比すると Skill Base の行動は reactivity が、Rule Base の行動には pro-activeness が、そして、Knowledge Base の行動には social ability が、それぞれ、必要とされることがわかる。要するに、Skill Base の行動は「Signal により自動的に継続」されるが、この際には Reactivity が必要とされる。Rule Base の行動は「特定目的指向で、ある目的を達成するために必要な規則を連ねて最終的な行為」とするが、この際には Pro-activeness が必要とされる。さらに、Knowledge Base の行動は「眼前の事態があいまい、複雑、あるいは、なじみがない場合にモデルを組み立てて関わって行く」のであるが、Social Ability によりユーザーとの Interaction が必要とされる。

2.3.1.1.2. VPA アーキテクチャの見直し

(1) VPA の境界面について

ここで、VPA における境界面について考察すると、図 2.3-1 に示すように 2 種類あると考えられる。すなわち、User (Operator) と他サブシステムとの間の境界面である。つまり、User との境界面を Upper Layer で、他サブシステムとの境界面を Lower Layer で扱うこととした。また、Middle Layer では VPA 固有の動作を行うものとした。

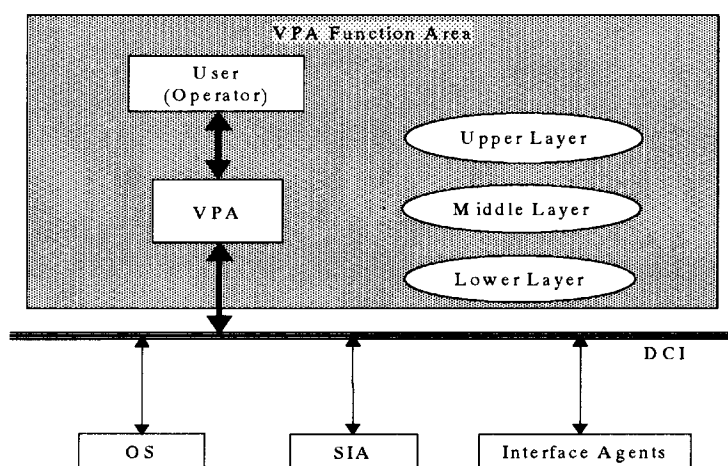


図 2.3-1 VPA の境界面

以上から VPA の機能階層について表 2.3-1 に示すような関係が考えられる。

なお、上記の必要能力の記述中で「本来の設計目的」としているのは、VPA では「運転員が必要とするその時々プラント状態を直感的にわかりやすく表示すること。」に該当する。

表 2.3-1機能階層の関係

階層	認知階層	Intelligent Agent	INTERRAP	VPA
Lower Layer	Skill Base	Reactivity	BBL, World Model	Agent Interaction
Middle Layer	Rule Base	Pro-activeness	LPL, Mental Model	VPA 固有の動作
Upper Layer	Knowledge Base	Social ability	CPL, Social Model	User Interaction

(2) VPA アーキテクチャ

VPA Architecture の見直しを行い、User (Operator) との Interaction、他エージェントおよび各階層間の動作を明らかにした。

ここで、VPA Architecture を図 2.3-2に示す。

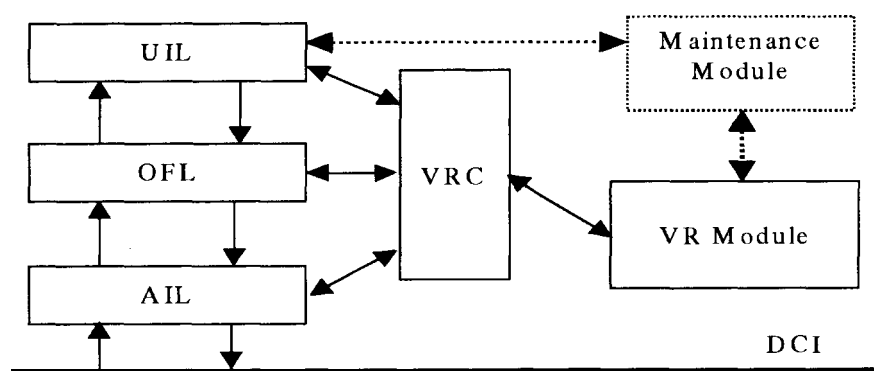


図 2.3-2 VPA Architecture

図中で、VRC は VR 制御モジュールを表し、Maintenance Module はミドルウェア化対応のメンテナンス時の仮想プラント修正用のモジュールである。

(3) 階層毎の動作

(ア) 最上層 UIL: User Interaction Layer

ユーザとのインタラクションを行う階層である。また、ユーザの仮想空間内における操作に対して、ユーザ意図を推定するとともに対応する情報を提示する。

(イ) 中層 OFL: Original Function Layer

自律監視を行う。ここでは、自動パトロールおよび製品性状監視を行う。

(ウ) 最下層 UIL: User Interaction and intention Layer

他エージェントとのインタラクションを行う。すなわち、外部からの要求に対応して必要な情報を提示する。

2.3.1.1.3. 運転員の意図についての検討

(1) 運転員の意図について

当初、マウスカーソルのカレントの位置に応じて、運転員の意図推定を行おうと検討した。すなわち、マウスカーソルを使って特定機器を選択した場合、マウスカーソルが現在位置を中心としたある一定の半径内にある場合、マウスカーソルを中心とした一定の半径内にオブジェクトがない場合および視点移動をしている場合である。

しかし、その後の検討により、仮想プラント表示で表示された仮想空間内をオペレータがマウスにより自在にウォークスルーを行う状況が必要とされないこと、マウスカーソルのカレント位置を仮想空間内で特定するにはマウスカーソル自体の表示を3次元空間内に3Dオブジェクトとして作成し、かつ、3Dマウスカーソルと周辺の3Dオブジェクトとの衝突検知を行う必要性が明らかとなった。マウスカーソルの3D化、衝突検知自体は技術的に可能であるが、特に衝突検知については、マウスカーソルと周辺の3Dオブジェクトとの衝突検知処理に多大の負荷がかかり、プラットフォームのパフォーマンスを勘案した結果、断念せざるを得なかった。

これらの検討の上で、運転員意図推定を行ったのは運転員のマウスカーソルによる3Dオブジェクトの選択についてである。

(2) 運転員意図推定

運転員の意図をマウスカーソルによる3Dオブジェクトの選択から推定した。

具体的には、運転員が仮想空間内で任意の3Dオブジェクトをマウスカーソルに合わせて左ボタンをクリックすることにより選択する。

このような選択動作を行うことは、運転員がその対象に対して明確な関心があるものと考えすることは自明のことである。

ここでは、運転員が対象に対して関心があるとした時に仮想プラント表示インタフェースがどのような情報を運転員に対して表示するかを検討し、以下の

情報を運転員に対して提示することとした。

センサデータ表示：全ての対象に関して関連するセンサデータを表示する。

機器操作パネル表示：対象が操作可能である機器の場合について、機器操作パネルを表示する。

なお、仮想空間に表示される対象には、全て、機器タグ名が表示されているので、運転員は仮想空間に表示された対象が機器タグレベルで確認ができる。

2.3.1.2. 自律機能の拡張

(1) 自律的状态監視方法

自律的状态監視を製品収率、各製品間混合比率に焦点を置いて行った。このような製品状態監視については、他のエージェントでは直接監視を行っていないこと、可視化することでオペレータにわかりやすく提示できることが、その主な理由である。以下に内容を示す。

● 監視対象

製品性状として「製品収率」と「混合比率」を計算して表示するが、異常報知までを行うこととした自律監視では、「混合比率」を対象とした。

「混合比率」では、Naphtha-Kerosene: M_{WK} 、Kerosene-LGO: M_{KL} 、LGO-HGO: M_{LH} 、HGO-Bottom: M_{HB} のそれぞれの各オーバーラップ部分を計算して監視を行った。

● 監視レベル

監視レベルは、「正常」－「注意」－「警告」－「異常」の4状態について行った。この場合、オーバーラップ部分が大きくなるのが注意～異常として運転員に報知する必要がある、逆にオーバーラップ部分が小さくなるのは良くなる方向であるので監視範囲からは除外した。

● 基準値

プラント（シミュレータ）が運転を開始後、定常状態に落ち着いた直後のセンサ値を初期値として、計算式に当てはめ、各オーバーラップ部分の基準値とした。基準値を表 2.3-2に示す。

● 初期値

「混合比率」を求めるためのセンサ値とその初期値を表 2.3-2に示す。

● 計算式

混合比率

$$M_{WK} = -7.89 \times (\text{KERO_IBP} - \text{WSR_ASTM95}) + 230.1$$

$$M_{KL} = -10.2 \times (\text{LGO_IBP} - \text{KERO_ASTM95}) + 99.1$$

$$M_{LH} = -0.323 \times (\text{HGO_IBP} - \text{LGO_ASTM95}) - 6.8$$

(ただし、 M_{LH} 最大値は HGO 収率までであるので $M_{LH} \geq OR_H$ なら、 $M_{LH} = OR_H$ とする。)

$$M_{HB} = 9 \cdot OR_H$$

● 製品収率

全流量 ALL = FC29 + FC33 + FC35 + FC39 + FC42

WSR 収率 (%) $OR_W = (FC29 / ALL) \times 100$

KERO 収率 (%) $OR_K = (FC33 / ALL) \times 100$

LGO 収率 (%) $OR_L = (FC35 / ALL) \times 100$

HGO 収率 (%) $OR_H = (FC39 / ALL) \times 100$

残渣油収率 (%) $OR_b = (FC42 / ALL) \times 100$

● 識閾値

識閾値レベルを以下に示すとともに、表 2.3-2 に各混合比率についての識閾値を示す。

正常

正常 < 基準値 + 0.5 %

注意

基準値 + 0.5 % ≤ 注意 < 基準値 + 0.7 %

警告

基準値 + 0.7 % ≤ 警告 < 基準値 + 1 %

異常

基準値 + 1 % ≤ 異常

表 2.3-2 諸表

初期値			収率(%)	識閾値				
WSR	FC29(Kl/h)	122.2	18.51	混合比率	基準値	注意	警告	異常
	IBP(°C)	57.8		MWK(%)	4.45	4.95	5.15	5.45
	ASTM95(°C)	135.8						
KERO	FC33(Kl/h)	105	15.90	MKL(%)	2.20	2.70	2.90	3.20
	IBP(°C)	164.4						
	ASTM95(°C)	226.5		MLH(%)	3.02	3.52	3.72	4.02
LGO	FC35(Kl/h)	98	14.84					
	IBP(°C)	236		MHB(%)	4.15	4.65	4.85	5.15
	ASTM95(°C)	330.7						
HGO	FC39(Kl/h)	32	4.85	MHB(%)	4.15	4.65	4.85	5.15
	IBP(°C)	300.3						
	ASTM95(°C)	390.5		MHB(%)	4.15	4.65	4.85	5.15
BTM	FC42(Kl/h)	303.1	45.90					
	(Kl/h)	660.3						
ALL	(Kl/h)	660.3						

(2) 自律的状态監視のタイミング

自律的状态監視は、常時、行う。
上述したようにナフサ、灯油、LGO、HGO、残渣油の流量、IBP、ASTM95を取得して、混合比率と照合する。
混合比率に関しては、他のサフシステムでは監視を行っていないため、仮想プラント表示が検知した異常状態は、他サフシステムに対してアラームを発生させる。

(3) サンプル

自律的状态監視の表示サンプルを図 2.3-3に示す。

図では、上からナフサ (WSR)、灯油 (KERO)、LGO、HGO、残渣油、それぞれの製品収量、混合比率を示す。
混合比率が大きくなると、対象の部分が赤く表示され、運転員にわかりやすく表示される。

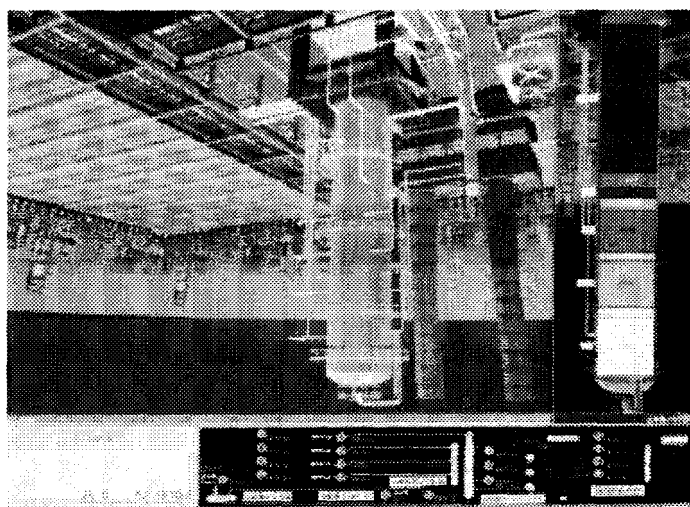


図 2.3-3 自律監視機能・混合表示

2.3.1.3. 複数シナリオによる評価

2.3.1.3.1. 複数異常シナリオへの対応

(1) 各シナリオでの動作

今年度、異常想定シナリオとしてシナリオ 1、シナリオ 3、シナリオ 7、シナリオ 8、シナリオ 9 の 5 シナリオに対応した。

VPA はこれらのシナリオについて、「異常発生機器表示」、「異常原因機器表示」、「現在状態表示」、「近未来状態表示」、「機器操作パネル表示」&「操作手順検索表示」機能をこの順番に異常状態表示シーケンスとしてオペレータに提示した。

ただし、VPA では異常原因機器表示機能以降の情報提示は意味表示インタフェースからの情報提供に依存するため、意味表示インタフェースからの情報がないとこれ以降の情報提示は行われない。

さらに、情報提示のタイミング管理は大画面表示エージェントが行っているため、異常状態においては VPA が自律的に提示情報生成を行うが、その表示については大画面表示エージェントの制御の元に行う。

(2) サンプル

VPA の異常状態表示シーケンスのそれぞれについての提示情報の各画面を以下に示す。

- ・ 近未来状態表示

近未来状態表示画面を図 2.3-6に示す。

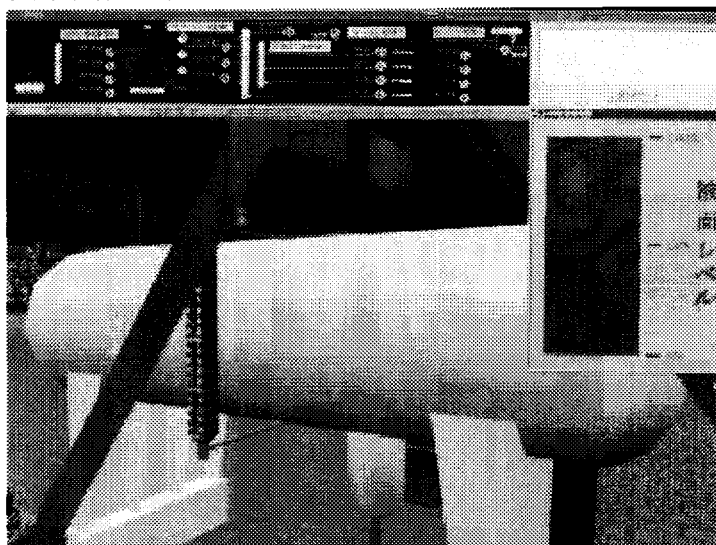


図 2.3-6近未来状態表示画面

- ・ 機器操作パネル表示

機器操作パネル表示画面を図 2.3-7に示す。

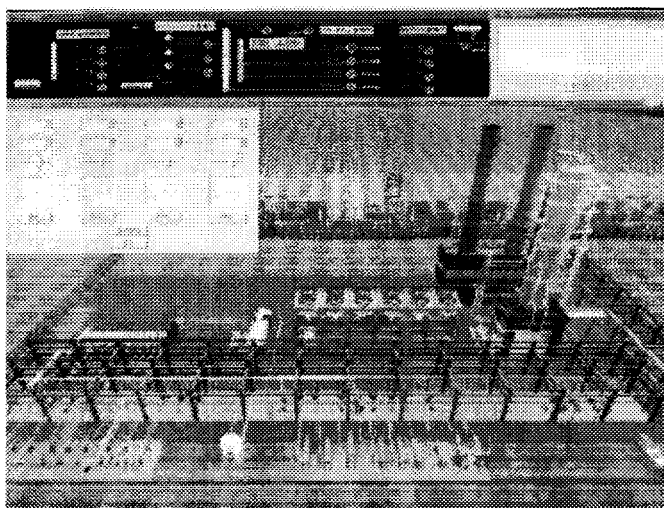


図 2.3-7機器操作パネル表示画面

2.3.1.3.2. 想定外異常への対応

(1) 想定外異常について

VPA においては想定外異常とは、対象とする機器が 3D オブジェクトとして認識されていないことを示す。すなわち、映像上にはモデルとして作成および配置はされていてもシステム的には機器タグがないために認識できない、ことを

指す。

このため、VPA が想定外異常に対応している、ということは、全ての機器に対して 3D オブジェクトとして作成しており、しかも、それぞれに機器タグが付されて個別に認識が可能である、と言い換えることができる。

本年度は、主要な機器について機器タグを付加し、個別に認識が可能な状態とした。

(2) 想定外異常対応

VPA で想定外異常に対応できる機器は、ほとんど全ての機器である。これらのうち、特に今年度は、パイプ類、熱交換器類、ポンプ類を対象とした。

この結果、VPA では仮想石油プラントのほとんど全ての機器に機器タグを付加し、想定外異常には 85% 以上対応できる結果が得られた。

さらに、プロセスデータ可視化の項にも成果を記すが、プロセスデータ可視化も汎用モジュール化を進めた結果、想定外異常対応は満足の行くものとなった。

2.3.2. ミドルウェア化

2.3.2.1. VR ツールのミドルウェア化

2.3.2.1.1. VR ツールのミドルウェア化

VPA においてミドルウェア化は、可搬性、移植性、汎用性が良好なことを目的とした。VPA では、これらを以下のように解釈した。

可搬性は VPA アプリケーションのソースもしくは、実行ファイルを OS 等種類の異なるハードウェアにインストールしてもそのまま、実行可能なこと。

移植性は VPA アプリケーションのソースの大部分を OS 等種類の異なるハードウェアに移植する場合、必要最小限の変更で対応できること。

汎用性は VPA アプリケーションで提示する情報もしくは、情報提示の方法が石油精製プラント以外の産業用プラント、その他システムにも適用可能なこと。

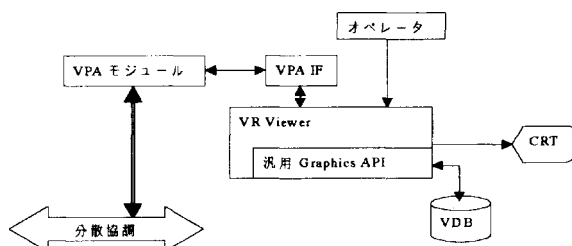


図 2.3-8 ミドルウェア化 VPA 構成図

VPA では、上記の概念の元にミドルウェア化を進めた。

VPA には、既述したように VPA モジュール、VPA インタフェース、VR Tool の大きく 3 つのモジュールから構成される。

このうち、現状では VPA モジュールのみがミドルウェアとしての要件を満たしているが、他の二つのモジュールについては満たしていない。特に、VR Tool については dVISE と呼ばれる専用の VR ツールを利用しているため、可搬性、移植性については対応しているとはいえない。

したがって、ここではこれら二つのモジュールについて、ミドルウェア化を進めることとした。

ここでミドルウェア化 VPA の構成を図 2.3-8 に示す。

このうち、VR Viewer は従来、dVISE が行ってきた機能を行うものであり、VPA IF は VPA と VR Viewer のインタフェースを行う。さらに、VR Viewer は汎用グラフィクス API を使って、3 次元描画を行う機能を有する。

2.3.2.1.2. 汎用グラフィクス API

VPA では汎用グラフィクス API として、OpenGL を利用することとした。この理由として、グラフィクスワークステーションと PC の両方で利用できること、以前作成した映像データベースが流用できることである。

また、OpenGL はそれ自体が 3 次元映像を描画する際のグラフィクスライブラリー群であるので、特にプリミティブなオブジェクトを描画する際にはプログラミングの自由度が大きいことも理由として挙げられる。

2.3.2.1.3. 汎用シーングラフ API

(1) 汎用シーングラフ API

Viewer は、汎用グラフィクス API の他、映像データベースのローダ機能およびシーングラフ展開機能、カリング機能などを有するシーングラフ API を利用して開発期間の短縮化を考慮した。

VR Viewer 開発を行う際には、汎用性のあるシーングラフ (3D 映像データベース) API を選択することが重要である。そこで VPA では、汎用シーングラフ API を利用して可搬性、移植性を高めることを目的とした。

汎用シーングラフ API は、汎用グラフィクス API の上位に位置付けられ、ユーザはプリミティブな描画のためのコーディングを行わず、シーングラフ単位で描画が行えるという利点がある。

(2) xsg

汎用シーングラフとして、研究計画時には Microsoft および SGI が提唱して

いた Fahrenheit と呼ばれる汎用シーングラフ API を使うことを構想した。その後、特に PC 版については Microsoft の xsg が β 版から製品版になるとの情報から汎用シーングラフ API として xsg を選択した。

この理由として、当時、汎用シーングラフ API 開発 Fahrenheit プロジェクトが GWS と PC の両方に対応するとの計画があったこと、低レベルグラフィクス API として OpenGL (xsg は DirectX も対応) に対応するということ、他に汎用シーングラフ API がなかったことが挙げられる。

(3) 汎用シーングラフ API の現状

その後の xsg と Fahrenheit についてであるが、xsg は Version1.0 がリリースされたものの、Microsoft の正式な製品とはならず、メンテナンスモードとなってしまうた。したがって、今後のバージョンアップ、メーカによるサポートの両方は期待できない。また、Fahrenheit プロジェクト自体も現在、頓挫してしまっており、これらから汎用シーングラフ API は存在しないのが現状である。

(4) 今後

現状での調査によると、汎用シーングラフ API ではないが、SGI において IRIX および Linux 対応の汎用シーングラフ API として OpenGLPerformer が開発されている。NT (PC) 対応としては、現状では xsg の他は存在しない。

このため、Linux を OS としたリアルタイムビジュアルシミュレーション用アプリケーション開発が進むことが予想される。

2.3.2.1.4. 汎用 VR Viewer の開発

(1) 目的

現状の VR ツール、dVISE は専用のアプリケーションであるため、VPA のミドルウェア化を考慮した場合、このままでは完全な意味でのミドルウェア構築を行うことができない。そのため、ここでは dVISE と同程度の機能を有する汎用 VR Viewer を開発し、dVISE と代替することとした。

(2) 汎用 VR Viewer の構成

汎用 VR Viewer および汎用 Scene Graph API、汎用 Graphics API 間の構成を図 2.3-9 に示す。

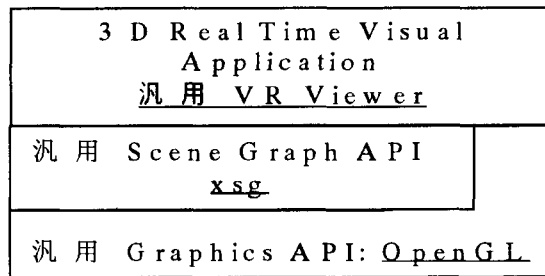


図 2.3-9 汎用 VR Viewer の構成

(3) 機能

現状の VR ツールである dVISE の機能を調査し、これを汎用グラフィクス API として OpenGL、汎用シーングラフ API として xsg を利用して代替機能の実現を行った。

以下に汎用 VR Viewer として実現した機能を示す。

Scene Graph Setup

Scene Graph をロードするとともにセットアップを行う。なお、映像データベースは OpenFlight で開発されているが、汎用 VR ツールでは一旦、OpenGLPerformer の標準フォーマットである PFB 形式に変換した後、xsg 標準フォーマットとしてロード、セーブを行う。

Scene Graph Traversal

Scene Graph のソート、カリング、衝突検知を行う。

3 次元描画機能

投影変換、ビューイング変換、モデリング変換、2 次元クリッピング、ラスタライズ、テクスチャマッピング、Anti-Aliasing、Depth Buffering、ライティング設定等の 3 次元描画処理を行う。

イベント処理機能

VPA モジュールからのイベントおよびオペレータのマウス操作によるイベントに対応した処理を行う。ここでは、dVISE で行っていた他サブシステムからの分散協調を介した「機器表示要求」、VPA モジュールからの異常状態表示シーケンスに基づく「異常発生機器表示」、「異常原因機器表示」、「異常対応機器表示」、オペレータからのマウス操作に従った「ウォークスルー」、「機器情報表示」の各処理を行う。

入力機能

マウスからの入力を可能とする。ここでは、SGI 標準の 3 ボタンマウスを使用可能とした。

編集機能

オフラインで仮想空間に表示された各機器のサイズ変更、色変更、位置変更（移動）、回転を行い、かつ、変更後に別名でシーングラフを保存する。ただし、この場合、シーングラフを保存する際のフォーマットは xsg 標準フォーマットとなる。

2.3.2.2. プロセスデータ可視化モジュールのミドルウェア化

2.3.2.2.1. 汎用的プロセスデータ可視化手法

VPA ではプロセスデータの可視化を行い異常時に異常状態表示および近未来状態表示として情報提示を行った。昨年度までは、ある特定の機器に対してのみ、これらの情報提示を行っていたが、今年度ではミドルウェア化を視野に入れ、異常状態表示、近未来状態表示を汎用的なプロセスデータ可視化表示として行うこととした。

以下にプロセスデータ可視化についての考え方とその実現結果について記す。

（１）可視化対象プロセスデータおよび可視化対象に応じた表現方法

可視化の対象とするプロセスデータは、「圧力」、「濃度」、「温度」、「流量」とした。また、プロセスデータの可視化は、これら単独だけでなく、複合して表現する場合もあることを考慮して表現方法を考慮することとした。

また、可視化表現は対象となるプロセスデータの属性を考慮して行った。ここでは対象となるプロセスデータを「全体」として捉えるもの、「個別」に動きがあるものとして捉えるものの２種類に分類した。

すなわち、「全体」として捉える場合は可視化対象の範囲を一括して扱い、「個別」に捉える場合は可視化対象範囲内で個々の細分化された動きを有するもの、つまり、粒子として扱った。

プロセスデータ種類と気・液の状態、属性として動きを要するか否かについて表 2.3-3に示す。属性のうち、粒子について動的とあるのは粒子自体に動きがあるもの、静的とあるのは粒子としての表現は行うものの全体的な動きがあるものとして表示するものである。

表 2.3-3 属性

対象 プロセスデータ	状態	属性	表現方法
圧力	気体	動的	粒子
	液体	動的	粒子
濃度	気体	静的	粒子

	液体	静的	粒子
温度	気体	静的	全体
	液体	静的	全体
流量	気体	動的	粒子
	液体	動的	粒子

(2) 色の使用

対象となるプロセスデータはほとんどの場合、無色透明であることが多い。従って、可視化の際には通常状態の色をどのようにするのかを決める必要が生じる。

色の使用については以下の基準に従った。(ここで、番号は優先順位を示す。)

対象の固有色。

対象をイメージする色。

周囲の状態から目立たせる色。

また、これらの色について赤、青は異常時のみの使用とするので正常時には使用しない。異常時については識閾値との比較で以下の使用基準によった。

赤…「高い」、「熱い」、「多い」、「濃い」

青…「低い」、「冷たい」、「少ない」、「淡い」

また、それぞれの程度は異常色の濃淡によって表現した。

(3) 組み合わせ種類

プロセスデータを可視化する場合、対象となるプロセスデータ単体で表示する場合と組み合わせて表示した方が良い場合の 2 通りの方法が考えられる。表 2.3-3における温度の属性は静的であり、かつ、一様で全体的な表示であるので、温度と圧力、濃度または流量については、それぞれ、組み合わせて表現することとした。

基本的に属性が動的なもの同士の組み合わせ表示は行わないこととした。また、属性が静的なものについてはその動きを表現しないこととした。

したがって、「温度」についてのみ全体的に一様として表現し、他は「粒子」として扱うこととする。ただし、属性が静的であっても濃度のように「粒子」として表現する場合もある。

(4) 粒子の大きさ

粒子の大きさは、(気体粒子サイズ) > (液体粒子サイズ) として規定した。この理由としては、一般的概念とは逆に以下とした。

最も見にくいものを最も見やすくする。

最も動きがわかりにくいものを最もわかりやすくする。

したがって、圧力 L >流量 M >濃度 S (L , M , S は粒子サイズを表す。)とした。もう少し、詳細に表すと以下ようになる。

圧力・気体 LL >圧力・液体 LS >流量・気体 ML >流量・液体 MS >濃度・気体 SL >濃度・液体 SS

ここで、記号はそれぞれ、次を表す。

LL …粒子サイズ L の大きな方、気体

LS …粒子サイズ L の小さな方、液体

ML …粒子サイズ M の大きな方、気体

MS …粒子サイズ M の小さな方、液体

SL …粒子サイズ S の大きな方、気体

SS …粒子サイズ S の小さな方、液体

(5) 粒子の動き

プロセスデータを組み合わせた場合の動きを以下のように決めた。

圧力

圧力 高：粒子自体の色をピンク、動きを激しく。

低：粒子自体の色を水色、動きはおとなしく。

温度 高：粒子以外の範囲全体を赤。

低：粒子以外の範囲全体を青。

濃度

濃度については静的なものとしての扱いから、粒子の動きはない。

濃度 高：粒子数を増やす。粒子の色はそのまま（固有色）。

低：粒子数を減らす。粒子の色はそのまま（固有色）。

温度 高：粒子以外の範囲全体を赤。

低：粒子以外の範囲全体を青。

流量

流量は流速として表現する。

流量 大：粒子を動かす速度を速くする。粒子の色はピンク。

小：粒子を動かす速度を遅くする。粒子の色は水色。

温度 高：粒子以外の範囲全体を赤。

低：粒子以外の範囲全体を青。

(6) 可視化表現のまとめ

2.3.3. ユーザ評価

石油連盟傘下の各社から運転担当者、運転管理者、計装技術者、8名によりユ一ザ評価を行った。評価内容を以下に示す。

(1) 評価結果

- 評価が高かった項目
仮想プラントでの異常発生箇所はわかりやすい。
仮想プラントとITVとの重畳表示は有効である。
○評価が低かった項目
仮想プラントは実プラントほどの精密さはない。
仮想プラントからの運転操作は必要とは思われない。
温度、圧力、流量などの運転データの可視化は有効ではない。

(2) 評価結果の検討

「温度、圧力、流量などの運転データの可視化は有効ではない。」に対しての考察した結果、汎用性を重視するあまり直感的に訴える表現方法が不足していることが考えられる。しかし、現状の可視化方法を見直した結果、当初の計画にしたがった可視化表現が行われていなかったことが判明した。

(3) 対応

以下に修正内容を記す。
対象となる機器の図を背景に使用する。
対象に応じて、粒子の大きさを変更する。(気体>液体)したがって、(圧力・差圧表示の粒子)>(流量表示の粒子)とし、可視化対象の相違を明確にする。
温度を表現する際、粒子は動かさない。粒子に着色表示は行わない。粒子以外の部分で温度を表現する。下部に温度程度を対照できるバーグラフを表示し、液体の流量を表現する際には流量(流速)に応じた粒子の動きで表現する。粒子は速度に応じて着色する。粒子以外の部分は着色しない。下部に流量(流速)程度を対照できるバーグラフを表示し、現在状態をバーグラフで表す。タイトルを上部に表示する。
液体の流量は上記④で改善が見られない場合は、通常からの遷移状態のアニメーション表示を行う。

液面レベル表現の場合、通常から現在および近未来状態への遷移状態アニメーション表示を行う。

(4) 考察

ユーザ評価結果は、現在、石油精製プラントの運転、保守、計装に携わっている人たちからの意見であり、傾聴すべきものである。しかし、PHMで指向した「人に優しいプラントヒューマンインタフェースの開発」という全く新しい概念は、従来のプラント状態の表現方法と比較して通り一遍の説明では直ちに理解できなかったことも考えられる。したがって、仮想プラント表示インタフェースにおける情報提示の考え方、特にプラントデータ可視化方法についての説明の他、実際に使用してある程度の習熟が必要であることも考えられる。

2.4.分散協調処理技術の研究

分散協調処理技術は、次の 2 大機能を実現することで、分散協調処理機構を利用するサブシステム間の効率的な連携処理を実現し、またサブシステムの追加、変更を容易とすることでシステムの柔軟性を高める技術である。

協調処理機能

システム全体の状況に応じてサブシステムのタスク分担を行い、システム実行効率を向上したり、ロバスト性を向上する。

分散処理機能

サブシステムの追加、変更に対応可能な通信手段を実現し、システム構築を容易にし、またシステムの柔軟性を高める。

今年度は、まず前年度に開発した分散協調処理機構 第 2 版(以降 Ver.2 と略す)の機能、および性能面の評価を実施し、その結果に基づいてミドルウェア化等を睨みつつ機能拡張、改良を行った分散協調処理機構 第 4 版(以降、Ver.4 と略す)を開発した。

また、石油プラント高度情報化システム技術研究開発全体として前年度に引き続き融合システムの開発を行ったが、その分散協調処理基盤開発を効率化することを目的に、前年度開発の Ver.2 を元に重大障害への改修のみを行った分散協調処理機構 第 3 版(以降、Ver.3 と略す)を用意した。

以上の様に分散協調処理機構の開発を 2 本立てにした目的は、今年度の研究の主体となる Ver.4 の開発作業と、融合システム開発で実際に他サブシステムを接続した上で明らかとなる障害や不足機能への対応作業を分離することである。そのため、Ver.4 の実装が完了するまでは各作業を平行して実施し、最終的に Ver.3 での融合システム開発で得られた知見を Ver.4 へスムーズに反映させることができた。以上に述べたような今年度の開発工程を、図 2.4-1 に示す。

	00/4	5	6	7	8	9	10	11	12	01/1	2	3
1. 融合システム開発												
1.1 Ver.3改修			→							→		
2. 分散協調処理の改良												
2.1 前年度版評価		→										
2.2 実現方式検討			→									
2.3 Ver.4の実装									→			
2.4 Ver.3改修の反映										→		
2.5 評価											→	

図 2.4-1開発工程表

2.4.1.機能、性能の評価

Ver.2 は、昨年度、石油プラント高度情報化システム技術研究開発において初

めて開発された各要素技術の融合システムの分散協調処理基盤であり、インタフェースエージェント、仮想プラント表示インタフェース、プラントオンロジー、意味表示インタフェースの各サブシステムの機能連携を仲介する基盤技術として実使用され、貴重なデータ、知見等を得ることができた。

ここでは、今年度初頭に実施した Ver.2 の評価の結果明らかとなった、機能面における課題、性能面における課題について説明する。

2.4.1.1. 機能面における課題

2.4.1.1.1. サブシステム間同期機能

前年度融合システムでは、前年度成果報告書に記載の通り、各サブシステムがそれぞれ分担して処理した結果をユーザ(オペレータ)に矛盾なく効率的に提示するために大画面表示サブシステムに提示タイミングの同期機能を実装した。

しかし、各サブシステムは情報提示の度に大画面表示サブシステムの同期機能を利用して他サブシステムとの提示タイミングを調整する必要があるため、本来はサブシステム間の同期機能として連携基盤ソフトウェアが提供すべき機能と考えるのが自然である。そこで今年度は、分散協調処理機構に各サブシステム間の同期機能を実現するための方式検討を行った。図 2.4-2に、検討したサブシステム間同期機能のインタフェースを示す。今年度検討した同期処理方式では、システムを構成する全サブシステムの共有する同期ポイント情報を用いる。同期処理開始時に分散協調処理機構から全サブシステムに同期ポイントを通知し、各サブシステムが同期ポイントまで処理を実行した後、分散協調処理機構が全サブシステムが同期ポイントまで処理実行が完了したかどうかを確認し、全サブシステムに同期完了を通知する。

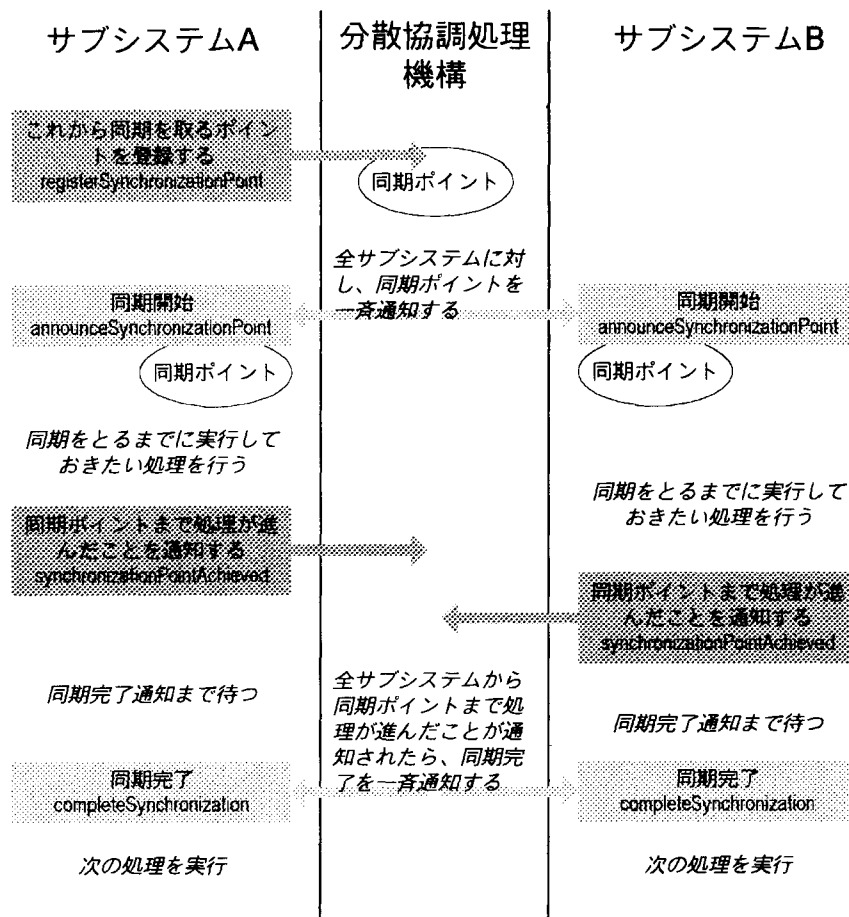


図 2.4-2サブシステム間同期機能のインタフェース

2.4.1.1.2. タスクエージェントの高度化

分散協調処理機構がタスクエージェントとして内部的に持つサブシステム固有の知識は、サブシステムの起動時に送信されてきた静的な情報と、実行中のタスクによる負荷情報であり、サブシステム間でタスクを割当ての際に参照される。Ver.2での動作状況を観察した結果、各サブシステムが行った処理結果や、また特定のサブシステムが管理することになっているデータなどについての要求処理が非常に多く、そのために分散協調処理機構を経由してサブシステム間で往復する通信がほとんどであることが判った。

そこで、タスクエージェントの持つ知識データベースを拡張し、サブシステムが他サブシステムから頻繁に要求されるような情報をタスクキャッシュとして管理する機能を検討した。図 2.4-3にタスクエージェントのタスクキャッシュ機能のインタフェースを示す。今年度検討したタスクキャッシュ機能では、タスクキャッシュと登録するデータの管理はサブシステムが行う。

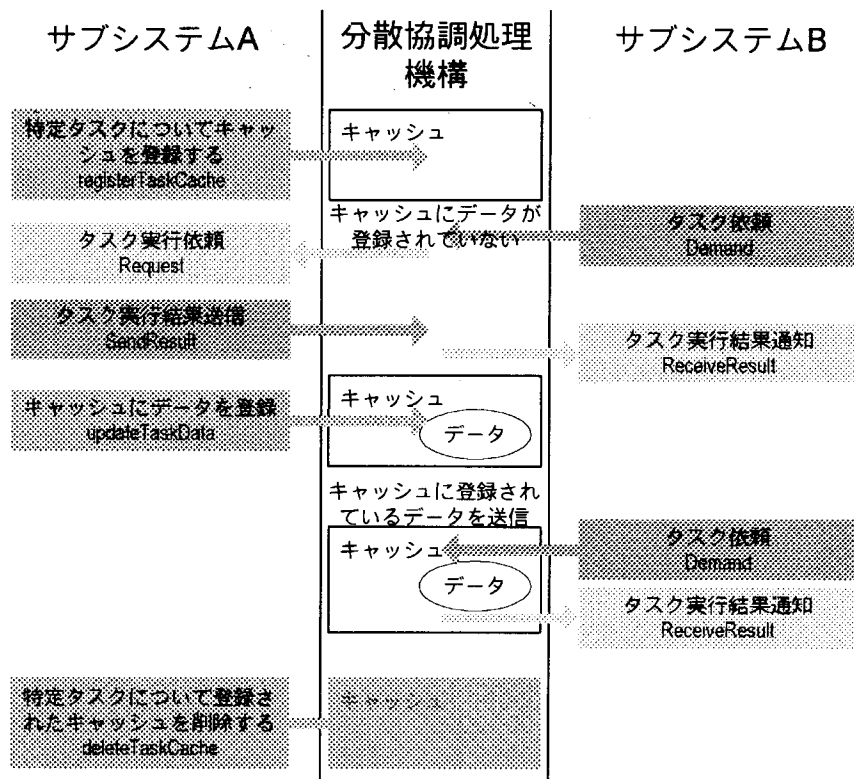


図 2.4-3キャッシュ機能のインタフェース

2.4.1.2. 性能面における課題

2.4.1.2.1. 安定化

性能面では、分散協調処理機構のメモリ使用量がデモシステム実行中に増加し、連続運転に支障をきたすという問題があった。Ver.2の分散協調処理機構では、サブシステムからタスク依頼に対してタスクエージェント間の協調処理を行うために多数のスレッドを起動しているため、融合システムの規模でタスク依頼が頻繁に行われると、個々のスレッドを処理し終えた後で実行される利用メモリの償却処理がなかなか終わらずに上記のような問題を引き起こすことが第一の原因と考えられる。

本問題に対応するためには、タスクエージェントによる協調処理機構でのスレッド利用方式を見直す必要があり、比較的規模の大きな改造を要するが、プロジェクト終了後の実用化へ向けても重要性、緊急性の高い問題であり、早期に改良作業を実施することとした。

2.4.1.2.2. 負荷分散

分散協調処理機構は、大まかには図 2.4-4に示す様に構成されている。協調処理サーバは、協調処理機能と分散処理機能をサブシステムに対して提供する

サーバである。サブシステムは、協調処理クライアントとして協調処理サーバを利用するが、そのプログラムインタフェースとして分散協調 API(Application Program Interface)を利用する。分散協調 API は、分散処理機能を実現する機構の一部である。

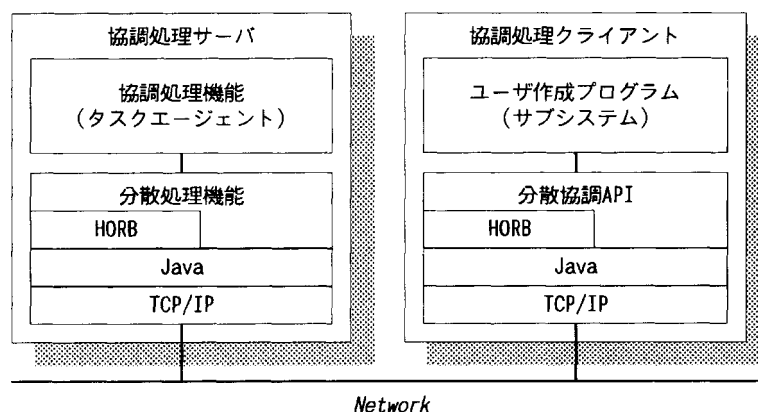


図 2.4-4分散協調処理機構 Ver.2 の機能構成

Ver.2では、全サブシステムへの一斉通知機能や、特定サブシステムへの直接指名によるタスク依頼などでは協調処理サービスを必要としないため、これらの機能を分散協調 API が実行するように実装した。しかし、通信時には相手側のサブシステムの IP アドレス等を参照する必要があるため、これらの情報は協調処理サーバの分散ディレクトリサービスが一元管理しているため、結果として協調処理サーバの負荷はそれほど低減されていなかった。

また、本来、協調処理機能を協調処理サーバに一元的に実装した目的は、タスク分担実行がシステム全体性能にさほど影響しない様に、ネットワークを介さないタスクエージェント間通信を実現するためであったが、タスクエージェントとサブシステムが分離したことにより、サブシステムの実行確認や、またサブシステムが他からのタスク依頼によってではなく自発的に何らかのタスクを実行した際の状況通知などのための通信が必要となり、これらの通信処理も協調処理サーバの負荷を上げる原因となっていた。

そこで今年度は、分散ディレクトリサービスを別機能として実装し、またタスクエージェントとサブシステムの通信処理を見直すことにより、Ver.2 開発時の目標であった協調処理サーバの負荷低減を実現することを決定した。

2.4.2. 分散協調処理機構の改良

2.4.1章で述べた評価結果に基づき、改良すべき項目を挙げ改良版の分散協調処理機構を開発した。まず、年度内に開発を完了するために、改良項目に優先

順位を付け、また開発効率を上げるために利用可能な既存技術の調査を行った。その結果、重要性和緊急性の高い「安定化」と、前年度の目標達成のための積み残しである「負荷分散」の 2 点を今年度を実施することとし、新たな機能追加となる「サブシステム間同期機能」と「タスクエージェントの高度化」に関しては方式検討を行うこととした。

2.4.2.1. アーキテクチャの見直し

分散協調処理機構は、技術調査と汎用性のあるインタフェースの検討のため、H10 年度の早い時期にプロトタイプ開発に着手し、その後、基本的にはこの初期プロトタイプに機能追加を行う形で実装されてきているため、今年度の実施項目として決定した前記 2 点の改良作業を行うためには、根本的なアーキテクチャから見直す必要があった。

そこで、改良作業の軽減と、またこの機会を利用して分散協調処理機構の標準技術との融合を狙い、関連する技術動向の再調査を行った。まず開発言語や環境については、これまでの成果の再利用という面から引き続き Java を利用することとし、Java 上のエージェント技術、分散オブジェクト技術を調査した。その結果、今年度開発する Ver.4 へ適用する候補として「JADE」、「Jini」の両技術に着目し、適用方式検討を行った。

2.4.2.1.1. JADE

JADE(Java Agent Development Framework、<http://sharon.cselt.it/projects/jade/>)) は、FIPA(Foundation for Intelligent Physical Agents)に準拠した Java エージェント構築環境である。分散環境において各ノードに分散したエージェントを作成できる他に、我々が開発した分散協調処理機構と同様の単一ノード上に複数のエージェントを集中させたようなモデルに対しても考慮されており、エージェント間通信の高速性についても期待できる。

JADE を利用する場合、我々が作成した分散協調処理機構は全て JADE をベースに作り直すことになるが、タスクエージェント機能、また分散ディレクトリサービスの機能については開発環境は揃っており、作業軽減の目的は果たせそうである。しかし、Ver.2 で実装したタスクキャンセル機能等は JADE を拡張して実装せねばならず、JADE の内部構造についても調査が必要となるため、今回の開発での利用は断念した。

2.4.2.1.2. Jini

Jini は Sun が中心となって標準化や開発が進められている Java を利用した分

散処理基盤であり、強力な分散ディレクトリサービスを実現するものとして近年注目を集めている。

Jini 自身にはタスク分担を実現する協調処理的な枠組みは規定されていないが、分散協調処理機構の分散処理機能を Jini に置き換えるだけでも、我々の開発作業をタスクエージェント機能の改良に集中することが可能であり、また標準技術と我々の研究開発成果との融合という点でも意義のある応用と思われる。そこで今年度の開発では、Jini 上に協調処理機能を実現し、分散協調処理機構 Ver.4 として開発を行うこととした。

2.4.2.2. Jini 応用による分散協調処理機構

図 2.4-5に Jini を応用し、再構築した Ver.4 の機能構成を簡略化したものを示す。Ver.4 では、Jini Lookup サービスと、分散トランザクション機能を利用した。

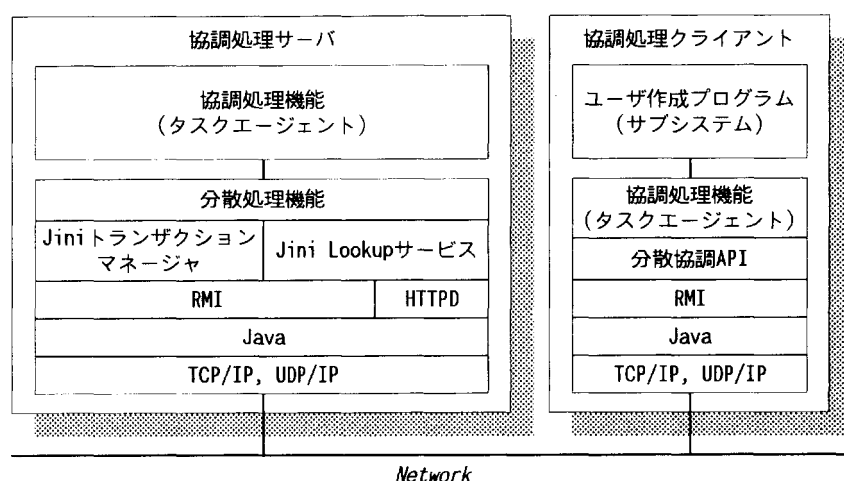


図 2.4-5分散協調処理機構 Ver.4 の機能構成

2.4.2.2.1. 分散処理機能の負荷低減

Jini Lookup サービスを利用することで、分散ディレクトリサービスと協調処理機能を完全に切り離すことができ、よりシンプルなアーキテクチャが可能となった。図 2.4-6に、Ver.4 でのサブシステム間の直接通信のフローを示す。なお、Jini Lookup サービスには、Jini 1.1 に付属する reggie を利用した。

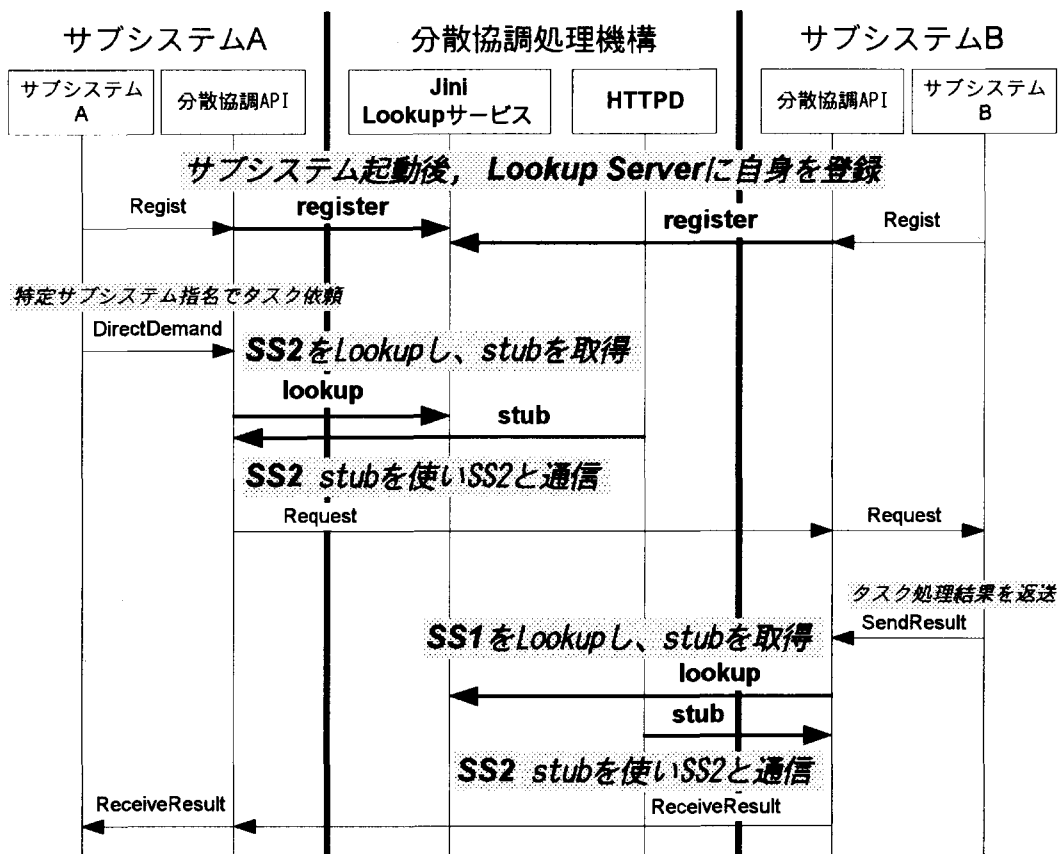


図 2.4-6サブシステム間の直接通信時のフロー

2.4.2.2.2. 協調処理機能の見直し

タスクエージェントの実体を協調処理サーバから分散協調 API 側に移動し、タスク分担を実行する場合にのみ協調処理サーバにタスクエージェントが集約されるようなアーキテクチャに変更した。

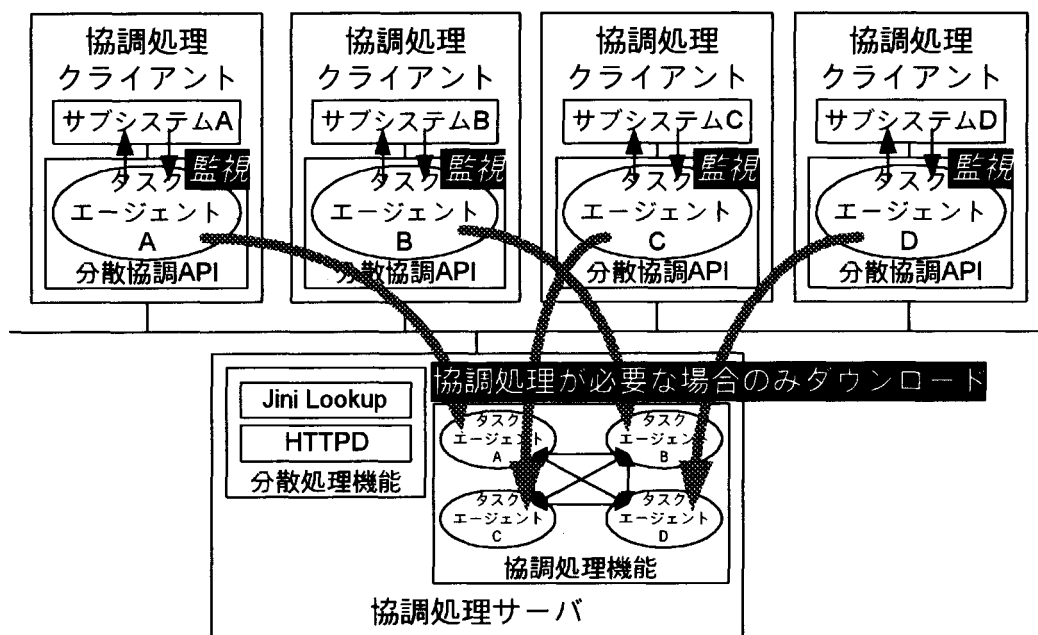


図 2.4-7タスクエージェントの配置

このようなアーキテクチャの変更により、通常時にはタスクエージェントとサブシステムは同じノード上にあり、タスクエージェントはサブシステムの実行確認や、またサブシステムが他からのタスク依頼によってではなく自発的に何らかのタスクを実行した際の状況をローカルに監視できるため、Ver.2 に比べて通信量の低減や負荷の軽減を実現することができた。

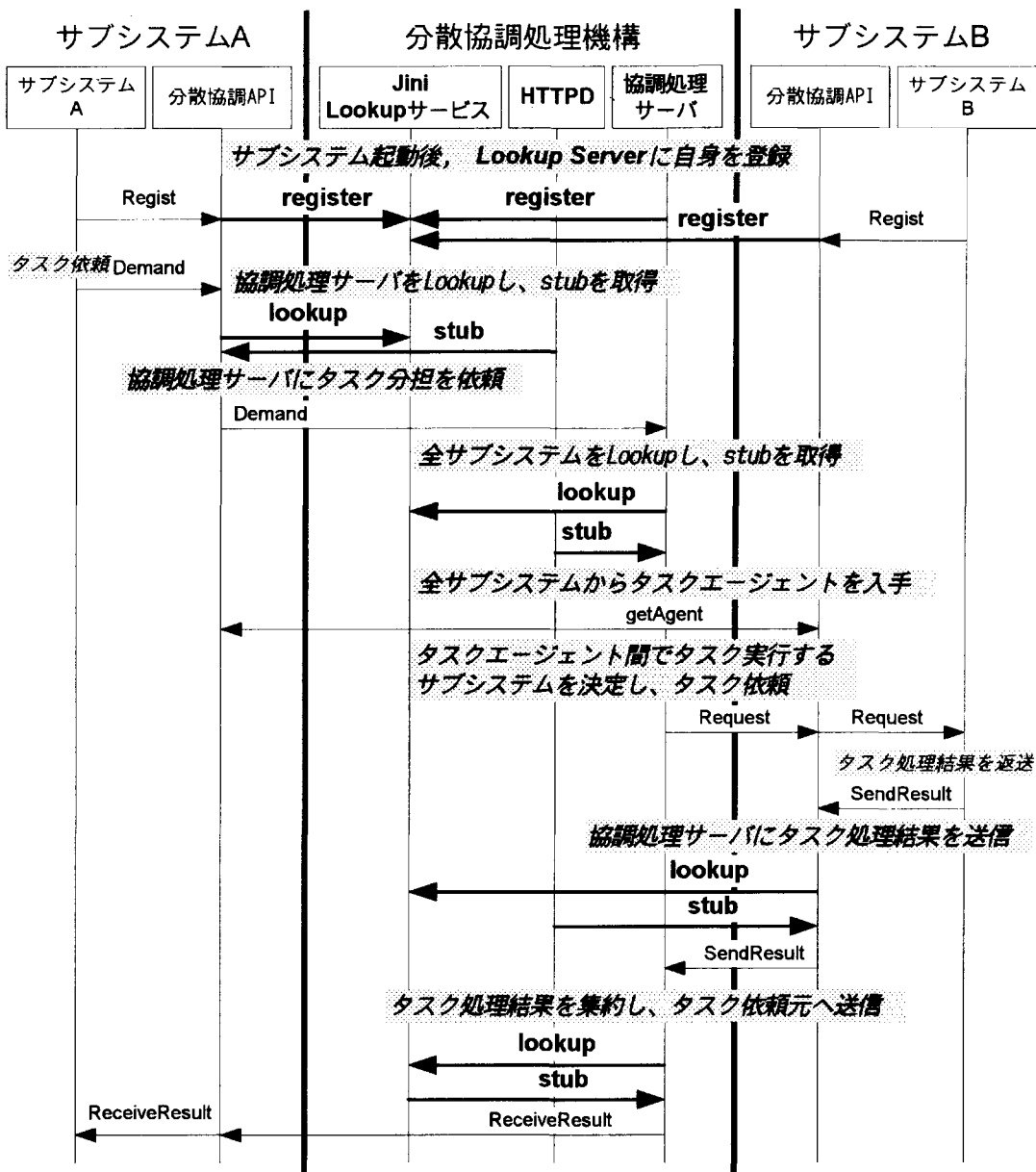


図 2.4-8タスク分担処理の処理フロー

2.4.2.2.3. 分散トランザクション機能

Ver.2では、タスク依頼に対して結果が返らない場合や、またオペレータによって処理が取消された場合に対する機能として、タスク依頼キャンセル機能を検討し実装したが、タスク分割時における処理に一部不完全な部分が残っていた。一方Jiniでは、分散システムの処理単位(トランザクション)で複数の分散サービス(分散協調処理ではサブシステムに相当する)間で処理中止を実現するための機能として分散トランザクション機能が定義されている。Ver.4の開発では、Ver.2のタスク依頼キャンセル機能を継続するのではなく、分散システムの

標準技術として検討されている Jini 分散トランザクション機能の適用し、検証することとした。

一般的に、RDBMS(Relational Data Base Management System)等で実現されているトランザクション機能は、そのトランザクションを管理している機構(例えば RDBMS)自身が機能を提供し、整合性の保証を行うが、Jini における分散トランザクション機能では、トランザクションに関連する全ての分散サービス自身がその機能を提供し、全体の整合性の保証を行う必要がある。Jini では、そのトランザクション管理体系を “Manager”、“Client”、“Participant” の 3 つのオブジェクトにより実現することを決めている。

Manager

トランザクションを生成・管理するオブジェクトであり、他 2 つのオブジェクトの関連や呼出しを管理する。

Client

トランザクションのセマンティクス・オブジェクト(トランザクションを表すオブジェクト)を生成し、実際にトランザクション処理を起動する。

Participant

トランザクション中の個々のサービスを実行するオブジェクトであり、Client が生成したセマンティクス・オブジェクトを共有してトランザクションの一貫性を保証する様に実装する必要がある。

分散協調処理では、Participant が各サブシステムに相当し、Client、Manager はサブシステムに分散トランザクション機能を提供するサービスで、Ver.4 では協調処理サーバに Client の機能を実装し、Manager には Jini 1.1 に付属する mahalo を利用した。図 2.4-9に、分散トランザクション機能利用時のフローの例を示す。なお、図中のフローでは、Jini Lookup サービスと HTTPD による処理は省略している。

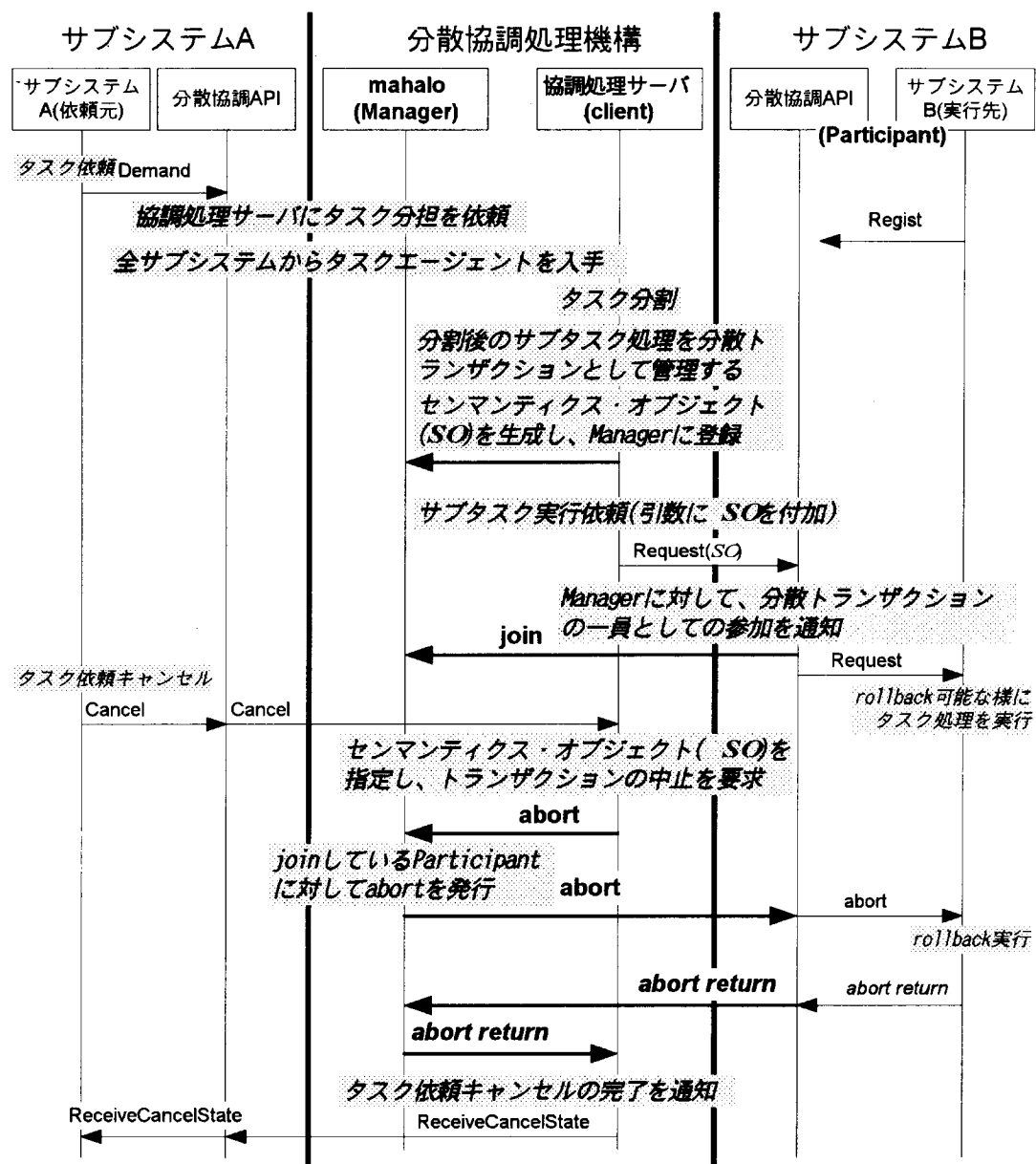


図 2.4-9タスク依頼キャンセル時の処理フロー例

2.4.3. 融合システム開発とそのフィードバック

融合システム構築用に作成した Ver.3 では、WG 各社での利用により多数の障害の発見、また不足機能の洗出しなどが行われた。これらの結果は、平行して開発を行った Ver.4 にも反映され、より完成度を高めることができた。

2.5. プラントモデルとインタフェースシステムの研究

2.5.1. 本年度の概要

当グループの役割は、昨年度から引き続く実施業務として、各サブシステムが必要とする装置機器データ、装置プロセスデータ、プラント運転データ、プラントモデルデータ、プラント運転に関わる情報の提供と、各サブシステムへのサポートを行った。具体的な業務内容としては、プラント異常シナリオの追加作成と精度の向上を図り、シミュレータ上で再現実証し操作方法を提供した。また意味表示インターフェースで作成中のプラント簡易モデルのチェック及びチューニングを行った。また、故障機器表示に新しく配管を含めることを提案しデータを提供した。

更に前年度からの継続業務とは別に、本年度は研究開発の最終年度として、プロトタイプシステムの各成果物が実用化に向けて有効であるか、ユーザーの立場から評価を行った。

2.5.2. 異常シナリオの内容充実とシミュレータでの検証

昨年度までに9個のシナリオを作成した。シナリオは各要素技術でプラントの運転解析、異常診断、異常原因推定、及びその異常対応作業の導出など、一連の業務の流れでどのような機能が必要かを洗い出すために必要である。前年度作成した「ポンプ性能低下によるリフラックスドラム液面上昇」、「燃料ガス比重変動による加熱炉出口原油温度低下」のシナリオに引き続き「トッパー圧力上昇」「加熱炉原油チャージ量減少」「加熱炉排ガス CO 濃度の上昇」「灯油製品性状悪化」「残油・原油熱交換器漏洩」「加熱炉チューブコーキング」「トレイフラッディングトラブル」のシナリオをシミュレータで再現した。

評価用に実装されたシナリオは5個であり、提供したシナリオは十分の数である。

更に、シナリオに対応するだけでは、機能の評価はできないため、シナリオ以外の異常にも、ある程度対応できるようなシステム作成を要望し、部分的には対応可能となった。

2.5.3. 各サブシステムの検証

2.5.3.1. 意味表示インターフェース用データの収集、検証

意味表示インターフェースで開発している、トッパーの高速簡易シミュレータモデルへの計算式、経験式を提供している。簡易モデルの計算結果を検証し、適切ではない結果に対してはチューニング方法を検討し提示した。本年度は最終年度であり、最も複雑な主精留塔モデルの精度向上を計った。精留塔内部ト

レイの気液負荷は、別途、精度が高いスタティックなプロセスシミュレータを使用して計算し、結果を比較して簡易モデルの精度を確認した。

また、5個のシナリオについて、システムが診断した異常原因候補の妥当性、診断ルールのチェックを行った。さらに、異常に対する、対応操作の効果を評価する方法について、チェックを行った。

2.5.3.2. プラントオンロジー用データの収集、検証

昨年に引き続き、システムが提供する情報が、運転員に分かり易い言葉になっているか検証し、不適切な言葉は修正を依頼した。

更に、異常発生時の情報に、配管情報について項目追加を依頼し、配管情報を提供した。プラントの機器は配管で接続されており、異常が想定される部分に繋がる配管ルート情報は、複雑に入り組んだ配管を明確にしめすことができ、オペレータにとって有効と思われる。

2.5.3.3. VR 用データの収集、検証

昨年に引き続き、加熱炉、蒸留塔など仮想プラントの臨場感を高め、プラントの状態監視を直感的に行える表示方法についてサポートを行った。また原油や製品の配管ルートを作成し提供した。オンロジーとの連携で、異常発生箇所の把握、異常内容が更に正確に把握できるようになった。

2.5.3.4. 大画面表示画面の機能検証

昨年のパワーポイントベースのイメージの画面設計から本年度は実際のデータを受け渡しする実用ベースのシステム構築が行われたが、使用する立場から画面の構成、機能について提案を行った。

2.5.4. 類似システム調査

ある報告書に、某計装メーカーが類似システムを開発しているとの情報があり、調査した。システムの概要は、仮想プラント上に、運転操作手順に基づいて、次に操作すべき機器と操作手順を表示するものであった。毎朝スタートアップし、夜間停止する装置で、毎日使えるものであったが、実用化には至っていない。今回のシステムと比較し、機能的には本研究開発で目指している機能の一部しか使用しない小規模なものであるが、実用化に対する費用対効果の判断が難しい様である。

2.5.5. ユーザー評価

今年度の大きな検討項目はユーザーの立場からのシステムの評価である。本

プロジェクトには二つの大きな目標がある。一つは運転員に有効な情報の提供、もう一つは情報の効果的な表示方法の提供である。当初10年間予定のプロジェクトが前期5年間で終了となったためか、このシステム開発ではユーザーの立場から見て実用化に焦点を当てた適用業務の開発はこれからと言う状況で、実用的に完成した部分は余り多くない。たとえば、プラント運転に役立つ情報の技術的評価より、分散協調システムによるデータのやり取りでの、操作や使い勝手の良いユーザーフレンドリーなシステムの構築が主に追求された。実用性を評価するには、異常シナリオどうりの画面展開だけではなく、シナリオ以外の想定外異常にも対応可能なシステムでなければならないが、今回はこのように、幾つかの異常シナリオには対応するものの、シミュレータで種々異常を発生させての対応結果が十分には実証されていない状況でのユーザー評価となる。

ユーザー評価の一つとして、石油連盟を通して石油各社に対して、また日本能率協会を通して、石油精製以外の多様な業種の企業の参加を得て、説明会とデモンストレーションを行い、参加メンバーに評価のためのアンケートをお願いした。アンケートの結果は、全体的に肯定的な評価であり、異常事例データベースによる類似異常の早期発見、対応操作提示、あるいはモデルベースでの異常原因推定、異常対策指示の提示等の機能的なものについての期待は大きい。アンケートの点数を100点満点として、石油連盟の場合、約60点、日本能率協会の場合、約70点であった。この差の理由は、石油連盟の場合、身近な装置であり、仮装プラントの臨場感、異常対応検出、対応操作候補の導出など、現在使用している分散型計装(DCS)以上の機能を期待しており、完成度への要求水準が高い事、日本能率協会の場合、説明会が時期的に後であったため改良が進んだ事、馴染みの無い装置のため、若干自分の問題としてとらえる当事者感が薄い事が推定される。

石油連盟と日本能率協会へのデモを実施後のアンケート結果を表2.5-1に示す。数字は評価分類を最良項目を100、最悪項目を0としての平均値である。

表 2.5-1 アンケート結果

アンケート解答者	石油関連企業 (石油連盟)	石油外企業 (日本能率協会)
アンケート項目	1/24 実施	3/8,9 実施
異常検知機能	60.4	71.6
事例ベース監視機能	77.1	71.2
仮装表示の機能、出来栄え	55.0	71.4
モデル推論機能	59.4	78.0
事例ベースからの情報	58.3	76.2
大画面の機能、出来栄え	54.0	67.1
音声コマンドの機能、出来栄え	57.5	64.4
オントロジーの日本語生成の妥当性	71.4	66.7
システム管理の容易性	42.9	52.4
評価の平均値	58.1	69.4

2.5.5.1. 石油連盟ユーザーアンケート結果

1) プラントデータ監視システム (DCS エージェント)

プラントシミュレータからの秒単位の運転データを常時監視し、異常を検知し、他のサブシステムに異常を通報する。システムを稼働させるトリガーになる技術である。運転データの上下限、変化率、単調変化などをチェックする。デモの結果、チェック機能、異常の検知タイミングはほぼ妥当との評価を得た。変動すべきなのに変化しない場合の検知ができる機能の要望があったが、今回の機能には無く、今後の開発課題である。

2) 事例ベース監視システム (事例ベース監視エージェント)

異常経験をデータベース化し、運転状況を常にデータベースの事例と照合し、異常があれば事前に察知し、その異常が過去の事例と同じ状況であれば、その原因や過去の対策を提示する。このような異常パターン認識による診断方法について、かなり有用とされている。監視用事例データの管理登録は容易とは思われているが、実プラントで事例となる状況を全て作り出すのは難しいとの意見が出た。

一度登録した異常を、シミュレータで繰り返した場合の、登録した異常原因、過去の対応操作の情報提示は妥当と判断されているが、情報の整理、編集など運用が大事との指摘があった。

3) 仮想プラント表示システム (仮想プラントエージェント)

仮想プラントでは、プラントの状態を直感的に把握するため、また異常発生個所を間違いなく把握するため、プラントを3次元仮想モデル化して表示している。最近3次元グラフィックの活用が進んでいるためか、今回のモデルの描写は、実プラントほどの臨場感は与えていない。(尚、この後かなりの改善が図られている)。異常発生個所や異常内容は分かりやすいとの評価を得ながら、必要性については否定的な意見が多い。異常時には、操作すべき計器が自動的に表示され、セット値の変更などの運転操作が可能となっているが、操作はDCSと競合しており、運転操作の機能は不要との意見が多い。ただし、少数ではあるが、DCSなしでの運転希望もある。

仮想プラントの画面の中にITV(現場設置テレビ)の画面を重複して表示することが可能である。機能としては、便利で有効との評価ではあるが、システムの負荷増大、ITVの台数の限度から、必要性には疑問が多い。

4) モデルベース推論システム(意味表示インタフェースエージェント)

異常を検知後、プラントモデルを使って、異常をそのままにしていたらどうなるか、近い将来を予測する機能がある。現実の運転では、そのような悠長な暇はないという意見もあったが、概して有効の評価を得た。異常の対象機器だけではなく、関連機器のデータ、あるいは、DCSや実プラントで見ることが出来ない、トレーやダウンカマー等塔内部の予測表示、製品品質の予測などの要望があった。

また、モデルベースでの異常原因推定は、機器の繋がり、機能の繋がりから成り立っており、従来のエキスパートシステムとは異なった機能である。このシステムの機能への興味は高かったが、デモがスベアポンプの無いポンプの性能低下であったので、分かりやすい例題ととられ、もっと複雑な例題でのデモの要望があった。実施すべき対応操作の導出を含め、モデルの妥当性の検証方法をどうするか問題の提起があった。また、あれば便利だが、運転員が考えなくなることを心配する意見もあった。

5) 大画面統合表示システム(VR含み)

このシステムでは、各サブシステムが作成する全てのデータ、異常通知、異常原因推定、対応操作候補その他必要とする情報を自動的にあるいは運転員の要求により、大画面に表示する。大画面では画面を分割し、新しいデータは大きく、古いデータは小さくなるが、運転員の操作で入れ替えは可能である。製油所の運転員や技術者の要求から「Operator/Co-operatorモデル」のコンセプトにより設計されている。表示するデータの内容、画面の展開と入れ替え、画面割り当てとサイズ、操作性など良好な評価を得ている。ただし、既存の運

転操作端末 DCS との使い分けが明確ではなく、個々の機能では評価が高いが、使い方に疑問が残る。画面上で、人間などを模擬した絵や写真などで応対する擬人化については、今回はロボットの絵であったが、あまり良い評価が得られなかった。擬人化そのものの必要性が感じられていない。温度、圧力、流量等の運転データをグラフィックで可視化の検討を行ってきたが、もともと可視化のイメージがなく、また必要性も少なく評価が低い。

6) 音声コマンド

このシステムでは、音声によりシステムとアクセスすることが可能である。言葉の使い方、語彙のフレキシビリティは妥当であり、音声そのものも聞きやすい。また応答のスピード、タイミングも適切との評価である。しかし、実用を想定すると、声や音が入り乱れる計器室で問題なく使えるのかとの疑問が残っている。更には多数の音声重なった時の遅れなどにも不安が残る。

7) オントロジー（メッセージ生成システム）の作成言語

異常を検知したり、異常原因を推定したり、各サブシステムが情報を表示したい場合、オントロジーに表示内容の言葉の作成を依頼し、言葉を受け取り表示する。今回のデモでの表示内容は、妥当だとの評価を得た。

8) システムの保守運用に対する意見

実用化を想定すると、システムの保守運用は大きな問題である。先に述べた事例データベースとの参照には、異常データを多く貯えれば、効果は更になる。また、プラントの改造に伴うシステムの修正が容易か否かもユーザーにとっては大きな問題である。

事例データベースは、異常と認定し、原因が一つに絞られれば、自動的に登録できる。また、事後修正の場合は、データを呼び出し、追加修正ができる。これらの操作は、ほぼ容易だと評価された。オントロジーと意味表示インターフェースのプラントモデルの保守は、システム構築、運用を容易にするためエディター化されていて、追加修正は容易に出来るよう配慮されているが、プラントモデルの保守は、プラントを機器の繋がり、機能の繋がりに分けたモデルとなっており、石油業界に馴染みがなく、難しそうだと評価を受けた。実際に操作してみないと分からない、作業方法の問題ではなく、作業量の膨大さ、検証方法が問題との指摘がある。

その他の意見

異常発生は、定常運転より、装置の立ち上げや停止時に多い。そのような時

に使用できればすばらしい。

実用化にはまだと言った感があるが、省力化、オペレータの経験不足を補うには価値あるものに思える。

運転員のレベルは想定しているより高い。不必要な機能が多い。

トラブル原因を勉強させるための教育用であれば有効である。

事例を蓄積してのガイダンス機能など、部分的に実用化できそうなものもある。

2.5.5.2. 日本能率協会ユーザーアンケート 結果

アンケートに回答した企業の業種は、電力供給、ガス供給、エンジニアリング、計装メーカー、ファイン化学、セメント、醸造、システム開発等、多種にわたる。回答状況は次のとおりである。

1) プラントデータ監視システム（DCSエージェント）

データ監視機能、異常検知タイミングはほぼ妥当との評価である。ノイズの影響に対する疑問があった。ノイズとは、操業運転時時の計測データの定常的、且つ正常な外乱変動のことである。今回はシミュレーションベースなので、ノイズはない。実用化に対しては、ノイズに対するロバスト性が重要な問題になる。

2) 事例ベース監視システム（事例ベース監視エージェント）

異常パターン認識による異常診断は有用と判断されている。しかし、実プラントの異常は頻繁ではなく、異常パターンの登録を増加させることは難しいとの意見や、全タグを監視するのは今回のコンピュータでは能力上困難であり、異常を検知するタグのグルーピングの決め方についての疑問があった。今回は、全タグ130個のうち、7個のタグを常時監視して、異常を検知している。9種のシナリオについては監視可能のようであるが、もっと多くの異常を感知できるか検証する事が必要である。

3) 仮想プラント表示システム（仮想プラントエージェント）

石油連盟説明時に比較し改善がなされ見易くなっていて、評価が高い。異常発生個所や異常内容は分かり易いが、プラント改造時の保守、複数異常時の対応等への疑問がある。ITVとの併用は概ね便利で有効と見られている。

4) モデルベース推論システム（意味表示インタフェースエージェント）

モデルベース推論への評価は高い。従来のエキスパートシステムの後継とし

での期待が大きいものと思われる。特に、プラントのリミットに達する前に、早急に実施すべき対応処置を指示する機能への期待は大きい。ただし、モデル化が難しいこと、モデル化の条件から外れた場合の精度への疑問があった。

5) 大画面統合表示システム (VR 含み)

表示データの内容、表示レイアウト、画面展開など良い評価を得ている。ただ、擬人化については、あまり評価が良くない。必要性そのものから疑問がある。DCSと別な情報表示として要望は強いが、DCSの機能と重複するところもあり、DCSとの整合性を明確にする必要がある。

6) 音声コマンド

音声の言葉使い、入力時の認識の汎用性は妥当と評価されているが、デモ時若干応答がスムーズでない時があり、聞き取りづらいことがあったようである。音声コマンドでのタグ呼び出しより、DCSのタッチスクリーンの方が速いとか計器室内は種々の声や音が入り乱れるため、この騒音の問題を指摘する意見があった。

7) オントロジー (メッセージ生成システム) の作成言語

全体に適切と見られている。ちゃんとした日本語になっているが、異常時には迅速性が求められるため単語の羅列の方が良いとの意見があった。

8) システムの保守運用に対する意見

今回はオントロジーの構築、保守のデモは都合により実施できず、意見はない。意味表示インターフェースは前回石油連盟への説明時より評価は良くなったが、それでも、見慣れぬためか、難しいと見られている。実プラントは常に改造、増設が行われているため、メンテナンスが追いつかないことを心配する意見がでている。ただ、簡単と評価している人もおり、システム関連の人にとっては、特に難しい事も無いのかもしれない。

9) その他の意見

ベテランオペレータの技術伝承が難しくなっている現状では、技術継承の観点から有効なシステムである。

DCS 等他の制御装置との整合性を明確にすべきである。

DCS は完成に近づいており、ヒューマンインターフェースは今後の進歩が見込まれる分野である。今後の進歩を期待している。

異常検知では、狼少年にならないこと (計測データのノイズ等における異常

と正常の確実な判断)、異常時に多数の警報がでるがオペレータがパニックにならないよう不要な警報の抑制が必要である。

異常時対応のみではなく、日常の効率運転にも有効な支援ができるようにして欲しい。

モデル化を容易にできるようにして欲しい。例えば、P&ID からの自動作成教育用ツールとしても効果が期待できる。

2.5.5.3. ユーザーとしての評価（実用化に対する考察）

次に本システムの実用化の可能性についてユーザーの立場から評価を行う。

5 年間プロジェクトの 3 年目から参加したが、分散処理で複数のサブシステムに別れていること、サブシステムの中には石油精製業のメンバーには理解しづらい内容もあり、ここでは、理解しづらい当初計画に対する完成度というよりは、ユーザーの立場から実用化が期待できるか、どのような問題があり、どのような改善が必要か、を検討した。

評価のベースとしては、石油連盟および日本能率協会への説明会時のアンケート調査結果も参考にした。

参加した当初は、このプロジェクトがヒューマンメディアと言う語感が示すように、画像を中心とした人間の直感、感性に訴える表示技術の開発がメインであった。しかしプラントにとっては表示技術と共に、表示するデータの内容が大きな比重を占める。運転員にとっては、必要なデータは正確でなければならない。間違ったデータ、あるいは不必要なデータを如何に美しく、見易く表示しても、ユーザーにとっては何の価値も無く、むしろプラント運転に混乱を与えるだけで、逆に安全上危険でもある。そこで、ユーザー評価は、このプロジェクトの当初の目的からは若干見方を変え、ユーザーに必要かつ信頼できる情報が作成され提供されているかと言う観点を折込んだ評価とした。

各要素技術の分散強調によるシステム結合テストの結果は、今年のプレ最終評価以降格段の進歩が見られた。シナリオという限定された条件下ではあるが、各要素技術ともほぼ当初の計画仕様通りの動作結果が得られ、システムを結合したプロトタイプモデルの開発という面では計画通りの結果が得られた。しかしながら、10 年間のプロジェクト期間が半減したこともあり、今回のシステムはまだ実用品として完成したものではなく、分散協調システムとしてサブシステム間の通信が可能となったところで、幾つかの異常シナリオでの作動を確認してはいるが、実用化に向けて取り組まなければならない点も多く残されている。またシステム検証内容も、想定された条件すべて網羅したものではない。例えば、シミュレータベースでも、長時間連続で作動させた場合の信頼性や、複合的にトラブルが発生した場合の処理能力、更にデータにノイズが載ってい

るような実プロセスの諸問題の処理など、検証していかなければならない。

以上のように、直ぐに実用化の評価を下す段階ではないが、そのような前提で、実用化に向けて、次の5項目に分け評価した。

- (1) システム構築における必要なユーザー提供資料
- (2) 開発期間、開発コスト、システム導入における問題点
- (3) 全体システムの有効性、効果
- (4) 個別システムの有効性、効果
- (5) システムの保守、改造の容易さ

(3) と (4) の全体及び個別システムの有効性、効果が主であるが、他の3項目にも若干の考察を行った。

1) システム構築における必要なユーザー提供資料

実際運転しているプラントに適応するシステムを構築するためには

- ①構造を含む機器データ
- ②機器の機能の説明
- ③実装置の写真や絵
- ④運転の目的、方法、操作の説明（運転マニュアル）
- ⑤運転データ（プロセスデータ）
- ⑥簡易プラントシミュレータモデル作成のための計算式

等をユーザーとして提供する必要がある。1項から5項までは、実存のプラントには、ほぼ完備されているか、直ぐにでも準備することは容易である。しかし、6項の簡易プラントモデル作成用の計算式提供には多少困難を伴う。

今回は、トッパーの主精留塔の計算式は精留塔廻りの温度、流量などの運転データから熱収支、物質収支をベースにした計算式で、部分部分に経験式を織り交ぜ、この装置の大まかな挙動を推測する事とした。

簡易シミュレータ導入の目的は

- ・ 導出された対応操作における最適な操作量の決定
- ・ 操作による副作用の事前検討

を高速で計算し、運転員への支援情報とすることにある。高速と精度にはトレードオフの関係にあり、高精度はそれほど必要とはしないが、種々の状況に応じた計算結果の検証が難しい。さらに、原料となる原油の性状は運転中は正

確には把握できず、原料タンク切替えのたびに大きく変動する可能性があり、更に検証が難しくなると考えられる。

同じトッパーでも、ひとつとして同じものは無く、装置ごとにシミュレータを作成する必要がある。意味表示インターフェースでは塔や熱交換器などユニット化を計っており、今後はプラントモデル構築の迅速化が期待できる。石油精製の他の装置には、トッパーと全く異なる装置も多い。特に、触媒を使用している、接触分解装置、接触改質装置、脱硫装置、流動接触分解装置など、反応が絡む装置は、簡易モデル作成がトッパーよりかなり困難と思われる。更にライセンス装置によっては、反応部分はブラックボックスとなっており、モデル作成がより困難となろう。

2) システム導入の期間、コスト、体制等の問題点

テスト的に実プラントに導入する場合も、実用のための導入も、投資判断は異なるが、導入作業の考え方は同じである。

先ず、システムを設置する場所の問題がある。現在のシステムは分散協調のシステムとなっており、8基のパソコンが並び、かなりのスペースを必要とする。運転員が装置を運転する計器室には、まず、スペースがあるとは思えないし、まして多数の作業員が中でインストール作業する事は出来ない。今回の開発では各研究室が個々にコンピュータを使用しているが、二つ以上のシステムを同一計算機上で稼動させることにより、コンピュータ基数を少なくする必要がある。このハードウェアの統合は特に問題なく実施可能であろう。

現在の到達レベルを踏まえて、各要素技術から提示された、システム開発期間とコストは次のとおりである。

インターフェースエージェント	数千万、慣し運転含め約1.5年
意味表示インターフェース	モデル化に約半年
オントロジー	言語データベース作成に約3ヶ月
仮想プラント	3000万、作成に約半年
大画面	1000万、約半年
分散協調システム	不明

個々のサブシステムではあまり明確ではないが、プロジェクトでは総コストを概略1億円程度と見ている。開発期間は1年程度であろう。

導入の投資判断については難しい問題である。異常はそれほど頻繁に起きるものではなく、まして現在でも事故に至るトラブルはそれほど多くない。しか

し、一度でも事故を未然に防ぐことができれば、投資の効果は十分である。安全のための保険的な意味合いが強くなるが、この保険をどう判断するかである。またアンケートでの意見にもあるように、日常の運転の効率化・最適化に寄与するような適用業務を開発できれば、運転人員の削減や経験の少ないオペレータの採用等の副次的効果も考えられ、導入判断がより容易になる。

実用化以前の実プラントでの実機テストを実施する場合も同じであるが、実システム開発に向けては、開発期間に関して、運転現場での作業を極力短くする必要がある。これは、各要素技術の技術的な開発だけでなく、分散協調システムにおける開発体制にも絡むが、各要素技術が離れた場所で個別システムになるのか、またどこかで集まって開発するのか、開発方法、責任分担、スケジュールを明確にすることを念頭に置かなければならない。

3) 全体システムの有効性・効果

このシステムの働きは、装置の異常を検知し、異常内容をオペレータに通知すると共に、異常原因と異常対策を分かりやすく、見易い形で提供することにある。シナリオ想定範囲内で、この一連の働きは目標通り達成されている。要素技術によってはかなり完成されたものもあるが、全体システムとしては、当初のプロジェクト期間の半ばであり、分散協調の仕組が完成したばかりで、実プラントにおける全ての運転モードに対応した作り込みとはなっていない。更に2つ以上の異常同時発生時は多数の警報が発生するが、システムが適切に情報を制御できるか否かの検討は、次のステップの検討として、時間的制約のため、最初から今回の検討対象から外れている。未検討事項を検討し、また、石油連盟および日本能率協会の各ユーザーからの指摘事項について一つ一つ検証を行い、実用化に向けて完成度を高める必要がある。

従って、直ぐ実用化がどうかとの評価はむずかしいが、石油産業ユーザーの視点から見て、興味深いシステムである。危険物を大量に取り扱う装置産業では、安全性の向上は最優先の課題であり、各企業で人工知能、エキスパートシステム等研究が進められていたがあまり実用には結びついていない。このシステムには、異常事例データベースとの照合、プラントモデルを使った異常原因の推定、異常対策指示といわゆる経験ベースと理論ベースの両方から診断する今までに無い、新たな手法を使っており、実用化の期待は大きい。

実用化対象として「異常診断システム」への適用を重視する場合、システムを判断する基準としては、高度な完成度が求められる。少なくともシミュレータベースでも長期の連続運転を実施し、システムがダウンすること無く稼動し、いろいろな変動、異常に対し、表示結果が妥当なことを確認する必要がある。現状ではシナリオ外の異常を入力したり、異常の程度を変えると、音声出力が

多くなった時、音声出力が溜まり、画面と同期が取れなかったり、通信のエラーが発生したり、間違った表示を行ったりすることが時々発生し、もっと完成度を高める必要がある。長期連続運転による性能チェックが実施されていない現状から考えると、実用化の判断を行うにはシステムの完成度向上と機能チェックに後1～2年程度の時間が必要であろう。

一般企業の投資は、得られるメリットから判断される。このシステムの目的が、常時利益を生み出すものではなく、発生するかもしれない異常の検知が主体である。事故防止は最優先であり、一度でも実際の事故防止に役立てば、投資の効果は評価できる。しかし、起きるかもしれない異常に対する投資はいわば保険ということになる。従って実用を考慮するなら、必要不可欠な機能を残し、必要性が少ない機能は除去し、コスト削減も念頭にいれるべきである。

今回のプロジェクトでは、ベテランオペレータ用のシステムではなく、比較的経験の少ないオペレータを対象としたためか、未経験者対象に考えられる機能を全て組み込んできた。DCSとの機能分担、対象となるオペレータの技術レベルを明確にしていけば、コスト削減は可能と思われる。

その他に、石油連盟や日本能率協会のアンケートでは、通常運転ではそれほどトラブルは起きず、トラブルが多い非定常運転即ちシャットダウンやスタートアップの異常監視に使用できれば更に有効であり、またトラブルの原因究明の教育にも有効であるとの意見がある。

近年、オペレータ数の減少から、短期間での新人教育の必要性、ベテランオペレータのノウハウや技術伝承が企業の大きな問題になっているが、この問題への対策として期待は大きい。

4) 各要素技術の有効性・効果

① インターフェースエージェント

運転データの上下限、変化率、単調変化等のチェックはDCSと重複する機能もあるが、閾値をDCSより狭くして、早めに診断に入ることも可能となり不都合はない。

幾つかのデータの組み合わせで異常を検知する機能は興味深い。このシステムでは、130個の計測データの中から7個を選出しこの組み合わせで異常を検知する。今回作成した9個のシナリオはこの7個の変数の組み合わせで検知可能であるが、もっと多様な異常に対応するには、もっと多くの変数チェックが必要であるが、その数は見極めがつかない。全変数をチェックできれば問題ないが、数に制限がある場合、適切な変数の選択は難しい問題である。今回はトライアンドエラーで変数を決定したが、変数選択を自動化する事が好ましい。

変数7個の制限は、2秒ごとのチェックを実施するコンピュータ処理スピード

のハードの問題であり、2秒周期を更に長くしたり、ハードウェアの増強で多少の増加はそれほど問題ではない。ただし、実プラントの変数の数はもっと多く、チェックすべき変数の概略数の見極めは必要である。この組み合わせが、異常事例のデータになる。日常の運転でそれほど頻繁には発生しない異常事例を数多く登録するには時間がかかるが、システム構築時の複雑なプログラムを組む必要が無いメリットがある。ただし、多くの種類の原油を処理するトッパーでは、原油の種類によって運転条件が変わるため、システム立ち上げ時にはそれらの変化が異常ではないことを登録する必要がある。

変数の変化の事例では、今回はシミュレータの計算結果で運転データにノイズがない。従来から指摘しているが、実プラントではデータにノイズがある。ノイズの影響を最小限にする対策を考慮する必要がある。今後の実用化研究が期待される。

音声認識は特に問題ないが、音声入力はオントロジーとは別になっている。開発期間と費用の問題で、今回は統合が図れなかったが、音声入力もオントロジーの語彙作成能力と統合できれば汎用性は大きくなると思われる。今後の課題である。

最後に問題となるのは、検証である。定周期でチェックする変数と変数の変化をチェックする閾値が十分かどうか、このシステムがこの装置のどの異常も検知できるかどうかの確証が必要である。即ち、ユーザーが検収を判断する基準を明確にする必要がある。

② オントロジー

オントロジーのプラント関連の日本語作成機能は問題なく適切な日本語を作成提供している。同じ機器でも、異常の内容で異なる表現をすることがあるが、その使い分けも適切である。プログラムもパッケージ化されており、システム構築、プラント改造などに伴う保守も容易でユーザーサイドでも受け入れ易い。

オントロジーでは機器、配管の接続情報を管理している。装置産業には、配管が縦横に入り組み、特に、出荷設備やタンク間の移送配管は毎回使用するルートが異なり、そのたびにその時点での最適なルートを選択している。このような分野に適用を検討することも有意義なことと思われる。

③ 意味表示インターフェース

ヒューマンメディアの題目である、「運転員の視点を考慮した理解し易い情報提示」もプラント運転の安全に関わる場合、情報に正確さが要求される。ユーザーとしてはその正確さへの期待が大きい。

インターフェースエージェントから異常を検知した計器データを受け、プラ

ントモデルを使って、異常原因候補を選定し、原因候補の妥当性をチェックし、妥当性のある原因候補を推定する。更に、異常対応操作を提示する。操作指示の場合、運転マニュアルなども表示すればさらに効果的となる。異常時にオペレータが必要とする一連の情報が提供され有効である。

今まで検討されていた、エキスパートシステム、人工知能とは異なり、プラントモデルをベースにしたシステムで、異常検知におけるエキスパートシステムがそれほど実用化されなかったため、新たな手法として興味深いシステムである。エキスパートシステムでは、ベテランの経験、知識を前もって入力する必要があった。意味表示インターフェースでは、経験知識の入力の代わりに、プラントモデルを構築する必要がある。

プラントモデル構築には、MFM (Multi-level Flow Modeling) で作成する機能一目標レイヤーを示すモデルとプラント状態予測のための高速シミュレータモデルがある。ユーザーとしては、高速シミュレータモデルのための、プラント挙動を表わす計算式を提供する。相変化を伴わない熱交換器やポンプなどは簡単であるが、相変化がある機器、特に側線が多いトッパーの計算式はかなり困難であった。一度作成すれば、別なトッパーでも 2 個目からはずっと容易にはなるであろう。しかし、石油精製での装置は非常に多く、接触改質装置、接触分解装置、水添分解装置等触媒を使ったプラントも多い。複雑でライセンス装置はモデル作成にライセンサーの協力も必要となる。

モデル作成に労力はかかるが、ユーザーにとっては魅力的な技術である。現在は幾つかのシナリオに対応できる程度であるが、もっと完成度を高めて多くのケースを実装して評価する必要がある。

④ 仮想プラント

プラントの 3 次元のグラフィックは、最近では一般化してきており、臨場感に対する要求は高い。異常発生時、仮想プラント上で該当機器が色が変わりブリンクするなどして異常発生箇所、異常発生内容は仮想プラントで見ると解り易い。定常運転時も VR 上のセンサーをスキャンし、センサーデータを可視化して表示している。可視化には、温度、流量、圧力などを、色や気泡などの集約度などで表示している。次々に表示されるデータが基準より高いか低いかは瞬間的に雰囲気判断できる。温度、圧力、流量など、色や泡粒などで可視化を検討したが、もともと可視化のイメージが無く、あまり有効とは思えない。ただし、代替え案も浮かばない。

大画面の自律化の一環で、操作が必要なタグを仮想プラントに自動表示し、セット値の入力変更ができるようになっている。必要なタグの操作画面が自動表示されることもあるが、便利だとの評価と、DCS との整合性から混乱する意

見に別れる。機器の操作はプラント運転への影響も大きく、DCS のみにする方が混乱がなくよいと思われるが、各企業での DCS との機能分離、整合性の判断による。

VR 画面に ITV の併用表示の機能については、機能は有効だが運用に問題がある。ITV の現状は、プラントに複数の ITV と CRT が設置されている。センサーでは計れない加熱炉の炎や、フレアスタック、あるいは集中的に監視が必要とされる場所や機器を常時写している。ITV が 1 個のプラントでは、VR 画面上に ITV を 1 画面のみ表示しても効果はあるかも知れないが、多数の ITV があるプラントではやはり CRT の設置は必要である。VR 上の ITV 表示はプラントの規模、ITV の数、VR のシステム負荷から総合的に判断する必要がある。

VR 画面は経験の少ない運転員には、有効ではある。またあれば、便利であるが、必要性から検討する必要がある。実際、アンケートでも、VR は必要がなく、2 次元のフロー図表示でも問題ないという意見もある。ITV 機能、セット値入力機能など、プラントの規模、対象とするオペレータの技術レベルから総合的に判断し、不要な機能を削り、コストを低減する検討も必要と思われる。

当初設備データとの融合なども提案したが、本研究の主旨と異なるため、開発から外している。個別技術として応用技術を模索する事が期待される。

⑤ 大画面

大画面のデータ表示は、異常通知、現在状態提示、近未来状態表示などからオペレータによる対応操作表示、異常リセットの順に表示され、通知方法内容とも問題はない。基本的には、最新情報が大きく表示され、古くなると小さくなるが、オペレータの操作で再度大きく表示することも可能である。

画面の大きさも可変のケースと固定のケースが選択でき、オペレータの好みに合わせることができる。

画面の統合と分散配置の考え方は、従来からプラントオペレーション技術の中で議論されてきた。今回は百花総論的に可能な情報全てに対して対応できる事を証明したが、実用化にあたっては、既設 DCS との機能分担についてさらに深い議論が必要と思われる。

今回擬人化が検討された。擬人化は、銀行端末を始め種々のマンマシンインターフェース使用されており、今後のトレンドであるとの事である。大画面の片隅に、ロボットや事務服の女性の絵や写真があり、そこから棒で問題のポイントを指し示す。問題となっているデータの判別は分かり易いが、色を変えたり、ブリンクさせたりしても判別はできる。若い人には抵抗はないとの意見もあるが、プラント運転に相応しいか疑問がある。少なくとも、今の擬人化はあまり適切だとは思われない。

表示内容が多数連続で入力された時、音声と画面にずれが発生することがある。

機能は問題ないが、現状の運転はDCSで行われており、運転員はDCSから離れることは出来ない。この使い分けを明確にする必要がある。

⑥ 分散協調システム

マルチタスクで各サブシステムを分散的に管理するシステムとして、プロトタイプとしては十分な機能を発揮した。ただし、実プラントで連鎖的なトラブルが発生し、多数の異常状態を想定した場合の処理能力が十分か検証する必要がある。

また、物理的な問題として、少なくともコンピュータの数は減す必要がある。現状の計器室には、パソコンを7～8台も設置できるスペースはない。せいぜい1～2台にする必要がある。また、装置の改造などに伴うシステムの修正がそれぞれの要素技術を個別に修正するのではなく、一つのシステムを修正すれば他のシステムも自動的に修正できるようシステムが好ましい。ユーザーから見れば、システムは一つであり、システムの導入や保守での責任窓口は一つにして欲しいところである。要素技術間で責任分担が混乱する事がないような、体制、分担が明確にならなければならない。

2.6. プラントオントロジーの研究

2.6.1. オントロジーサーバー

オントロジーサーバーは、その中でもオントロジーの共有・再利用を支える核となるシステムである。これまでの報告書でも部分的な機能や実現の概要を述べてきたが、本節ではオントロジーサーバーの最終版について述べる。

オントロジーサーバーの中心的な機能はオントロジーとそれに基づいて作成されたモデルを管理する機能である。主な機能はオントロジー、モデルの整合性を検証しながら構築し、保管しておくことである。そのためにオントロジー、モデルの構造、それらに対する操作を明らかにする。最後にオントロジーサーバー全体の開発について述べる。

オントロジーサーバーは、オントロジーの内部表現に関する枠組みを与える。オントロジーの内容を表現しようとしたとき、オントロジーエディタで図的に表示されているような構造的な部分と、図的表現が向かない線形的な部分がでてくる。本節では、線形的な部分を単に公理と呼ぶ。公理には特定の概念定義に含まれるクラス内公理と「概念」とは独立して概念化されるクラス間公理がある。どちらも表現の枠組みは同様である。

オントロジーは概念と概念間の関係から成る。さらに概念は、全体概念、関係概念の2つに分かれ、基礎論に基づいた深い議論があるが、ここではオントロジーサーバー上で扱う枠組みだけを扱う。公理についても同様に、オントロジーサーバーにおける取り扱いについて考察する。

次に、それらの概念を扱う際に必要な、定義、編集などに関する操作を考える。またオントロジーサーバーが行うオントロジー、モデルに対する整合性検証について説明する。

2.6.1.1. オントロジーの構成

2.6.1.1.1. 全体概念

全体概念とは「あるもの」をその一部分をなす複数の概念から構成される「全体」としてとらえ概念化したものである。例えば、「自転車は、車輪とフレームからなる」といったときの「自転車」という概念にあたる。その構成は下記のようになっている。

- ・ 概念名
 - ・ 概念の呼び名。
- ・ 上位概念
 - ・ このクラスの直接の上位概念名。概念は、その上位概念が規定する制約をすべて満たさなければならない。(上位概念で規定する制約

を継承する。)

- ・ クラス内公理

- ・ このクラスのスロット値に関する制約の線形表現。表現方法の枠組みについては次節で述べる。

- ・ スロットの集合

- ・ スロット名

スロットのロール名。

- ・ 全体概念との関係 (part-of / attribute-of)

概念とスロットとの関係。part-ofかattribute-ofを取る。

- ・ クラス制約

スロットの値となるインスタンスが所属するクラスに関する制約。

- ・ 数の制約 (+ / * / n / n...m) ※n, mは正数

スロットに入る値の数を制約する。

*: 0個以上の値が入らなければならない

+: 1個以上の値が入らなければならない

n: n個の値が入らなければならない

n...m: n~m個の値が入らなければならない

2.6.1.1.2. 関係概念

関係概念とは複数の概念の概念間に成り立つ「関係」を概念化したものをいう。例えば「AさんとBさんは夫婦である」といったときの二人の間の「関係」を概念化したものが「夫婦関係」という関係概念である。その構成は全体概念とほぼ同じだが、スロット制約のうち「概念との関係」が、全体概念 participate-in、 attribute-of のいずれかを取る、という違いがある。

2.6.1.2. 公理の構成

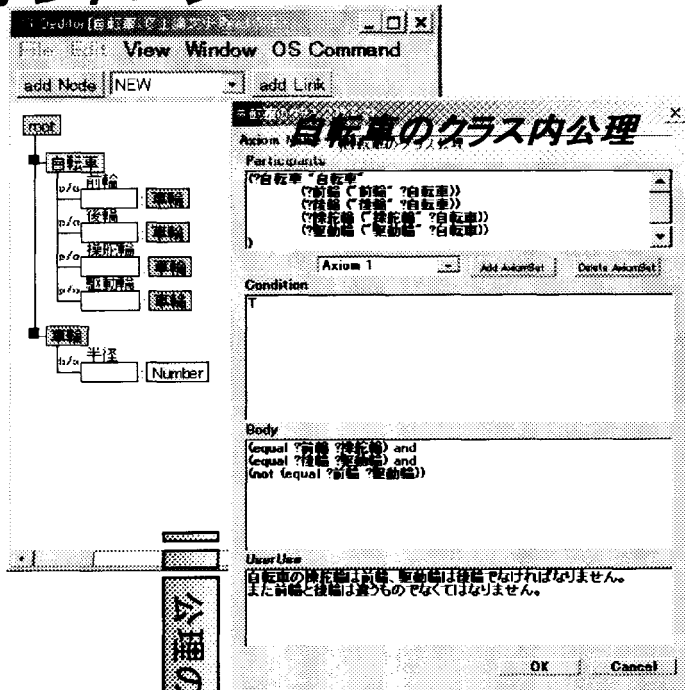
オントロジー定義の構造的な部分は、グラフィカルな編集手段が適しているが、そうでない部分は線形的に記述される。我々はその線形的に書かれる制約を単に公理と呼んでいる。オントロジーサーバーでは、特定の概念に付加的に記述されるクラス内公理と、それだけで概念化されているクラス間公理の2種類を扱う。

公理は大きく分けて、Participants と AxiomBody の2つの部分からなる。以下、まず公理系の変数定義と制約の書き方を説明し、次に Participants と AxiomBody の記述について述べる。

図 2.6-1に公理の定義と、適用のイメージを示す。自転車は、車輪である前

輪、後輪、操舵輪、駆動輪からなり、インスタンスを作ったとき、前輪と操舵輪に入る値、後輪と駆動輪に入る値は同じであり、前輪と後輪に同じ値が入ってはならないことが公理として記述されている。そして自転車#1 インスタンスを作ったとき公理が適用され、?自転車には、自転車#1 が、?前輪、?後輪、?操舵輪、?駆動輪には、それぞれ車輪#1、車輪#2、車輪#3、車輪#2がバインドされている。そしてこれは、前輪と操舵輪が同じでなければならないという制約を違反しているため、「自転車の操舵輪は前輪、駆動輪は後輪でなければなりません。また前輪と後輪は違うものでなくてはなりません。」というメッセージが表示されている。

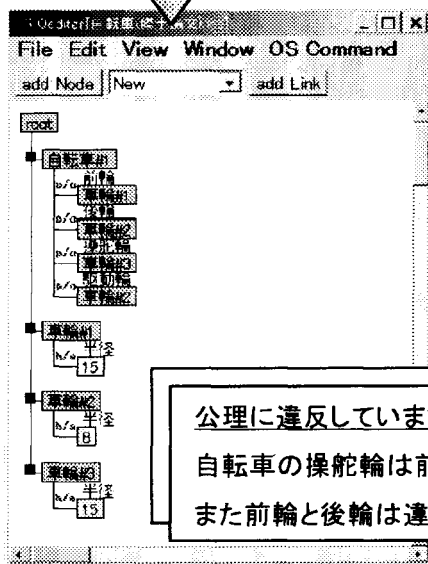
オントロジー



(1)Participants

(2)AxiomBody

モデル



インスタンス変数へのバインド

?自転車 ← 自転車#1

スロット変数へのバインド

?前輪←車輪#1

?後輪←車輪#2

?駆動輪←車輪#3

?操舵輪←車輪#2

公理に違反しています

自転車の操舵輪は前輪、駆動輪は後輪でなければなりません。

また前輪と後輪は違うものでなくてはなりません。

図 2.6-1公理記述の例

2.6.1.2.1. Participants

Participants には、制約の定義に参加する概念のインスタンスを表す変数を記述する。そして Participants で定義された変数を用い制約の実体を AxiomBody に記述する。

公理適用時に、インスタンスの値がバインドされる変数のことをインスタンス変数、インスタンス内のスロットの値がバインドされる変数のことをスロット変数と便宜上呼び分ける。図 2.6.1 のオントロジー中で、?自転車という名前で定義され、モデル中で自転車#1 がバインドされているのがインスタンス変数である。また同様にオントロジー中で?前輪、?後輪などの名前で定義され、モデル中で車輪#1、 車輪#2 などがバインドされているのがスロット変数である。

● 変数の定義

記述形式：(インスタンス変数名 クラス名

(スロット変数名 (スロット名 インスタンス変数名)*)

変数名は、 ? から始まる文字列である。

公理を評価するとき、“クラス名”が示すクラスのインスタンスが“インスタンス変数名”が示す変数にバインドされ、その名前で Axiom Body から参照される。

インスタンス変数にバインドされるインスタンスはさらにスロット値を持つ。スロット名、インスタンス変数名を並べて書くことで、インスタンスの中の“スロット名” が示すスロットの値を表現し、それがスロット変数にバインドされる。

2.6.1.2.2. AxiomBody

このフィールドに制約の実体を記述する (図 2.6-1)。

Axiom Body フィールドはさらに 3 つに分かれセットで記述される。Condition は、制約をチェックするかどうかの条件で、これが成り立つときにそれに対応する Body が評価される。Body に書かれた制約が違反していたときは、エラーとともに、UserUse に書かれた内容がエージェントに返される。

・ Condition

Body を評価するかどうかの条件。論理式で表現される。

・ Body

Participants で定義した、参加変数が満たすべき具体的な制約。論理式で表現される。

・ UserUse

オントロジーサーバーを使うエージェントが自由に使うことができるフィールド。Body が評価された結果、制約を満たしていなかった場合、エラーとともにこの内容を返す。

- 変数の参照

記述形式：?(スロット名 インスタンス変数名)

インスタンス変数にバインドされるインスタンスはさらにスロット値を持つ。スロット名、インスタンス変数名を並べて書くことで、インスタンスの中の”スロット名” が示すスロットの値を参照する。

- プリミティブ

公理内の制約を記述するためのプリミティブは公理系を利用するオントロジー利用エージェントからの要求をベースに定義されている。プリミティブには、制約をかける述語(真偽値を返す)や、クラスやインスタンスを参照するもの、構造を制御するものがある。その一部を示す。

(1)真偽値を返すもの

- ・ include: 所属関係を表す。
- ・ equal: 等価関係を表す。
- ・ order-of: 順序関係を表す。

(2)クラスやインスタンスを参照するもの

- ・ get-parts: part-of 関係にあるスロットの値を取り出す

(3)構造を制御するもの

- ・ any-element-or:
集合に対して、指定された評価を行い、結果の論理和を返す。
- ・ any-element-and:
集合に対して、指定された評価を行い、結果の論理積を返す。

- 論理

以下のルールに従って複数の制約を接続できる。

制約 1 and 制約 2 : 制約 1 を満たし、且つ制約 2 を満たす。

制約 1 or 制約 2 : 制約 1、制約 2 の片方、または両方を満たす。

(not 制約 1) : 制約 1 を満たさない。

2.6.1.2.3. 整合性検証

概念や公理、モデルに対する整合性検証は 2 つのレベルで行われる。まず、

(1)パーサーにより定義や操作に関する関数の文法的なチェックが行われ、次に(2)内容的な矛盾がないかを検証する。(1)については、定義などがオントロジーサーバーに送られてきたときにただちに確認するが、(2)については OS クライアントからの要求があったときに行われる。

(1)については、一般のプログラムコンパイラのパーサーと基本的な動作は変わらないため、特に述べてない。(2)について、オントロジー、モデル、公理のそれぞれに対してどのような整合性検証が行われるか解説する。

(A)オントロジーに対する整合性検証

その概念の上位概念、クラス制約として参照しているクラスが存在するかどうかが確認する。

上位概念との間で、制約に矛盾がないか確認する。

is-a 階層がループしていないかどうかを確認する。

(B)モデルに対する整合性検証

スロットの値に対して、クラス制約、数の制約を満たしているか確認する。

(C)公理に対する整合性検証

公理に参加するインスタンスを指定することで検証を行う。公理の `AxiomBody` フィールドに書かれた制約を満たすかどうか確認する。

2.6.1.2.4. オントロジーとモデルに対する操作

オントロジーとモデルの構築は、「定義」、「編集」、「整合性の検証」の繰り返しで行われる。ここでは、オントロジー（全体概念、関係概念、クラス間公理の集合）、モデル（全体概念インスタンス、関係概念インスタンスの集合）に対してなされる操作を明らかにし、それぞれの操作に対応する関数を規定していく。

```

(define-base-concept “自転車”
  :super-class ( “乗り物” )
  :axiom (
    (Participants (
      (?自転車 “自転車”
        (?前輪 (“前輪” ?自転車))
        (?後輪 (“後輪” ?自転車))))))
    (AxiomBody (
      (
        (Condition (T))
        (Body (not (equal ?前輪 ?後輪)))
        (UserUse (制約違反です。))
      )))
  :slot-def (
    (
      :name “フレーム”
      :relation “part-of”
      :plural “1”
      :constraint “フレーム”
    )
    (
      :name “前輪”
      :relation “part-of”
      :plural “1”
      :constraint “車輪”
    )
    (
      :name “後輪”
      :relation “part-of”
      :plural “1”
      :constraint “車輪”
    )
  )
)

(define-base-relation “夫婦”
  :axiom nil
  :slot-def (
    (
      :name “夫”
      :relation “participate-in”
      :plural “1”
      :constraint “男性”
    )
    (
      :name “妻”
      :relation “participate-in”
      :plural “1”
      :constraint “女性”
    )
  )
)

(get-superclass クラス名 上位概念名)

```

図 2.6-2全体概念、関係概念定義

(1)全体概念に対する操作

全体概念は、クラス名、上位概念、クラス内公理、スロットから構成される。スロットの内容は、さらにスロット名、スロットと概念との関係、数の制約、クラス制約からなる。そこで概念定義のために以下の関数を与える。

概念定義

```

(define-base-concept クラス名
  :super-class (上位概念名)
  :axiom クラス内公理
  :slot-def (
    (
      :name スロット名
      :relation スロットと概念との関係

```

```

        :plural 数の制約
        :constraint クラス制約
    ) *
)
)

```

図 2.6-2の左の定義は、自転車の定義である。上位概念は「乗り物」であり、車輪クラスである前輪、後輪を一つずつ部分に持つ。そして、クラス内公理として、前輪と後輪が同じものであってはならないことが表されている。

(2)関係概念に対する操作

関係概念は、クラス名、上位概念、クラス内公理、スロットから構成される。スロットの内容は、さらにスロット名、スロットと概念との関係、数の制約、クラス制約からなる。そこで概念定義のために以下の関数を与える。

概念定義

```

(define-base-relation クラス名
  :super-class (上位概念名)
  :axiom クラス内公理
  :slot-def (
    (
      :name スロット名
      :relation スロットと概念との関係
      :plural 数の制約
      :constraint クラス制約
    ) *
  )
)

```

図 2.6-2の右の定義は、夫婦関係の定義である。夫婦概念は、参加概念として、男性である夫と女性である妻を一つずつ持つことが表されている。

(3)概念に対する共通の操作

それぞれの概念について、構成要素を編集する関数を規定する。構成要素には、上位概念、クラス内公理、スロットがある。

上位概念に対する操作

- ・上位概念の追加

(add-superclass クラス名 上位概念名)

- ・上位概念の削除

(remove-superclass クラス名 上位概念名)

- ・上位概念の検索(直接の上位概念のみ)

(get-superclass クラス名 上位概念名)

- ・上位概念の検索(すべての上位概念)

(get-superclasses クラス名 上位概念名)

クラス内公理に対する操作

- ・クラス内公理の編集

(exchange-class-axiom クラス名 クラス内公理)

スロットに対する操作

- ・スロットの追加

(add-slot クラス名 スロット情報)

- ・スロットの削除

(remove-slot クラス名 スロット情報)

- ・スロット情報の変更

(exchange-slot クラス名 スロット情報)

- ・スロット情報の検索

(get-slot クラス名 スロット名)

(get-parts クラス名)

概念そのものに対する操作

- ・概念に関する質問

(superclass? クラス名 1 クラス名 2)

(subclass? クラス名 1 クラス名 2)

- ・オントロジーの整合性検証

(check-ontology クラス名)

- ・オントロジー全体の整合性検証

(check-ontology)

(4) クラス間公理に関する操作

上に示したクラス内公理と同様に、クラス間公理は、参加概念を規定する Participants と、制約の実体である、AxiomBody からなる。AxiomBody はさらに、適用条件である Condition、制約の実態である Body、制約が違反されたときにエージェントに返される UserUse からなるので、以下のように定義のための関数を規定した。

- ・概念定義

(define-axiom クラス間公理名

(participants (

(インスタンス変数名 クラス制約)*

)

(axiom-body (

(

(Condition 適用条件)

(Body 制約の実体)

(UserUse ユーザー利用部)

```

        ) *
    )
)
・クラス間公理を編集
(exchange-axiom クラス間公理名 クラス間公理)

・クラス間公理を適用 (整合性検証)
(apply-axiom クラス間公理名
  (インスタンス変数名 インスタンス ID) *)
)

```

(4)概念インスタンス共通の操作

インスタンスを作成するとは、概念のスロットに値をいれていく作業である。よって以下のような関数を規定した。

インスタンス作成

```

(make-base-instance クラス名
  :id インスタンス ID
  (
    :name スロット名
    :value 値
  ) *
)

```

インスタンスを編集するとは、基本的にスロットの値を変更することである。またインスタンスが持つさまざまな情報を取り出すための関数もここで規定している。

スロットの値に対する操作

・スロットへの値の追加

```

(add-slot-value インスタンス ID スロット名 値)
(add-slot-value インスタンス ID スロット名 値1 値2)
(add-slot-value-by-order インスタンス ID スロット名 順番 値1)

```

・スロットから値を削除

```

(remove-slot-value インスタンス ID スロット名 値)
(remove-slot-value-by-order インスタンス ID スロット名 順番)

```

・スロットからの値の取り出し

```

(get-slot-value インスタンス ID スロット名)

```

インスタンスそのものに対する操作

・クラス名の取り出し

(get-classname インスタンス ID)
 (get-classnames インスタンス ID)
 ・インスタンス検索
 (get-instances 条件)
 ・インスタンスに関する質問
 (instance-of? インスタンス ID クラス名)
 ・インスタンスの整合性検証
 (check-model インスタンス ID)
 ・モデル全体の整合性検証
 (check-model)

これらを表にまとめたものが表 2.6-1である。

表 2.6-1オントロジーサーバー基本関数一覧

OS API	
(1)全体概念に対する操作	
・概念定義	define-base-concept
(2)関係概念に対する操作	
・概念定義	define-base-relation
(3)概念に対する共通の操作	
・親クラスを追加	add-superclass
・親クラスを削除	remove-superclass
・親クラスを検索(直接の親クラスのみ)	get-superclass
・親クラスを検索(すべての親クラス)	get-superclasses
・クラス内公理を編集	exchange-class-axiom
・スロットを追加	add-slot
・スロットを削除	remove-slot
・スロット情報を変更	exchange-slot
・スロット情報を検索(名前で検索)	get-slot
・親クラスかどうか調べる	superclass?
・子クラスかどうか調べる	subclass?
(4)クラス間公理に関する操作	
・概念定義	define-axiom
・クラス外公理を編集	exchange-axiom
(5)概念インスタンス共通の操作	
・インスタンス作成	make-base-instance
・クラス名を取り出す	get-classname
	get-classnames
・スロットに値を追加	add-slot-value
	add-slot-value
	add-slot-value-by-order
・スロットから値を削除	remove-slot-value
	remove-slot-value-by-order
・スロットから値を取り出す	get-slot-value
・インスタンス検索	get-instances
・インスタンスに関する質問	instance-of?

2.6.1.3. オントロジーサーバーの開発

以上の考察から、オントロジーサーバーの開発を行った。オントロジーサーバーについて、そのモジュール構成や、個々のモジュールの働きを概説する。

2.6.1.3.1. 機能モジュールの構成

オントロジーサーバーは、4つの機能モジュールから成る。Network Module は、通信関連の処理をする役割を持ち、外部エージェントからのコマンドを Command Parser に渡す。Command Parser は、Network Module から渡された外部からのコマンドを解釈し、必要なモジュールに渡す役割を持つ。また User/Group Manager は、ユーザー認証やユーザー・グループ情報の管理をする役割を持つ。Ontology Manager は、Ontology Base に蓄えられているオントロジー・モデルに関するすべての操作を扱う。

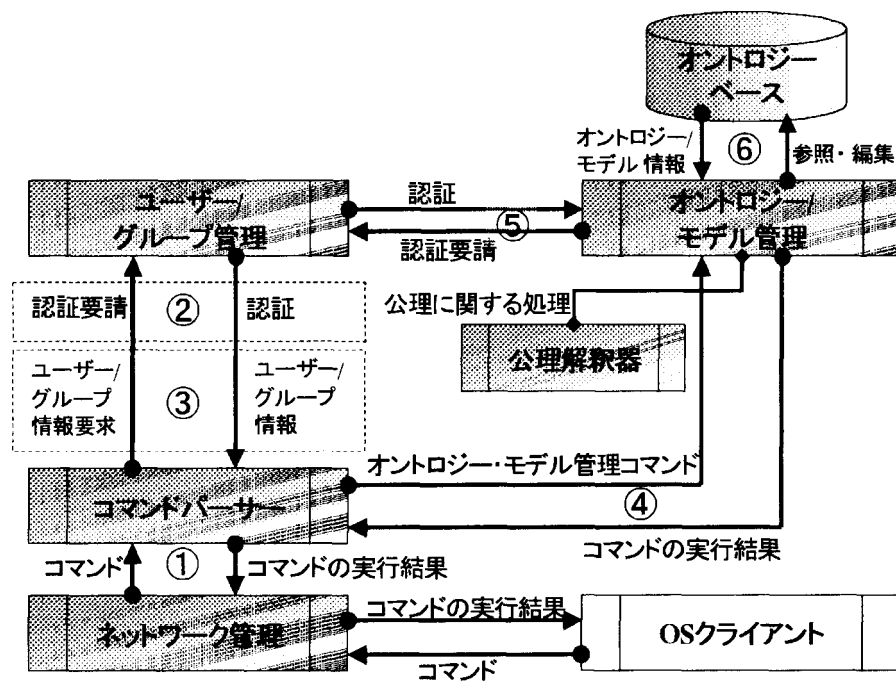


図 2.6-3 システムブロック図

各モジュールの働きを以下に示す。

- ネットワーク管理モジュール

OS クライアントとのネットワークを介した通信を司り、OS クライアントから

の要求はまずこのモジュールを通る。ネットワーク対応機能はこのモジュールで実現されている。受け取った内容は基本的にコマンドパーサーにそのまま流し、コマンドパーサーからは処理の結果の戻り値を受け取る(図 2.6-3 ①)。

- コマンドパーサーモジュール

ネットワーク管理モジュールから渡されたコマンドを解釈し、オントロジーへの操作関連コマンドであればオントロジー管理モジュールへ(図 2.6-3 ④)、ユーザー・グループ関連のコマンドであればユーザー/グループ管理モジュールに渡す(図 2.6-3 ③)。その際にコマンドの文法的な間違いなどがあればここで指摘する。また、コマンドを実行する前にまずユーザー/グループ管理モジュールから認証を受け取り(図 2.6-3 ②)、認証により実行権限を管理している。例えば新規ユーザーを作るコマンドを実行する権限は Administrator グループに属したユーザーしか行えず、勝手にアカウントを作られることはない。オントロジー管理モジュールからの戻り値やユーザー/グループ管理モジュールからのユーザー情報などは、ネットワーク管理モジュールを経由して、OS クライアントに渡る。

- ユーザー/グループ管理モジュール

中心的な機能は、ユーザー認証である。オントロジーサーバーにはいろいろな人が作ったオントロジーやモデルが管理されているので、それぞれのデータを適切に保護するため認証は重要な機能である。またユーザー・グループ情報の管理など、ユーザー/グループに関する情報や操作をすべて管理している。ユーザー/グループの追加・削除・編集や、アクセス可能なオントロジーの一覧作成などもこのモジュールが行う。

- オントロジー管理モジュール

オントロジーベースに蓄えられているオントロジー・モデルに関するすべての操作を扱う。オントロジーサーバーの核となるモジュールである。4.2.4。で整理したオントロジー、モデルを操作する関数は、すべてこのモジュールに属する。コマンドを実行した後、結果のエラー情報や、コマンドの戻り値をコマンドパーサーに返す(図 2.6-3 ⑤)。また、整合性検証もこのモジュールで行う。

- 公理解釈器

オントロジーのうち公理を解釈して内部状態に変換するモジュールである。また、実際に公理を適用する機能を持つ(公理に関する整合性検証)。オントロジー管理モジュールのサブモジュールで、オントロジー管理モジュールが公理に関する操作要求を受け取ると、このモジュールが呼ばれる。

2.6.2. プラントモデルの改善と構造検索

対象プラントのモデル化は昨年度までにほとんど終了していたが、12 年度に入って新たな機能追加があり、それに伴って昨年度まではモデル化されていなかったパイプをモデルに加えることと各センサの設置場所に関する再検討が行われたのでその結果について述べる。

2.6.2.1. パイプのモデル化

まず、モデル化する際の原則を述べる。

全てのモデル化対象はプラントオントロジーで定義されるクラスのインスタンスとしてモデル化される。

モデル化された部品を接続するために仮想的な「ポート」という概念を導入する。ポートはオントロジーにおいても定義されており、パイプを含むあらゆる部品の入出力に付随する。そして、部品間の接続はポートのインスタンス同士を接続することによって実現される。

プラントオントロジーに示されているように、対象は装置と対象物に分類される。対象物はプラント中を流れる原油などの流体である。装置のうち原油の位置だけを変えるパイプは導管とよばれ、その他の熱交換部品や蒸留塔などが装置である。装置と導管はその入出力口としてポートを持ち、それらの構造情報は隣り合うポートとポートの接続関係により表現される。(図 2.6-4(a))

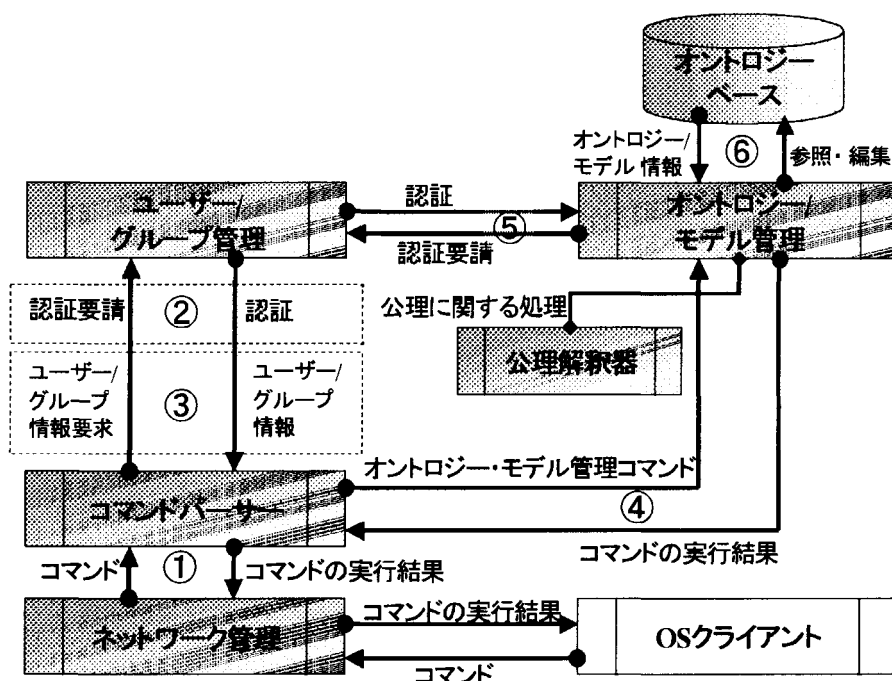


図 2.6.3 システムブロック図

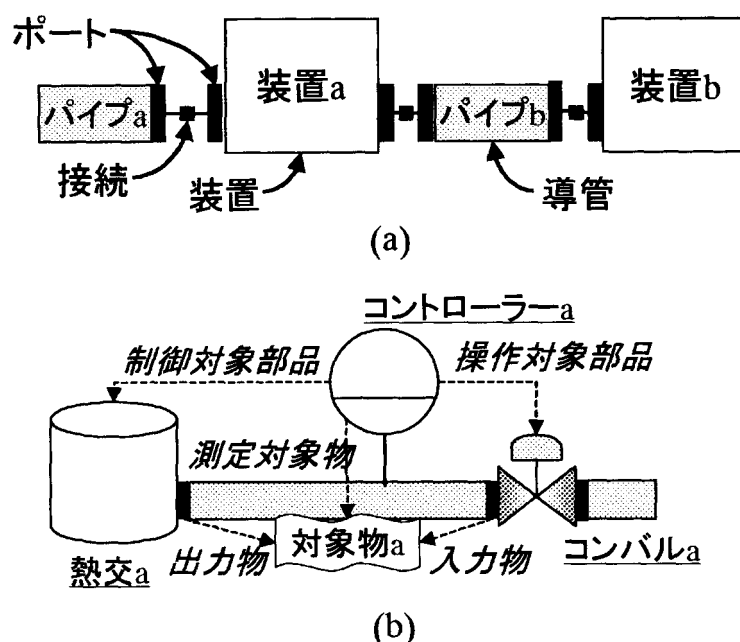


図 2.6-4対象物のモデリングの視点

また、「コントローラー」という装置は他の装置、対象物と様々な関係を持つ。コントローラーはパイプを流れる対象物や装置の「温度」「流量」「液レベル」といった様々な属性を測定し、その属性値を設定値に保つようにアクチュエータに対して開度を指示する装置である。コントローラーが測定する対象を「測定対象物」と呼び、コントローラーが開度を指定する対象を「操作対象部品」と呼ぶ。「操作対象部品」となる装置はアクチュエータのインスタンスに限るのでクラス制約としてそのことが記述されており、モデルを作成する際に、モデル構築者がコントローラーの操作対象部品としてアクチュエータ以外のインスタンスを指定した場合、クラス制約に反していることを指摘することが出来る。この様に、また、論理的にコントローラーが動作を制御している装置を「制御対象部品」と呼ぶ。この様に石油プラントモデルを記述するのに必要な概念が石油精製プラントオントロジーとして定義されている。

対象とする石油精製プラントモデルはこのオントロジーから生成される。オントロジーで各概念や概念間の関係を表す語彙とその定義を用意しており、オントロジーに基づいてモデルを作成、変更することでモデリングの作業が容易になり、一貫したモデリングも保証される。現在、日石菱油エンジニアリング株式会社提供資料をもとに、総部品数約 2000 の石油精製プラントモデルが構築されている。

2.6.2.2. センサ接続情報の扱い

2.6.2.2.1. 論理的接続の定義

ある装置の入力をチェックしているセンサの場合は、そのセンサが設置されている付近のパイプとする。ただし、reflux は入力として扱う。

それ以外の場合は、その検知対象データの呼び名（用途（測定場所）欄のデータ）が示唆する装置とする。ただし、C11 に関しては塔頂、塔底などに分割してそれぞれを独立した装置として扱う。

2.6.2.2.2. 物理的接続の定義

あるパイプを流れる原油などの属性をチェックしているセンサの場合は、そのセンサが設置されている付近のパイプとする。

それ以外の場合は、その検知対象が存在する装置、または、検知対象である装置そのものとする。

具体的なセンサ接続情報は付録 A に示す。

2.6.2.3. 構造検索コマンド

プラント運転支援システムに参加するサブシステムのいくつかは、プラントの構造情報を使った表示や、推論を行う。これらは同じプラントを対象としているが、扱うべき構造の詳細度や、利用目的が異なるため、独自のモデルをサブシステム間で持つよう設計されがちである。しかし、情報が分散してしまうと、変更などのメンテナンスをそれぞれのサブシステム間で整合が取りながら行うことは難しくなる。モデル検索エンジンは、プラントモデルの構造情報を一元管理し、それらに関する様々な質問に答えることで問題を解決する。

2.6.2.3.1. 設計

（1）プラントモデル構造情報の解釈

既に述べたように、装置と導管はその入出力口としてポートを持ち、それらの構造情報は隣り合うポートとポートの接続関係により表現される。石油プラントオートロジーに基づいたプラントモデルは、リッチな情報を持ち、様々な用途に使えるが、「構造情報の検索」という用途に絞った場合、冗長であったり、直接的ではなく扱いにくかったりした形をしている部分がある。そこで、モデル検索エンジンではプラントモデルを「構造情報の検索」に必要な操作を行うのにわかりやすく効率よいデータ構造に変換することを考えた。

第 1 に、装置と導管の接続に含まれる情報の中で大切なのが、導管の中を流れる対象物の向きである。例えば、「ある流量計の指示値がおかしいということがあれば、上流にある部品の異常を疑う」といったときに用いられる。そこで

構造情報は、上流、下流の装置がただちにわかるようにする。また配管は長さ、という属性を持っている。ある装置から何m以内の装置を探す、という要求は十分予想される。当然ながら、装置の名前や、その装置がどんなクラスに所属しているのかは重要な情報である。

そうした考察をもとにプラントモデルを解釈するコンポーネントを開発する。

(2) グレインサイズ管理

HM オントロジーサーバーでは、対象とするプラントの基本的な部品をすべて含むようなモデルを管理している。しかし、エージェントによっては、そのような詳細なモデルでは冗長な場合がある。本研究では、オントロジーサーバーが管理する構造モデルから、意味表示インターフェースサブシステムが要求する構造モデルへの写像を検討することで、簡易なグレインサイズの管理の実現を行う。

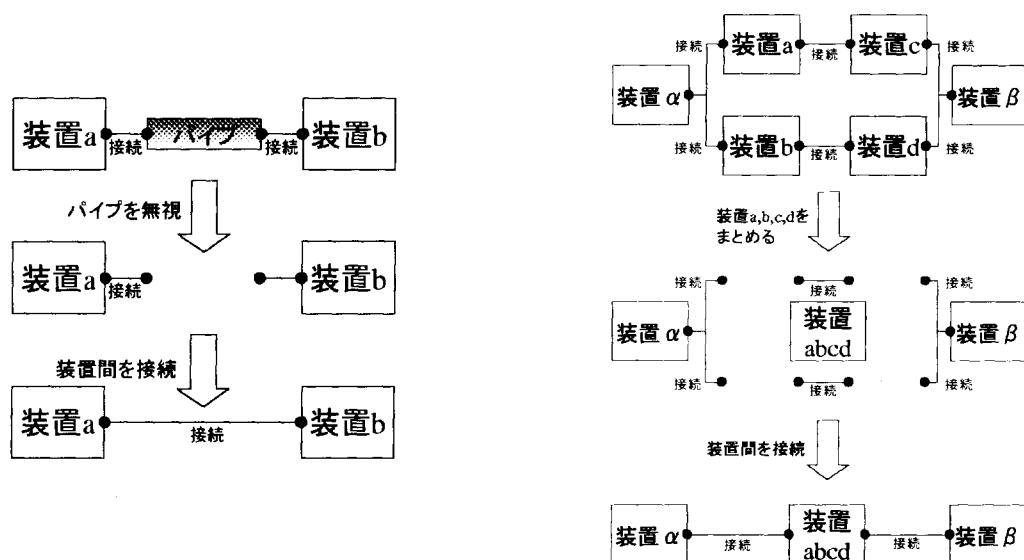


図 2.6-5 (a)部品を無視する例の構造モデル

(b)部品をまとめる例の構造モデル

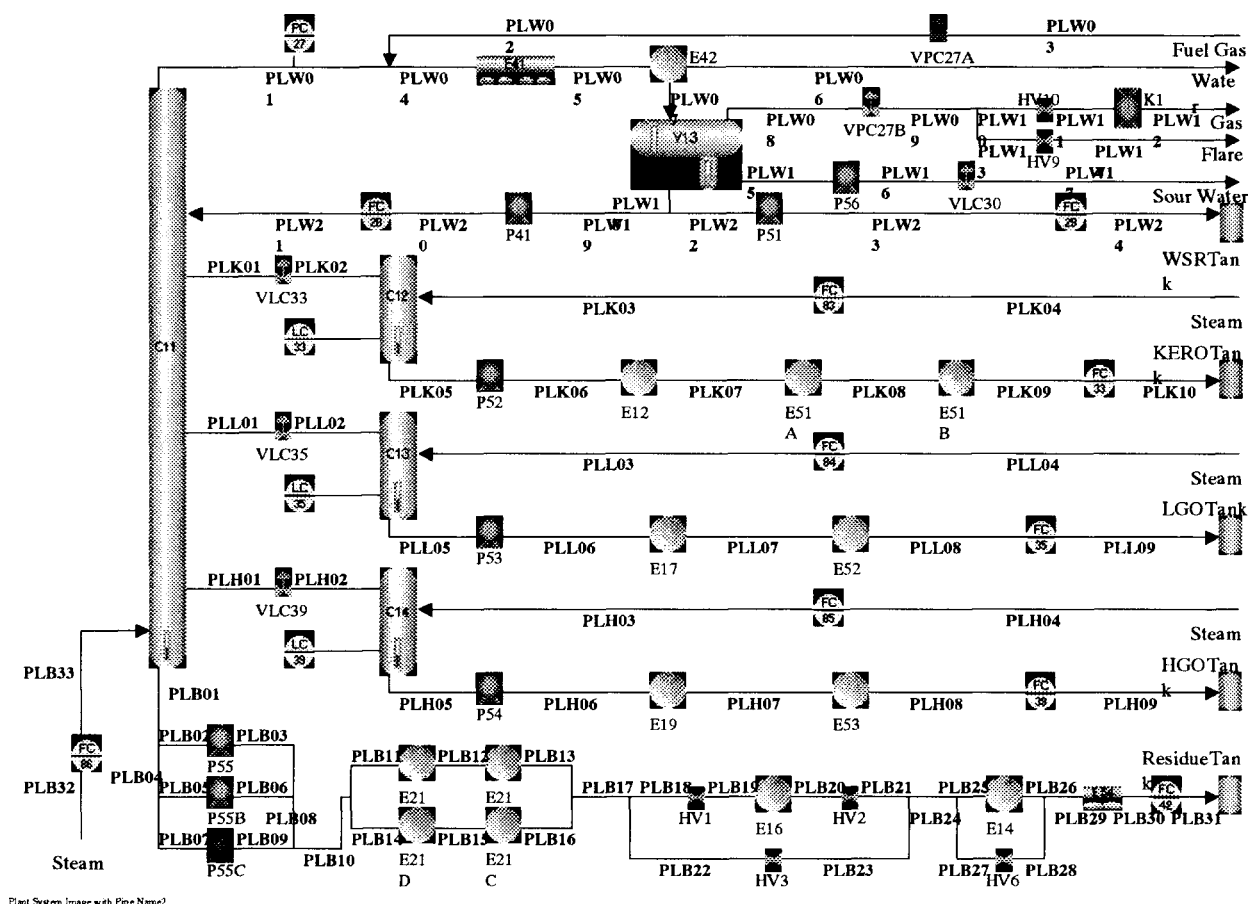


図 2.6-6大画面表示のためのプラントモデル構造モデル（一部）

オントロジーサーバーのと同等の構造を扱う大画面総合表示の構造モデル（図 2.6-6）と意味表示インターフェースの構造モデル（図 2.6-7）を比べてみると、基本的な構造は同一だが、部品数が大きく違うことがわかる。これは、「特定の種類の部品が無視されている」、「複数の同じ種類の部品がひとかたまりに扱われている」ことによる。例えば、パイプ(PLW01 など)は意味表示インターフェースの構造モデルには無い。また valveA は、VFC17、 VFC18、 VFC19、 VFC20 をまとめて扱ったものであることがわかる。そして、部品を無視した場合は、その部品の入力と出力がつながれ(図 2.6-5(a))、複数の部品をまとめた場合は、まとめた部品に含まれるすべての入力をまとめた部品の入力に、すべての出力をまとめた部品の出力とする(図 2.6-5(b))ことで構造の同一性が保たれている。これらの考察から、意味表示インターフェースに提供する構造モデルとの対応は、

Structure Layer

SCENARIO 1 and 3

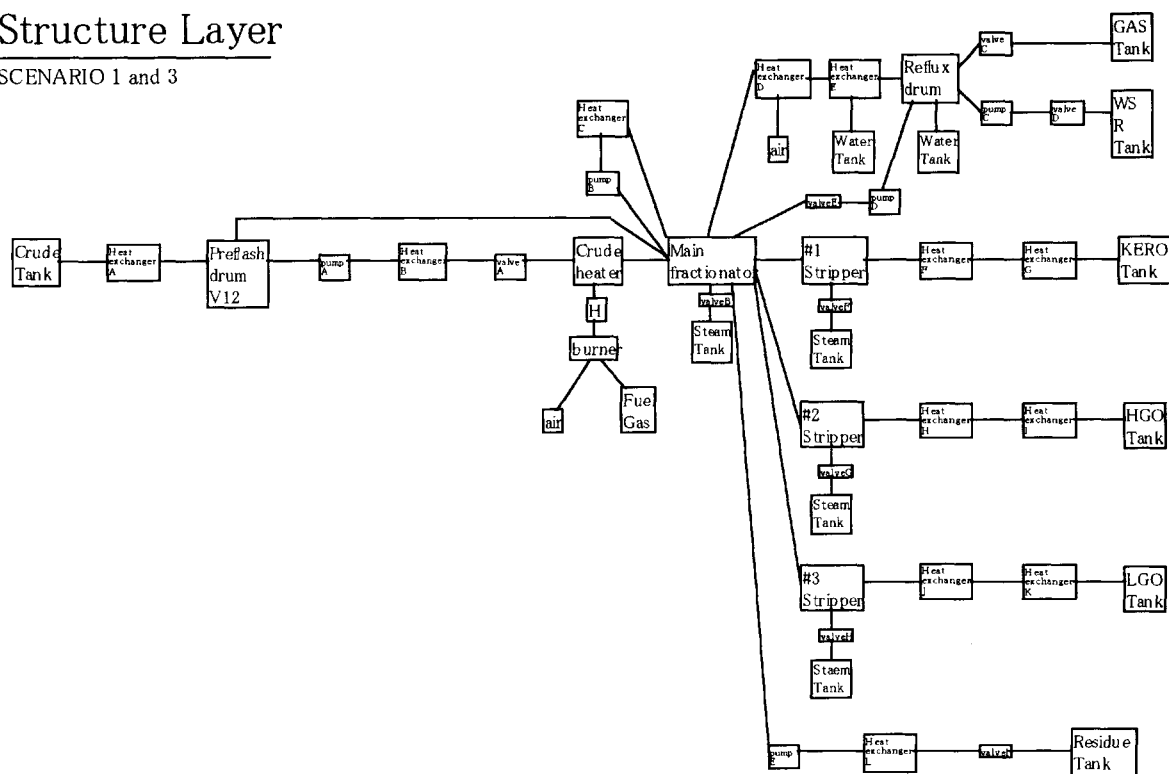


図 2.6-7意味表示インターフェースのためのプラント構造モデル

無視する部品：

パイプ、コントローラー、オフガスコンプレッサー

まとめる部品：

valveA : VFC17、 VFC18、 VFC19、 VFC20

valveB : VFC17、 VFC18、 VFC19、 VFC20

valveC : VPC27A、 VPC27B

pumpA : P12A、 P12B、 P12C

:

:

のように宣言的に書ける。これらの対応を目的に合わせて複数記述すれば、さまざまなグレインサイズに対応できる。

(3) 構造情報の検索

ここでは、次世代プラントの他のサブシステムから構造情報に関するどういった問い合わせがあるか考える。まずは、構造情報全体、そのものが欲しいと

いう場合である。実際に意味表示インタフェースや、大画面総合表示サブシステムは、プラントの構造情報そのものをオペレーターに示すため、全体の情報が必要になる。次に求められるのが、ある装置に異常がおきた時、その原因を調べたり、対策を考えたりするときに利用したい、という要求である。これに対しては装置間のルートを検索したり、逆に特定の装置をバイパスするような経路検索をしたり、重要な装置である熱交換器の関連部品を探したりできることが求められる。これらの要求をもとに以下のような検索を考えた。

(1)接続関係を使った検索

パイプなどによる物理的な接続情報を使った検索。

(2)部品間のルート検索

2つの部品の間のルート検索。

(3)バイパスルート検索

ある部品をバイパスするルート検索。

(4)センサ関連検索

論理的接続部品・物理的接続部品に関連する検索。

・論理的接続部品

(1) ある装置の入力をチェックしているセンサの場合は、そのセンサが設置されている付近のパイプとする。ただし、**reflux** は入力として扱う。

(2) それ以外の場合は、その検知対象データの呼び名（用途（測定場所）欄のデータ）が示唆する装置とする。ただし、C11 に関しては塔頂、塔底などに分割してそれぞれを独立した装置として扱う。

・物理的接続部品

センサが取り付けられている物理的な場所（部品）

(5)操作関連検索

対象物とその属性を操作している部品に関する検索。

(6)ITV 関連検索

カメラとカメラによって監視されている部品に関する検索。

(7)熱交換器周りの機器検索

ある熱交換器に関連する部品を検索。

2.6.2.3.2. 開発

図 2.6-8にモデル検索エンジン全体図を示す。モデル検索エンジンは、図のようにコンモデル解釈コンポーネント、グレインサイズ管理コンポーネント、構造検索コンポーネントからなる。モデル解釈コンポーネントでは、オントロジーサーバー上のプラントモデルから、構造に関する情報を取り出す。グレインサイズ管理コンポーネントでは、プラントモデルの構造情報を必要に応じたグレインサイズで再構成を行う。構造検索コンポーネントでは、構造情報を用いた部品間の経路検索などの検索を行う。以下、各コンポーネントについて説明する。

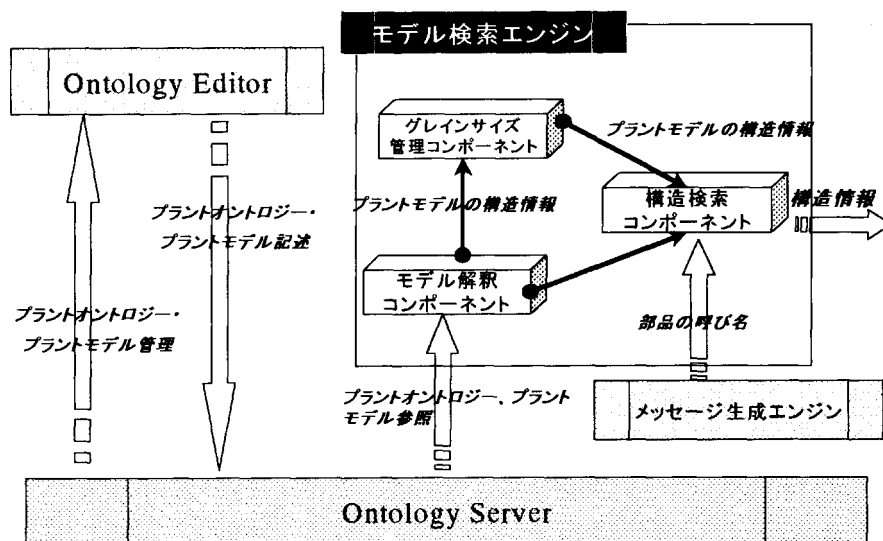


図 2.6-8モデル検索エンジン全体図

・ モデル解釈コンポーネント

オントロジーサーバー上のプラントモデルから、構造に関する情報だけを取り出し、構造情報に関する操作を行いやすいように準備するコンポーネントである。大きく二つに処理が分かれる。一つは(1)オントロジーサーバーのモデルを参照する関数を用いて、検索などの操作に適した構造情報だけのデータ構造を取り出す処理である。もう一つが、(2)オントロジーサーバー上のモデルが変更されていないかを監視し、変更が起きた場合、(1)の処理を再び行いこのコン

ポーネントが持つデータ構造の更新を行い、オントロロジーサーバー上のモデルとの整合性を保つ処理である。これらの処理によって作られた構造情報は、グレインサイズ管理コンポーネントに渡され目的のグレインサイズに変換される。

・ グレインサイズ管理コンポーネント

プラントモデルの構造情報を必要に応じたグレインサイズで再構成を行うコンポーネントである。モデル中に書かれたグレインサイズを決定するための情報は、(1)「無視する部品」と(2)「まとめる部品」からなる。(1)については、無視する部品のクラスか、無視する部品の名前が書かれる。また(2)のまとめる部品は、1つ以上の元の構造情報に含まれる部品とまとめた後の部品名のセットで記述される。まとめる部品については、同じクラスのインスタンスで無ければならない。

これらの情報からモデル解釈コンポーネントから受け取った構造情報に対して、「無視する」か、「まとめる」処理を行った後、「構造の同一性を保つ」処理を行う(図 2.6.5、 2.6.6)。そうして作られた目的にあったグレインサイズの構造情報を、HM のサブシステムに返すか、さらに構造検索コンポーネントに渡して検索処理を行う。

・ 構造検索コンポーネント

構造情報を用いた部品間の経路検索などの検索を行うコンポーネントである。付録Bの仕様に従ってインプリメントを行った。基本的には経路探索ルーチンから成る。モデル解釈コンポーネント、グレインサイズ管理コンポーネントの働きによって、同じプラントモデルに対する異なったグレインサイズの構造情報を受け取るが、構造検索コンポーネントとしては、それらを意識せず常に同じロジックによって検索を行うことが可能になっている。なお検索結果は、オントロロジーサーバーを通じてHMのサブシステムに渡される。

2.6.2.4. メッセージ生成エンジン

メッセージ生成に関しては昨年度で開発は終了しているので内容の本質は昨年度の報告書に記載したとおりであるが、対象とするシナリオが増加したのに伴って全ての想定するメッセージ生成のチェックを行い、その結果に基づいて細かな修正は行われている。付録Cに現在チェックが終了している全メッセージをしめす。

付録A センサー接続情報リスト

タグ名	用途（測定場所）	論理的接続機器	物理的接続機器
FC11A	デソルターへの原油張り込み	PLC03	PLC03
FC11B	デソルターへの原油張り込み	PLC06	PLC06
FC11C	デソルターへの原油張り込み	PLC10	PLC10
FC17	原油加熱炉張り込み	PLC49	PLC49
FC18	原油加熱炉張り込み	PLC53	PLC53
FC19	原油加熱炉張り込み	PLC58	PLC58
FC20	原油加熱炉張り込み	PLC62	PLC62
FC26	No1S/R 流量	PLS02	PLS02
FC28	トップリフラックス流量	PLW20	PLW20
FC29	W S R 抜き出し流量	PLW23	PLW23
FC31	No 2 S / R 流量	PLS07	PLS07
FC33	K e r o 抜き出し量	PLK09	PLK09
FC35	L G O 抜き出し量	PLK08	PLK08
FC37	No 3 S / R 流量	PLS12	PLS12
FC39	H G O 抜き出し量	PLH08	PLH08
FC42	B T M 抜き出し量	PLB30	PLB30
FC51	デソルターへの注入水	PLD02	PLD02
FC71	燃料ガス流量	PLFG(燃料ガス線)	PLFG(燃料ガス線)
FC75	燃焼用空気流量	PLA02	PLA02
FC83	No1 S / S スチーム注入量	PLK03	PLK03
FC84	No 2 S / S スチーム注入量	PLL03	PLL03
FC85	No 3 S / S スチーム注入量	PLH03	PLH03
FC86	ボトムスチーム注入量	PLB32	PLB32
FI14	プレフラ塔頂ガス量	V12	PLC66
FI21	オフガス流量	V13	PLW08
FI30	V13 廃水流量	V13	PLW17
FI41	オーバーフラッシュ量	C11	C11
FI81	過熱スチーム流量	PLSHSTM(加熱炉加熱スチーム線)	PLSHSTM(加熱炉加熱スチーム線)
LC12	V 1 1 (デソルター) 界面	V11	V11
LC15	V 1 2 (プレフラ) 液面	V12	V12
LC29	V 1 3 液面	V13	V13
LC30	V 1 3 廃水液面	V13	V13
LC33	C 1 2 (No 1 S / S) 液面	C12	C12
LC35	C 1 3 (No 2 S / S) 液面	C13	C13
LC39	C 1 4 (No 3 S / S) 液面	C14	C14
LC42	C 1 1 (主精留塔) ボトム液面	C11	C11
PC12	V 1 1 (デソルター) 圧力	V11	V11

PC27	C 1 1 塔頂圧力	C11	PLW01
PC75	加熱炉炉内圧力	H11	H11
PDI12	原油、デソルター水ミキシング用	PLC17	PLC17
PDI41	C 1 1 塔頂塔底の差圧	C11	C11
PI11	P 1 1 A, B, C, 出口合流部圧力	PLC12	PLC12
PI14	V 1 2 (プレフラ) 塔頂圧力	V12	PLC66
PI15	P 1 2 A, B, C, 出口合流部圧力	PLC39	PLC39
PI17	H 1 1 コンベクション出口原油圧力	H11	H11
PI18	H 1 1 コンベクション出口原油圧力	H11	H11
PI19	H 1 1 コンベクション出口原油圧力	H11	H11
PI20	H 1 1 コンベクション出口原油圧力	H11	H11
PI25	C 1 1 への張り込み原油圧力	PLC65	PLC65
PI26	V 1 3 (リフラックスドラム) 圧力	V13	V13
PI42	C 1 1 塔底の圧力	C11	PLB01
PI51	P 1 1 Aの出口圧力	P11A	PLC03
PI52	P 1 1 Bの出口圧力	P11B	PLC06
PI53	P 1 1 Cの出口圧力	P11C	PLC10
PI54	P 1 2 Aの出口圧力	P12A	PLC32
PI55	P 1 2 Bの出口圧力	P12B	PLC35
PI56	P 1 2 Cの出口圧力	P12C	PLC38
PI57	P 5 6 (廃水移送ポンプ)	P56	PLW16
PI58	P 5 1 (WSR抜き出しポンプ) 出口	P51	PLW23
PI59	P 5 9 (トップリフラックスポンプ) 出口	P41	PLW20
PI60	P 4 2 (No 1 S/R) 出口圧力	P42	PLS02
PI61	P 5 2 (Kero抜き出し) 出口圧力	P52	PLK06
PI62	P 4 3 (No 2 S/R) 出口圧力	P43	PLS07
PI63	P 5 3 (LGO抜き出し) 出口圧力	P53	PLL06
PI64	P 5 4 (HGO抜き出し) 出口圧力	P54	PLH06
PI65	P 4 4 (No 3 S/R) 出口圧力	P44	PLS12
PI66	P 5 5 A (BTM抜き出し) 出口圧力	P55A	PLB03
PI67	P 5 5 B (BTM抜き出し) 出口	P55B	PLB06

	圧力		
	P 5 5 C (B T M抜き出し) 出口		
PI68	圧力	P55C	PLB09
PI71	加熱炉燃料ガス圧力	PLFG(燃料ガス線)	PLFG(燃料ガス線)
PI72	加熱炉パイロットガス圧力	PLFG(燃料ガス線)	PLFG(燃料ガス線)
		PLSHSTM(加熱炉加	PLSHSTM(加熱炉加
PI82	過熱スチーム圧力	熱スチーム線)	熱スチーム線)
TC25	加熱炉出口原油温度	H11	PLC65
TC28	C 1 1 塔頂温度	C11	PLW01
TC75	加熱炉排ガス温度	E61	PLA02
TI11	デソルター入り口原油温度	PLC17	PLC17
TI12	デソルター出口原油温度	V11	PLC18
TI13	プレフラ入り口温度	PLC29	PLC29
TI14	プレフラ塔頂ガス温度	V12	PLC66
TI15	プレフラ塔底温度	V12	PLC30
TI16	加熱炉入り口原油温度	PLC48	PLC48
	加熱炉コンベクション出口原油温		
TI17	度	H11	PLCH1
	加熱炉コンベクション出口原油温		
TI18	度	H11	PLCH2
	加熱炉コンベクション出口原油温		
TI19	度	H11	PLCH3
	加熱炉コンベクション出口原油温		
TI20	度	H11	PLCH4
TI21	加熱炉出口原油温度	H11	PLC51
TI22	加熱炉出口原油温度	H11	PLC55
TI23	加熱炉出口原油温度	H11	PLC60
TI24	加熱炉出口原油温度	H11	PLC64
TI26	No1S/R 抜き出し温度	PLS02	PLS02
TI27	No1S/R 戻り温度	PLS05	PLS05
TI29	C 1 1 塔頂、E 4 1 出口温度	E41	PLW05
TI30	V 1 3 温度	V13	PLW18
TI31	No 2 S/R 抜き出し温度	PLS07	PLS07
TI32	No 2 S/R 戻り温度	PLS10	PLS10
	K e r o 主精留塔からの抜き出し		
TI33	温度	PLK01	PLK01
TI34	K e r o、タンクへの流出温度	PLK10	PLK10
	L G O 主精留塔からの抜き出し温		
TI35	度	PLL01	PLL01
TI36	L G O、タンクへの流出温度	PLL09	PLL09
TI37	No 3 S/R 抜き出し温度	PLS12	PLS12
TI38	No 3 S/R 戻り温度	PLS15	PLS15
	H G O 主精留塔からの抜き出し温		
TI39	度	PLH01	PLH01

TI40		HGO、タンクへの流出温度	PLH09	PLH09
TI42		C 1 1 塔底温度	PLB01	PLB01
TI43		BTM、タンクへの流出温度	PLB30	PLB30
TI76		加熱炉燃焼ガス温度	H11	PLA01
TI81		過熱スチーム加熱炉入り口温度	PLSHSTM(加熱炉加 熱スチーム線)	PLSHSTM(加熱炉加 熱スチーム線)
TI82		過熱スチーム加熱炉出口温度	H11	PLSHSTM(加熱炉加 熱スチーム線)
VI12		デソルター電圧	V11	V11
AI76		加熱炉排ガス HC濃度	H11	H11
AI77		加熱炉排ガス O2濃度	H11	H11
AI78		加熱炉排ガス CO濃度	H11	H11
WSR	IBP	オンストリームアナライザー分析 値	PLW23	PLW23
WSR	ASTM	オンストリームアナライザー分析 値	PLW23	PLW23
WSR	RVP	オンストリームアナライザー分析 値	PLW23	PLW23
KERO	IBP	オンストリームアナライザー分析 値	PLK09	PLK09
KERO	ASTM	オンストリームアナライザー分析 値	PLK09	PLK09
KERO	SMOKE	オンストリームアナライザー分析 値	PLK09	PLK09
KERO	FLASH	オンストリームアナライザー分析 値	PLK09	PLK09
LGO	IBP	オンストリームアナライザー分析 値	PLL08	PLL08
LGO	ASTM	オンストリームアナライザー分析 値	PLL08	PLL08
LGO	PP	オンストリームアナライザー分析 値	PLL08	PLL08
HGO	IBP	オンストリームアナライザー分析 値	PLH08	PLH08
HGO	ASTM	オンストリームアナライザー分析 値	PLH08	PLH08
HGO	PP	オンストリームアナライザー分析 値	PLH08	PLH08
RESIDUEV ISC	SSU	オンストリームアナライザー分析 値	PLB30	PLB30

付録B モデル検索コマンド仕様

処理タスク名：TASK_OS_SEARCH_MODEL

※ 斜体で示したものは
※ 省略不可能な引数

●フォーマット：

①接続関係を使って検索

(1) 条件にあった部品を検索

```
[ "plant-structure"
  ["connection-part", DIRECTION, PART1,
    "depth", DEPTH, "class", CLASS, "distance", DISTANCE]
]
```

説明：

PART1 部品から *DIRECTION* の方向に *DEPTH* の深さまで部品の接続関係を用いて検索する。
検索結果のうち、*CLASS* クラスのインスタンスであり、*PART1* 部品から *DISTANCE* m以内の部品を返す。

DIRECTION:

"upstream" 上流の部品を検索

"downstream" 下流の部品を検索

(省略時 上流と下流の部品を検索)

DEPTH:

検索する深さ(省略時 "1")

CLASS:

クラス制約(省略時 制約無し)

DISTANCE:

"Nm" Nは正数。単位 (m) を付ける：Nメートル以内にある部品を検索(省略時 制約無し)

戻り値：

[part1、 part2…]

例：

1)E16 の前後のバルブを調べる

```
[ "plant-structure"
  [ " connection-part " "E16" "class" "手動バルブ" ]]
```

※E16 の上流、下流をそれぞれ調べ、最初に見つけた手動バルブのインスタンスを上流、下流とも一つずつ返す

結果：["HV1" "HV2"]

(2) すべての部品を要求(For LD エージェント)

```
[ "ld-plant-structure"
  ["connection-part", "all"]]
```

戻り値：

```
[
  [:NAME PLC02 :CLASS パイプ :CONNECTION | :NAME / 部品名
    [:NUMBER 4 | :NUMBER n / 部品数
      [:NAME PLC01、 :STREAM、 -1] | :STREAM -1 / 上流にある部品
      [:NAME P11A :STREAM 1] | :STREAM 1 / 下流にある部品
      [:NAME PLC05 :STREAM、 0]、
      [:NAME PLC09 :STREAM、 0]、 | :STREAM 0 / 上記以外で繋がっている部品
    ]
  ]
]
```


(3) すべての部品を要求(For SI エージェント)

```
[ "si-plant-structure"  
  ["connection-part", "all"] ]
```

Phase2 と変更なし。

②部品間のルート検索

```
[ "plant-structure"  
  ["route", DIRECTION, PART1, PART2, "pass", PART3, "class", CLASS ]  
]
```

説明：

PART1 部品から PART2 部品まで DIRECTION 方向に進んで到達するルートの中で PART3 部品が含まれるルートを数え上げる。各ルート上に存在する部品のうち CLASS クラスのインスタンス部品の列を返す。

DIRECTION:

"upstream" 上流の部品を検索

"downstream" 下流の部品を検索

(省略時 上流と下流の部品を検索)

"pass" PART3:

通過部品。インスタンス、クラスの指定が可能 (省略時 制約無し)

CLASS:

クラス制約 (省略時 制約無し)

戻り値：

部品の並びが上流、下流の順でリストで返る

$[[route_1, route_2, \dots], [route_3, route_4, \dots]]$ ※ $route_i = [part_i, part_i, \dots]$

上流の検索結果

下流の検索結果

例：

1) V11 と V12 の間のパイプを検索

```
[ "plant-structure"  
  [ "route" "downstream" "V11" "V12" "class" "パイプ" ] ]
```

※V11 と V12 の間の部品の並びのうち、パイプのインスタンスであるものを取り出す。

結果[[["PLC18" "PLC19"..."PLC23" "PLC29"] ["PLC18" "PLC24"..."PLC28" "PLC29"]]]

2) V11 と V12 の間のパイプのルートのうち HV4 を通るものを検索

```
[ "plant-structure"  
  [ "route" "downstream" "V11" "V12" "pass" "HV4" "class" "パイプ" ] ]
```

※V11 と V12 の間の部品の並びのうち、HV4 を含むものを選び、パイプのインスタンスであるものを取り出す。

結果：[[["PLC18" "PLC19"..."PLC23" "PLC29"]]]

③バイパスルート検索

```
[ "plant-structure"  
  ["bypass", PART1, PART2, "pass", PART3, "class", CLASS ]  
]
```

※PART2 は省略可能

説明：

・ PART1 の上流の最初の分岐点を探す(分岐点 A とする)

・ PART2 (PART2 が省略された場合、PART1) の下流の最初の分岐点を探す(分岐点 B とする)

・ A から B へ下流方向へ向かって、到達可能なルートを求める。求めたルートのうち PART3 を含むルートのうち、CLASS クラスのインスタンスの部品の列を返す。

"pass" PART3:

通過部品。インスタンス、クラスの指定が可能 (省略時 制約無し)

CLASS:

クラス制約（省略時 制約無し）

戻り値:

部品の並びが上流、下流の順でリストで返る

[[route₁, route₂...]] [[route₃, route₄...]] ※route_n = [part_i, part_j...]
上流の検索結果 下流の検索結果

例:

1)HV1 と HV2 をバイパスするルートにある手動バルブを検索

```
[ "plant-structure"
  [ "bypass" "HV1" "HV2" "class" "手動バルブ" ] ]
```

※HV1、HV2 をバイパスする部品の並びの一番内側なものうち、手動バルブのインスタンスであるものを取り出す。

結果: [[["HV3"]]]

④部品間のルート上にある部品の集合

```
[ "plant-structure"
  [ "set", DIRECTION, PART1, PART2, "pass", PART3, "class", CLASS ]
]
```

説明:

PART1 部品から PART2 部品まで DIRECTION 方向に進んで到達するルートの中で PART3 部品が含まれるルート数を数える。各ルート上に存在する部品のうち CLASS クラスのインスタンス部品の列を返す。route との違いは戻り値のみで、一つのベクターに重複部品を省いた部品集合を返す。

DIRECTION:

"upstream" 上流の部品を検索

"downstream" 下流の部品を検索

(省略時 上流と下流の部品を検索)

"pass" PART3:

通過部品。インスタンス、クラスの指定が可能（省略時 制約無し）

CLASS:

クラス制約（省略時 制約無し）

戻り値:

[part1, part2...]

例:

1)V11 と V12 の間の全パイプを検索

```
[ "plant-structure"
  [ "set" "downstream" "V11" "V12" "class" "パイプ" ] ]
```

※V11 と V12 の間の部品の並びのうち、パイプのインスタンスであるものを取り出す。

結果["PLC18" "PLC19"...PLC23" "PLC29" "PLC24"...PLC28"]

2)V11 と V12 の間のルートのうち HV4 を通るルート上の全てのパイプを重複を除いて検索

```
[ "plant-structure"
  [ "set" "downstream" "V11" "V12" "pass" "HV4" "class" "パイプ" ] ]
```

結果: ["PLC18" "PLC19"...PLC23" "PLC29"]

⑤センサ関連検索

```
[ "plant-structure"
  [ "sense", PART ]
]
```

説明：

PART にくるものの所属クラスによって処理が異なる

- A) PART が測定部品である場合、論理的接続部品・物理的接続部品を返す
- B) PART が測定部品以外の場合、PART と論理的接続をしている部品・物理接続している部品を返す

論理的接続部品

- (1) ある装置の入力をチェックしているセンサの場合は、そのセンサが設置されている付近のパイプとする。ただし、reflux は入力として扱う。
- (2) それ以外の場合は、その検知対象データの呼び名（用途（測定場所）欄のデータ）が示唆する装置とする。ただし、C11 に関しては塔頂、塔底などに分割してそれぞれを独立した装置として扱う。

物理的接続部品

センサが取り付けられている物理的な場所（部品）

戻り値：

[[LogicalEquip1、 LogicalEquip2、 ...]、 [PhysicalEquip1、 PhysicalEquip2、 ...]]

例：

- 1) FI30 の論理的接続・物理的接続の部品を検索

["plant-structure"
["sense" "FI30"]]

結果：[["V13"] ["PLW17"]]

- 2) V13 と論理的接続をしているセンサ・物理接続しているセンサを検索

["plant-structure"
["sense" "V13"]]

結果：[["LC29" "LC30" "PI21" "TI30" "FI21" "PI26"] ["LC29" "LC30" "PI21"]]

⑥操作関連検索

["plant-structure"
["operate"、 PART]
]

説明：

PART にくるものの所属クラスによって処理が異なる

- A) PART が対象物と属性のペアの場合、対象物の属性を操作している部品を返す。
- B) PART が部品の場合、その部品が操作している対象物と属性のペアを返す。

戻り値：

[part1、 part2...]

例：

- 1) 加熱炉入口の流量を操作している部品を検索

["plant-structure"
["operate" ["加熱炉入口" "流量"]]]

結果：["FC17" "FC18" "FC19" "FC20"]

⑦ITV 関連検索

["plant-structure"
["itv"、 PART]
]

説明：

PART にくるものの所属クラスによって処理が異なる

- A) PART がカメラである場合、監視されている部品を返す
B) PART がカメラ以外の場合、PART を監視しているカメラを返す

戻り値：

[part1、 part2…]

例：

1)APH を監視している部品を検索

```
[ "plant-structure"  
  [ "itv" "APH" ]]
```

結果：["ITV1"]

⑧熱交換器周りの機器検索

```
[ "plant-structure"  
  ["around"、 PART、 "class"、 CLASS]  
]
```

CLASS:

クラス制約（省略時 制約無し）

説明：

PART の上流・下流を調べ最初の分岐点までの部品と、存在すればバイパスルートの部品を数え上げる。PART は基本的には熱交換器を想定している。

戻り値：

[part1、 part2…]

例：

1)E16 周りの部品を検索

```
[ "plant-structure"  
  [ "around" "E16" ]]
```

結果：["HV1" "HV2" "HV4" "HV5"
 "PLB19" "PLB20" "PLC12" "E15" "PLC13" "PLC14" "HV3" "PLB22" "PLB23"]

⑨センサが異常を感知したときの異常表示箇所検索

```
[ "plant-structure"  
  ["trouble"、 PART]  
]
```

説明：

戻り値：

[part1、 part2…]

例：

1)FI30 が異常を感知したときの異常発生箇所を検索

```
[ "plant-structure"  
  [ "trouble" "FI30" ]]
```

結果：["V13" "P56"]

参考：

FI30 の関連部品を検索

```
[ "plant-structure"  
  [ "sense" "FI30" ]]
```

結果：[["V13"] ["PLW17"]]

付録C 確認済みメッセージのリスト

シナリオ	チェック	生成文章
1	○	警告、リフラクストラム液レベルが異常です
1	○	警告、リフラクストラム液レベルが上昇しています
1	○	リフラクストラムがオーバーフローまで 10 分です
1	○	異常原因として、以下の候補が考えられます。
		1. ナフサ抽出しポンプ性能低下
		2. リフラクストラム液レベル調整弁異常
		3. リフラクストラム液レベルセンサ異常
		4. トップー制御系異常
		5. 原油の大幅な軽質化
1	○	異常原因の推定は、以下の結果となりました。
		○ 1. ナフサ抽出しポンプの性能低下
		× 2. リフラクストラム液レベル調整弁異常
		× 3. リフラクストラム液レベルセンサ異常
		× 4. トップー制御系異常
		× 5. 原油の大幅な軽質化
1	○	ナフサ抽出しポンプの性能が低下している可能性があります
1	○	過去の事例では、以下のように操作しました。
		FC17 のセットポイントを 128 にしました。
		FC18 のセットポイントを 128 にしました。
		FC19 のセットポイントを 128 にしました。
		FC20 のセットポイントを 128 にしました。
1	○	対応操作として、以下のような候補があります。
		1. 原油タンク出口の流量を低下させてください。
2	○	警告、蒸留塔の No. XX トレイが異常です
2	○	警告、蒸留塔の No. XX トレイ負荷が上昇しています
2	○	フラッティングまで 30 分です。
2	○	異常原因として、以下の候補が考えられます。
		1. 原油チャージ量増加
		2. 計器指示不良
		3. 原油性状軽質化
		4. 塔圧力低下
		5. 加熱炉出口原油温度上昇
		6. 製品灯油抜き出し流量計指示不良
2	○	異常原因の推定は、以下の結果となりました。
		○ 1. 原油チャージ量増加
		○ 2. 計器指示不良
		○ 3. 原油性状軽質化
		× 4. 塔圧力低下
		× 5. 加熱炉出口温度上昇

		○ 6. 製品灯油抜き出し流量計指示不良
2	○	対策としてチャージ流量計の流量を下げてください
2	○	対策として、通油量を低下させてください
3	○	加熱炉出口原油温度が異常です
3	○	警告、加熱炉出口温度が低下しています
3	○	加熱炉出口温度が下限値を超えるまで 30 分です
3	○	加熱炉燃焼用空気が過剰
3	○	APH の効率が低下している可能性があります
3	○	燃料ガス比重変動の可能性があります
3	○	異常原因として、以下の候補が考えられます。
		1. 加熱炉燃焼用空気が過剰
		2. 計器不良
		3. FG の C/V 異常
		4. APH 効率低下
		5. プレフラのフラッシュ量低下
		6. FG バーナ消火
		7. チャージ量増加
		8. 加熱炉入口温度低下
		9. ガス比重変動
3	○	異常原因の推定結果は、以下となりました。
		× 1. 加熱炉燃焼用空気が過剰
		× 2. 計器不良
		× 3. FG の C/V 異常
		○ 4. APH 効率低下
		× 5. プレフラのフラッシュ量低下
		× 6. FG バーナ消火
		× 7. チャージ量増加
		× 8. 加熱炉入口温度低下
		○ 9. ガス比重変動
3	○	対応操作として、以下の候補が考えられます。
		1. 通油量を低下させてください。”
		2. 燃料ガスの比重を増加させてください。”
		3. FG 系の圧力を増加させてください。”
		4. サイドリフラックスからの熱回収を増加させてください。
4	○	警告、蒸留塔塔頂圧力が異常です
4	○	警告、蒸留塔塔頂圧力が上昇しています
4	○	蒸留塔塔頂圧力が上限値を超えるまで 10 分です
4	○	異常原因として、以下の候補が考えられます。
		1. 蒸留塔塔頂圧力計指示不良
		2. 塔頂系冷却能力不足
		3. トップリフラックス流量低下
		4. チャージ量増加
		5. 処理原油増加

		6.オフガスコンプレッサトリップ
4	○	異常原因の推定結果は、以下となりました。
		× 1.蒸留塔塔頂圧力計指示不良
		× 2.塔頂系冷却能力不足
		× 3.トップリフラックス量低下
		× 4.チャージ量増加
		× 5.処理原油増加
		○ 6.オフガスコンプレッサトリップ
4	○	過去の事例では、以下のように操作しました。
		PC27 のセットポイントを 1.2 にしました。
4	○	対応操作として、以下のような候補があります。
		1. PC27 のセットポイントを上げてください
		2. 原油タンク出口の流量を低下させてください
5	○	加熱炉#1 パス流量コントロールバルブが異常です
5	○	加熱炉チャージコントロールバルブ FC17FC18FC19FC20 が異常です
5	○	警告、加熱炉チャージコントロールバルブ開度が上限値を超えています
5	○	加熱炉チャージ量が下限値を超えるまで 30 分です
5	○	異常原因として、以下の候補が考えられます。
		1. 加熱炉入口流量計指示不良
		2. 加熱炉入口コントロールバルブ作動不良
		3. 熱交漏れ
		4. ストレーナ・熱交換器系統つまり
		5. ポンプ故障・性能低下
5	○	異常原因の推定結果は、以下となりました。
		× 1.加熱炉入口流量計指示不良
		× 2.加熱炉入口コントロールバルブ作動不良
		× 3.熱交漏れ
		○ 4.ストレーナ・熱交換器系統つまり
		○ 5.ポンプ故障・性能低下
5	○	対応操作として、以下の候補が考えられます。
		1.通油量を低下させてください
		2.ポンプの回転数をアップしてください
6	○	警告、加熱炉排ガス CO 濃度が異常です
6	○	警告、加熱炉排ガス CO 濃度が上昇しています
6	○	加熱炉排ガス CO 濃度が上限値を超えるまで 30 分です
6	○	異常原因として、以下の候補が考えられます。
		1. FDF 故障・停止
		2. IDF 故障・停止
		3. 原油加熱炉出口温度不良
		4. 原油流量コントロール不良
		5. FG 使用量コントロール不良
		6. FG 重質化

		7. コイル油漏れ”
6	○	異常原因の推定結果は、以下となりました。
		× 1. FDF 故障・停止
		× 2. IDF 故障・停止
		× 3. 原油加熱炉出口温度不良
		× 4. 原油流量コントロール不良
		○ 5. FG 使用量コントロール不良
		○ 6. FG 重質化
		○ 7. コイル油漏れ”
6	○	対応操作として、加熱炉燃料空気を増加してください。
6	○	対応操作として、緊急停止をしてください。
7	△	警告、灯油 95%蒸留点が異常です
7	△	警告、灯油 95%蒸留点が重くなっています
7	○	異常原因として、以下の候補が考えられます。
		1. 灯油抜き出し量増加
		2. 原油性状重質化
		3. アナライザ不調
		4. リフラックス量変動
		5. 原油チャージ量減少
		6. 原油・灯油熱交換器チューブリーク
7	○	異常原因の推定結果は、以下となりました。
		× 1. 灯油抜き出し量増加
		× 2. 原油性状重質化
		× 3. アナライザ不調
		× 4. リフラックス量変動
		× 5. 原油チャージ量減少
		○ 6. 原油・灯油熱交換器チューブリーク
8	○	警告、原油通油量が異常です
8	○	警告、原油通油量が減少しています
8	○	原油通油量が下限値を超えるまで 30 分です
8	○	異常原因として、以下の候補が考えられます。
		1. E16 チューブリーク
8	○	異常原因の推定結果は、以下となりました。
		○ 1. E16 チューブリーク
8	○	過去の事例では、以下のように操作しました。
		FC17 のセットポイントを 128 にしました。
		FC18 のセットポイントを 128 にしました。
		FC19 のセットポイントを 128 にしました。
		FC20 のセットポイントを 128 にしました。
9	○	警告、加熱炉#1 パス流量コントローラが異常です
9	○	警告、加熱炉#1 パス流量コントローラ開度が上昇しています
9	○	加熱炉#1 パス流量が下限値を超えるまで 30 分です
9	○	異常原因として、以下の候補が考えられます。

		1.原油チャージ量増加
		2.加熱炉チャージポンプ性能低下
		3.P12 下流の熱交換器配管詰まり
		4.加熱炉チューブ詰まり
		5.FC17 計器異常
9	○	異常原因の推定結果は、以下となりました。
		× 1.原油チャージ量増加
		× 2.加熱炉チャージポンプ性能低下
		× 3.P12 下流の熱交換器配管詰まり
		○ 4.加熱炉チューブ詰まり
		× 5.FC17 計器異常
9	○	過去の事例では、以下のように操作しました。
		加熱炉出口温度のセットポイントを 128 にしました。
9	○	対策として、加熱炉出口温度のセットポイントを低下させてください。

2.7. 意味表示インタフェースの研究

平成12年度の研究開発においては、

- ・異常原因推定手法の開発、
- ・運転知識と高速シミュレータによる対応操作の選定手法の開発、
- ・表示意図に基づく意味表示の実装、
- ・最終プロトタイプシステムの開発
- ・Java版プラントモデル構築ツールの整備、

を行った。以下では、その各々について成果を報告する。

2.7.1. 異常原因推定手法の開発

ユーザサイドの強い要望により、異常原因推定手法を新たに追加開発し、最終プロトタイプシステムに組み込んだ。本手法では、プラント構成機器やシステムに対する部分的な機能モデルと機器構造に基づき故障モードを定義し、故障モードの発生可能性を判定する推定ルールにより、異常状態が検知された場所から機器接続情報を利用して探索範囲を拡げながら異常原因の推定を行う。

本手法では、対象とするシステム全体を一つのモデルとして異常原因を推定するのではなく、システムの各機器やシステムごとの部分モデルを用いて推論を行うため、機器やシステムの種類に応じた異常原因の推定が行える。このことから、システム内の構成が変更された場合にも、変更された機器やシステムのモデルを新たに作成して、追加することで柔軟に対応することができる。また、部分モデルの詳細度は一定レベルに合わせる必要がないため、特定のコンポーネントに対して詳細な診断を行うことも可能である。さらに、異常原因の推定を各機器やシステムごとに順次探索範囲を拡大しつつ行うので、この推論過程を示すことで運転員が異常の影響波及を認識する事ができる。

(1) 推定のための部分モデル

推定の基礎となるプラント部分の機能モデルの作成には、MFMを用いる。まず、対象とする機器やシステムの目標の設定を行う。次に、機能を質量やエネルギーの保存を考慮してモデル化する。こうして作成した機能と目標を表現したモデル（機能－目標モデル）に、機器の構造を考慮して、次に述べる故障モードと推定ルールを診断知識ベースとして付加して推定モデルとする。

故障モードとは、機器やシステムで発生のある故障の種類であり、作成した機能－目標モデルと機器やシステムの構造の特徴から生成する。故障モードは対象とした機器やシステム自身での故障と、上流や下流に接続されている機器やシステムでの故障の2種類に分類できる。機器やシステム自身での故障とは、モデル化の

対象とした機器や系統で発生の可能性のある特定の故障である。また、接続されている他の機器や系統での故障とは、対象とした機器や系統の挙動に、影響波及する他の機器や系統での故障である。

推定ルールは、故障モードの発生の有無を部分モデル各部の変数の値の変化や機能間の関係から判定するための知識である。ここで、変数には流量、液位、温度、圧力、流量調節弁の開度などがあり、考慮した推定ルールは、次の 4 つに大きく分けることができる。

(i) 変数の値が正常範囲内であるか否かに基づく推定ルール

これは、変数の値が正常範囲内であるか正常範囲を越えて増加、あるいは、減少しているかにより故障モードの発生可能性を推定するルールである。

(ii) 保存則が成立しているか否かに基づく推定ルール

これは、コンポーネントの流入量、液位、流出量の関係が質量保存則を満たしているか、また、エネルギー保存則が成立するかによって推定を行うルールである。保存則が成立している場合は故障モードの発生は否定される。

(iii) 計装信号値のクロスチェックに基づく推定ルール

これは、プラントの構造上から関係のある計装信号を比較し、不整合が発生しているか否かによって推定を行うルールである。例えば、加熱炉においては、油の流路は通常複数あるが、加熱炉が正常であり温度センサが故障していなければ、それらの流路における温度はほぼ等しい。

(iv) 性能曲線との比較に基づく推定ルール

例えば、ポンプの正常性能は、ポンプ締め切りヘッドとポンプの最大吐出流量の関係から曲線で表される。この関係を用いることにより、ポンプの性能が低下しているか否かの推定が行える。

推定ルールによる推定結果は、故障モードの発生が肯定される場合、発生の可能性が示唆される場合、発生していないことが示唆される場合、発生が否定される場合の 4 段階に分類した。

なお、対象プラントに対する部分モデルの関連を図 2.7-1 に示す。図において、影をつけた部分モデルは本年度の研究開発において開発したものである。

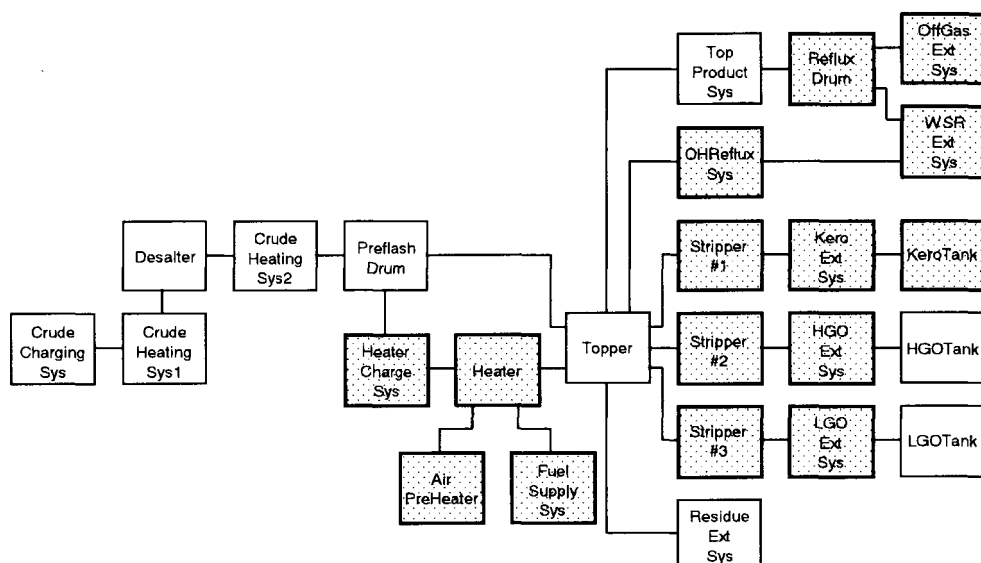


図 2.7-1異常原因推定のための部分モデル関連図

(2) 異常原因推定アルゴリズム

異常原因推定アルゴリズムは以下のとおりである。

- (Step1) 異常の兆候が検知されたコンポーネント（機器や系統）を探索範囲集合に登録する。
- (Step2) 探索範囲に登録されたコンポーネントの中から診断対象を1つ選ぶ。
そして、選ばれた診断対象は探索範囲集合から削除する。
- (Step3) 推定ルールを利用して、診断対象の故障モードの発生可能性を検討する。検討後、肯定、支持、不支持、否定の4種類に判定された故障モードはそれぞれのリストに登録する。
- (Step4) 接続されている上流や下流のコンポーネントの故障モードが肯定あるいは支持された場合には、そのコンポーネントを探索範囲集合に登録する。
- (Step5) 肯定された故障モードがある場合には、異常原因が同定されたものとしてそれを出力し、異常原因の推定を終了する。
- (Step6) 探索範囲集合が空集合でなければ、(Step2)へ戻る。
- (Step7) 支持に判定された故障モードがない場合には、推定結果を「異常原因不明」として異常原因の推定を終了する。

(Step8) 支持に判定された故障モードのうち、不指示や否定されていないものを異常原因として出力する。

(3) 異常原因推定例

異常原因推定例として、(i) ナフサ抜き出しポンプの性能低下によるリフラックスドラムの液位上昇異常と、(ii) 加熱炉への燃料比重変化による加熱炉出口温度低下異常の2つの異常事象に対する推定結果を表 2.7-1に示す。表には、異常原因推定の探索対象部分モデルと推定結果を示している。ここで、探索対象部分モデルにおける番号は探索順であり、同じ順位はどちらが先であるかは任意であることを表す。異常事象 (i) に対する異常原因推定結果は真の異常原因を推定できている。一方、異常事象 (ii) に対する異常原因推定結果では、真の異常原因を含む3つの異常原因が推定されている。しかしながら、これら3つの異常原因はプラントに設置されたセンサの値だけでは弁別できないものであり、推定結果は妥当と考えられる。

表 2.7-1異常原因推定結果例

異常原因	探索対象部分モデル	異常原因推定結果
ナフサ抜き出しポンプ性能低下	1.リフラックスドラム 2.ナフサ抜き出し系統	ナフサ抜き出しポンプ性能低下
加熱炉燃料比重変動	1.加熱炉 2.加熱炉空気余熱器 2.加熱炉燃料ガス供給系統	燃料ガス使用量コントロール不良 加熱炉空気余熱器の効率低下 燃料ガス比重変動

2.7.2. 運転知識と高速シミュレータによる対応操作の選定手法の開発

プラントの機能階層に基づき定性推論を適用した対応操作候補の導出手法では、可能性のある対応操作候補を導出することを目的としているため、対応操作の操作性や定量的効果を考慮していなかった。そのため、多数の対応操作が導出されるが、すべての導出対応操作を運転員に表示することは实际的ではない。そこで、導出された対応操作候補の優先順位づけを行い、絞り込まれた対応操作候補に対してその操作量や効果を数値シミュレーションにより評価することとした。以下に、対応操作候補の絞り込み手法、高速シミュレータの評価、そして、高速シミュレータを用いた対応操作候補の評価について述べる。

(1) 対応操作候補の絞り込み

運転員は異常時の対応操作として、運転員手順書に記載の操作を確実に実施することが求められている。また、日常の運転経験から、ある操作の難易性やプラント挙動への影響度を把握している。従って、運転員手順書や運転員の経験的知識を利用して対応操作を絞り込むことができる。ここでは、これらの知識を、(a) 操作性、(b) 通常性、(c) 操作効果の経験的知識の観点から整理した。操作性とは、操作の難易度を数値化したものである。例えば、中央制御室で操作可能な対応操作の操作性は高いが、現場での作業が伴う対応操作は操作性が低い。通常性とは、プラントの起動、シャットダウンやプラント運転状態の変更など、通常の運転操作において実施するかどうかの点で、対応操作を評価したものである。滅多に行わない操作は通常性が低い。また、運転員が持つ操作効果の経験的知識は、操作に対して効果の現れるプラント変数とその変化方向を組として格納しておく。

さらに、対応操作効果のプラント挙動への影響は、操作の種類によって以下の違いが一般的に生じる。すなわち、

- (1) 質量の流入や流出の効果は主に下流に及ぶ、
- (2) エネルギーの流入や流出の効果は主に下流に及ぶ、
- (3) 圧力の変動は上流にも及ぶ

である。これらのことから、プラント構成機器の作動流体の流れの位置によって、対応操作の効果の可能性を評価できる。

効果の可能性評価のために、隣り合うプラント機器の関係を次の行列 A により表現する。すなわち、行列 A の要素を a_{ij} とすると、

$a_{ij} = 1$ (S_i と S_j が隣接する機器であり、 S_i は S_j の上流にある場合)

$a_{ij} = 0$ (S_i と S_j が隣接していないか、 S_i が S_j の下流にある場合)

である。ここで集合 S をプラントに存在する機器 S_i の全体集合とする。これを基に、グラフ理論を利用して以下のようにして操作の効果の有無を評価する。まず、隣接行列 $A+I$ に対して、可到達行列を

$$(A+I)^{n-1} \quad (1)$$

により求める。ここで n は隣接行列の次元（プラントの機器数）である。そして、可到達行列の第 i 行 j 列要素を m_{ij} とすると、

$$A_i = \{ S_j \in S \mid m_{ji} = 1 \} \quad (2)$$

となる A_i は先行集合を求める。先行集合は、機器 S_j に質量やエネルギーの流入を行う操作効果の現れる機器の位置の範囲を表しており、これから対応操作が回復したいプラント挙動に対して効果があるかどうか分かる。

対応操作候補の絞り込みにおけるそれぞれの候補の優先順位は以下のように決定する。まず、運転手順や操作の難易性から操作性の評価の高いものを上位とする。次に、経験的な操作の影響に関する知識により優先づけを行い、最後

にプラント構成機器の作動流体の流れの位置による効果の可能性の有無に対する評価結果を考慮に入れる。

表 2.7-2に対応操作候補の絞り込みの適応例の一部を示す。ナフサ抜き出しポンプの性能低下による異常事象に対して、リフラックスドラムの液位上昇を回復するための導出された対応操作候補の絞り込みを行ったものである。欄Aは操作性、欄Bは通常性、欄Cは操作効果の経験的知識による評価結果であり、欄Dはプラント構成機器の作動流体の流れの位置による評価結果である。評価欄の記号の意味は、○（高い）、△（中程度）、×（低い）、？（不明）である。絞り込みにより、原油供給量減少、加熱炉各パス流量減少、加熱炉燃料供給量の減少の3つの対応操作候補の優先順位が高い結果となった。

表 2.7-2対応操作候補絞り込み結果例

	導出対応操作候補	A	B	C	D
1	ナフサ抜き出しポンプの性能回復（異常原因の回復）	×	○	○	○
2	原油供給量減少	○	○	○	○
3	加熱炉各パス流量減少	○	○	○	○
4	#1ストリッパへの抜き出し量増加	○	○	○	×
5	加熱炉燃料供給量減少	○	○	○	○
6	ナフサ抜き出しバルブ開度増加	○	△	○	×
7	常圧蒸留装置の注入スチーム流量減少	×	△	○	○
8	熱交換器（E51）冷却水流量減少	○	×	？	×
9	#1ストリッパの注入スチーム流量減少	○	×	？	×
・	・ ・ ・ ・ ・	・	・	・	・

（２）高速シミュレータの開発と評価

平成11年度に一応の完成を見た高速シミュレータを用いていくつかの運転状況に対して計算されるプラント状態を、日石三菱殿の所有する詳細な数値シミュレータによる計算結果と比較することにより、高速シミュレータの各部のパラメータのチューニングを行うとともに、高速シミュレータの適用性を評価した。

高速シミュレータの構成は、昨年度の計算体系に加えて、リフラックスドラムや各種の熱交換器、ポンプ、バルブを追加し、図 2.7-2の計算体系としている。これにより、対象とするプラントのほぼすべての機器が模擬されている。

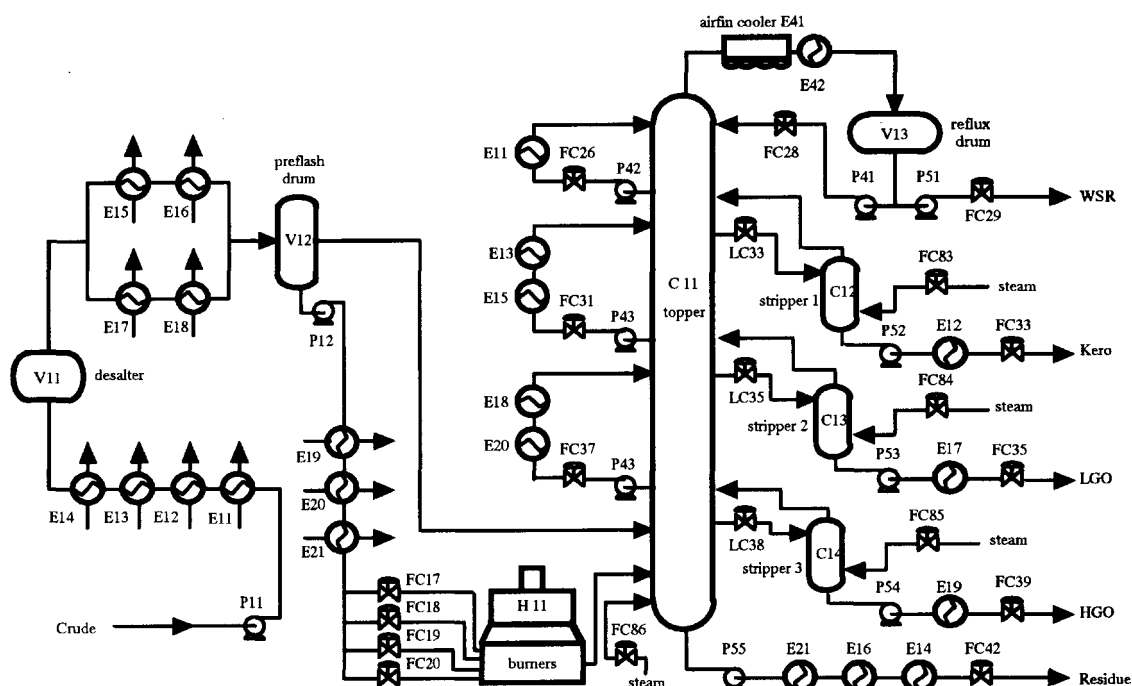


図 2.7-2高速シミュレータの計算体系

高速シミュレータにおける数値計算方法は、基本的には実プラントと同じように原油の流れに沿って計算する。しかし、図 2.7-2に示すように原油の上流と下流が熱交換器によって接続されており、下流の状態変化が上流の状態へ影響を及ぼすために収束計算する必要がある。具体的には、計算して求まった熱交換器の下流側の流量の値を上流側の値に用いてもう一度計算し、その値がある一定値に収束するまで繰り返す。収束条件は、加熱炉温度、蒸留塔の内部状態や熱交換などの影響を受けて変化するナフサ流量と廃油流量の計算結果の相対誤差が0.1 %以下になるまでとした。

適用性評価に先立ち、高速シミュレータにおける各コンポーネントの比熱や潜熱などのパラメータ値を、正常運転時のプラント状態の計算値を日石三菱殿が所有する PROVISION (静的シミュレータ) のシミュレーション結果と比較することによりチューニングした。

高速シミュレータの評価は、日石三菱殿が使用しているダイナミックシミュレータ (MSAT) による計算結果と比較することで行った。評価の一例として、プラントの正常運転状態より、原油張込量を変化させた時のナフサ流量、残油流量、加熱炉出口温度、および、トップリフラックス流量の変化割合を比較した場合を表 2.7-3に示す。なお、この計算においては、プラント全体としての

マスバランスをとるため、原油張込量の変化量と同じ割合だけ各ストリッパへの抜き出し量も変化させた。表からわかるように、各状態変数の変化量に多少の違いがあるが変化の傾向は一致していることがわかる。

表 2.7-3原油張込量と製品抜き出し量をバランスさせながら変化させた場合の MSAT による計算結果との比較

	原油張込量 (%)	95	90
ナフサ流量 (定格値との割合 %)	高速シミュレータ	99.5	98.5
	MSAT	94.9	88.7
残油流量 (定格値との割合 %)	高速シミュレータ	86.4	73.1
	MSAT	92.0	90.5
加熱炉出口温度 (定格値との割合 %)	高速シミュレータ	102.6	105.1
	MSAT	102.6	105.9
トップリフラックス流量 (定格値との割合 %)	高速シミュレータ	120.0	139.7
	MSAT	107.3	115.3

評価結果より、高速シミュレータはダイナミックシミュレータ MSAT とほぼ同じ傾向を示しており、簡易シミュレータとして利用できることが明らかとなった。

(3) 高速シミュレータを用いた対応操作候補の評価

絞り込まれた対応操作候補の操作量や効果は高速シミュレータを用いて求める。対応操作候補の効果の評価は以下の処理流れに従って行う。操作量は評価関数が最大となるように求める。評価関数の設定には、1. プラント状態の回復、2. 対応操作候補導出部での戦略的知識部の整合、3. プラント危険回避運転を考慮し、異常原因毎にあらかじめ設定している。

対応操作効果の評価のためのアルゴリズムを以下に説明する。

(Step1) 絞り込まれた対応操作候補を1つ選ぶ。対応操作候補がない場合は評価を終了する。

(Step2) 選ばれた対応操作候補に対する評価関数を選ぶ。そして、評価に必要なプラント計装信号の現在値をインタフェースエージェントに問い合わせる。

取得する。

(Step3) 定常運転時におけるプラント状態変数の値を初期値とし、ある操作量で対応操作を行った時のプラント状態の漸近値を高速シミュレータを用いて予測する。

(Step4) Step2 で選択された評価関数を用いて操作量の適切性を評価する。

(Step5) 操作量が適切な場合には、次の対応操作候補を評価するため Step1 に戻る。そうでない場合には、操作量を変更し Step3 に戻る。なお、操作量は正常運転値から 1 % ずつ増減させるものとする。

対応操作候補の操作量の評価例として、ナフサ抜き出しポンプの性能低下によるリフラックスドラムの液位上昇に対して求めた操作量を表 2.7-4 に示す。リフラックスドラムの液位を回復するために、リフラックスドラムで分離されるナフサ流量が FC29 流量（抜き出し弁流量）の現在値となる操作量を求めている。これにより、少なくとも液位上昇は回避される。ただし、対応操作候補 1 と 2 では、原油の張り込み量や加熱炉入口流量のみを減少させただけではプラントの運転状態（特に、加熱炉出口温度）が変化するので、加熱炉での加熱量も同じ割合で減少させる対応操作としている。このような同時に行うべき操作は、あらかじめ知識ベース化して用いている。

表 2.7-4 対応操作評価結果例

異常原因	ナフサ抜き出しポンプの性能低下		
評価関数	リフラックスドラムで分離されるナフサ成分流量を現在値とする 評価に必要なプラント計装信号：FC29 SP, FC29 PV		
対応操作候補	1. 原油供給量低下（加熱炉燃料供給量低下）	2. 加熱炉バス流量低下（加熱炉燃料供給量低下）	3. 加熱炉燃料供給量低下
対応操作評価結果	原油供給量 543.3 Kl/Hr 加熱炉燃料供給量 3664.3 Nm3/Hr P42 ポンピング流量 385.4 Kl/Hr P43 ポンピング	加熱炉バス流量 72.5 Kl/(Hr・バス) 加熱炉燃料供給量 2055.6 Nm3/Hr P42 ポンピング流量 216.2 Kl/Hr	加熱炉燃料供給量 3664.3 Nm3/Hr

	流量 102.5 Kl/Hr P44 ポンピング 流量 127.1 Kl/Hr	P43 ポンピング 流量 57.5 Kl/Hr P44 ポンピング 流量 71.3 Kl/Hr	
--	---	--	--

2.7.3. 表示意図に基づく意味表示の実装

平成11年度検討した表示意図に基づく意味表示手法を機能－目標レイヤーの可変詳細度表示として実装した。すなわち、正常時には原油から各製品成分が蒸留される質量の流れ部分のみを詳細表示としその他は簡略表示としているが、異常発生時や異常原因推定時には機能モデルの関連部分を自動的に詳細表示するようにした。図 2.7-3に正常運転時の機能－目標レイヤーを示し、図 2.7-4にリフラックスドラムの液位上昇として異常が検知され、その後、ナフサ抜き出しポンプの性能低下という異常原因が同定された時の機能－目標レイヤーを示す。図からわかるように、正常時には運転員は注意をあまり払う必要がないために簡略表示されている部分が、異常検知や異常原因に関連するために自動的に詳細表示に変更されている。

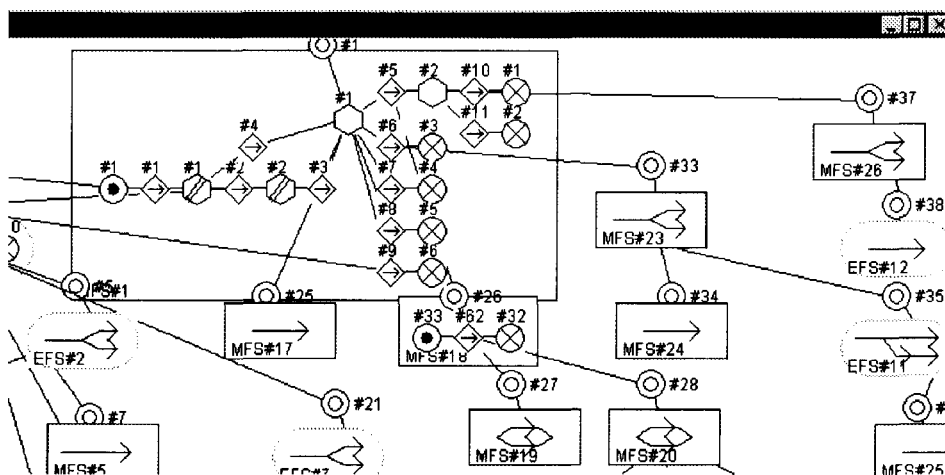


図 2.7-3 正常時における機能－目標レイヤーの表示

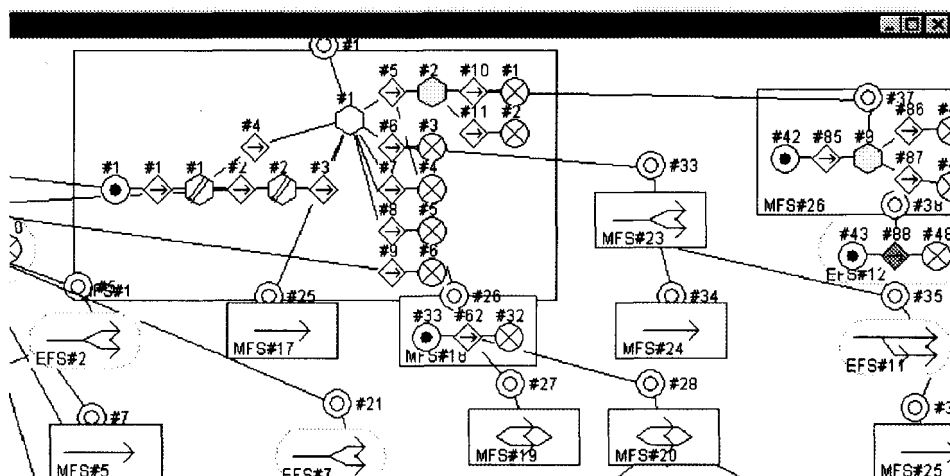


図 2.7-4異常検知と異常原因が同定された時の機能－目標レイヤーの表示例

2.7.4. 最終プロトタイプシステムの開発

大画面表示や追加シナリオに対応した最終プロトタイプシステムを開発し、分散協調環境下の実装した。最終プロトタイプシステムを効率的に開発するため、分散協調システムの利点を活かして、意味表示インタフェースを2つの知的エージェントに分割した。それらは、(i) 異常原因推定部、対応操作候補導出部、対応操作評価部から成る部分 (SIA1) と、(ii) トレンドグラフ、近未来グラフやプラント状態のグラフを表示するグラフ表示部 (SIA2) である。SIA1 の画面例を図 2.7-5に示す。最終プロトタイプシステムで実行できるタスクは、平成12年度開発したシステムと基本的には同じである。ただし、高速シミュレータを用いて対応操作の操作量を計算する、対応操作評価タスクを追加した。SIA2 で表示するグラフの内容を読みとり易くするため、グラフ表示サブプログラムを新たに実装した。なお、このグラフ表示サブプログラムは大画面エージェントでも利用されている。

- ・操作が可能な機器やその操作を実施した時の機能－目標モデルへの影響、対応操作の導出戦略を決める際に必要となるプラントの危険な状態に関する知識、
 - ・プラント機器名と構造モデルや機能－目標モデル内のシンボルとの対応、
- そして、
- ・ポンプ特性曲線データ
- である。

Java 版プラントモデル構築ツールの整備により、新規のプラントに対してやプラント機器構成が変更になった場合のプラントモデルの構築が効率的に行えるようになった。なお、本プラントモデル構築ツールを発展させて、本年度研究開発した異常原因推定手法のためのプラントモデルの構築をサポートするようにすれば、意味表示インタフェースがベースとするプラントモデルはほぼこのプラントモデル構築ツールによりカバーできることになる。

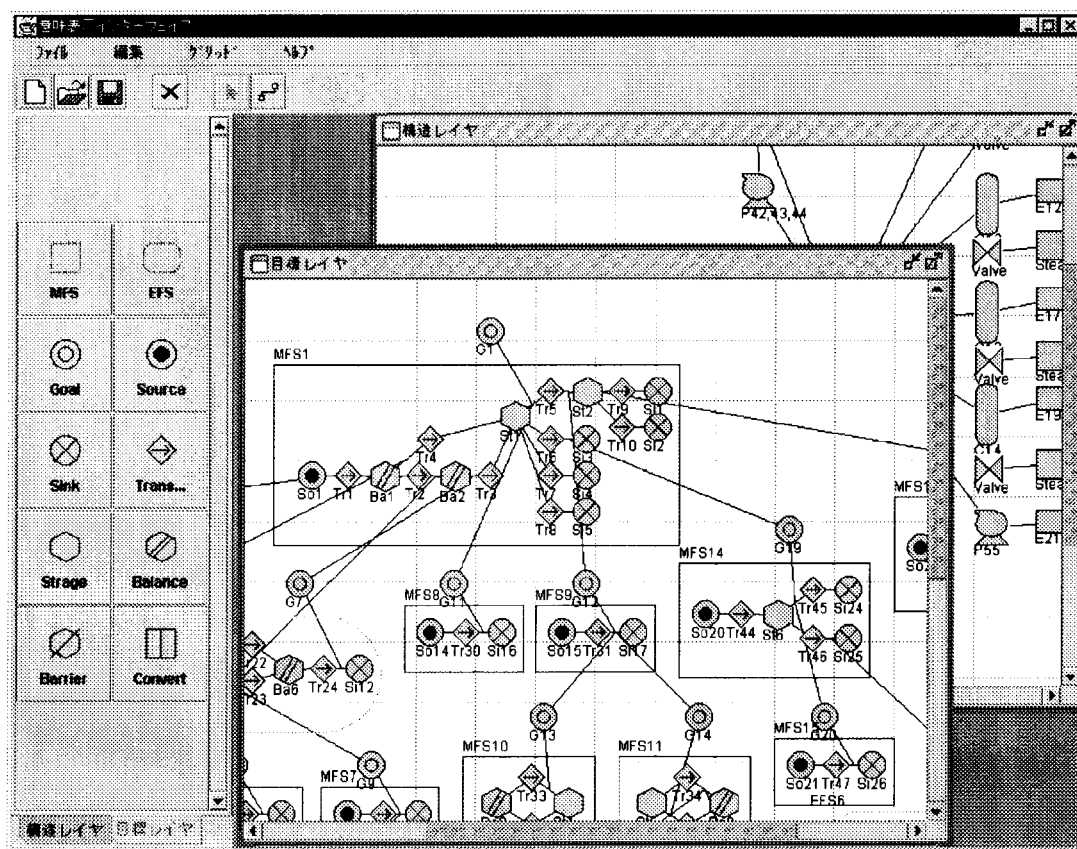


図 2.7-6プラントモデル構築ツール画面例

2.8. 大画面統合表示システム構築のためのガイドライン作成

「ヒューマンメディアの研究開発（石油プラント高度情報化システム技術研究開発）」（「次世代プラント用ヒューマンインタフェースの研究開発」）プロジェクトでは、プラント運転システムのインタフェース部の高度化を目的として、運転に必要な情報をプラント状況に合わせて統合的に提示し、運転員が直感的に情報を理解したり、プラント状況を素早く的確に把握できるような次世代のプラント運転インタフェースの要素技術を開発し、プロトタイプを実現する計画を進めている。

そこで、「大画面統合表示システム構築のためのガイドライン作成」では、各種シナリオに対する大画面統合表示パターンを検討し、大画面統合表示パターンのロジックを策定すると共に、大画面統合表示インタフェースのコンセプトモデルを策定し、次世代プラント用ヒューマンインタフェースのシステム構築に資する大画面統合表示のためのガイドラインを作成した。

2.8.1. プラント運転における各種異常シナリオの調査・分析

既存の9つの異常シナリオに現れる全100ステップを調査・分析し、大画面統合表示パターンを洗い出し、タイリング・パターン（TP）として分析・分類しとりまとめたものを表2.8-1に示す。

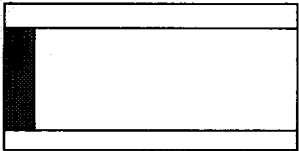
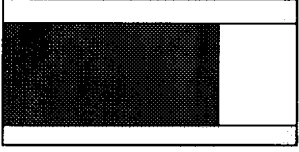
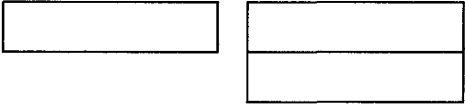
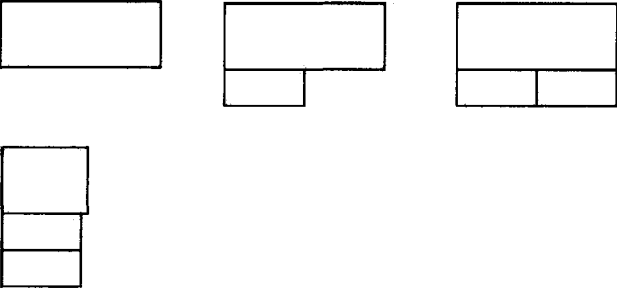
（ただし、表の中の略号は、

- ・TP : タイリング・パターン
- ・W : 全体ウインドウ
- ・O : 「操作・メッセージ表示領域」のペイン
- ・P : 「予測・状態表示領域」のペイン
- ・TP_W_i : 全体ウインドウのi番目のタイリング・パターン
- ・TP_O_j : 操作・メッセージ表示領域ペインのj番目のタイリング・パターン

ン

・TP_P_k : 予測・状態表示領域ペインのk番目のタイリング・パターンを表している。）

表 2.8-1 9シナリオの表示パターン

項目	表示パターン	対応ステップ／表示情報
(1) 全体ウィンドウ (TP_W_i)	TP_W_1  TP_W_2  TP_W_3 	TP_W_1 - 正常運転中 - 異常検知 - 異常通知 - 現在状態提示 - 近未来状態提示 - 異常リセット通知 TP_W_2 - 異常原因候補提示 - 異常原因提示 TP_W_3 - 対応操作候補提示 - オペレータによる 操作手順表示要求 - オペレータによる 対応操作導出要求 - 操作手順表示 - オペレータによる 対応操作 - 確認項目提示
(2) O領域ペイン (TP_O_j)	TP_O_1 TP_O_2  TP_O_3 TP_O_4 	- 異常原因候補検索結果 - 異常原因推定結果 - 過去の対応操作検索結果 - 対応操作候補導出結果 - 操作手順検索結果 - 操作手順
(3) P領域ペイン (TP_P_k)	TP_P_1 TP_P_2 TP_P_3 TP_P_4 	- プラント系統図 - トレンドグラフ - 近未来グラフ

2.8.2.大画面統合表示パターンのロジックの策定

2.8.2.1. タイリング制御の一般的指針

次世代プラント用ヒューマンインタフェースにおいては、人と機械システムが協調して問題を解決していくために提供される、人を中心に考える設計を指向してデザインされた、プロセスを共有できる環境 HOPE_cps (Human-centered Oriented Processes Environments for Collaborative Problem Solving) を構築する。

以下では、この HOPE_cps 環境における情報内容の提示に関するウインドウ・タイリング制御の一般的指針を示す。

タイリング制御の一般的指針は以下の3つの指針から構成される。

- (1) 基本指針
- (2) 応用指針
- (3) 表示優先度の評価項目と処理方式の指針

2.8.2.1.1. 基本指針

表 2.8-2に、オペレータからの表示介入を含まない、システム側からの情報内容の提示に関するウインドウ・タイリング制御の基本指針を示す。

表 2.8-2タイリング制御の指針（基本指針）

No.	項目	指針
1	HOPE_cps(Human-centered Oriented Processes Environments for Collaborative Problem Solving) におけるタイリングのデザイン・プリンシプル	(1) HOPE_cps では、人間の広義の問題処理機構に可能な限り適応したコンセプト・モデルを考える。 (2) ここでは、人間の • 全体抽象化把握、 • Operation Logic、 • Prediction、 • Attention、 • Imagination の機能に相応するものとして、大画面全体の構成を • T (タイトル) 表示領域、 • O (操作・メッセージ) 表示領域、 • P (予測・状態) 表示領域、 • A (警報メッセージ) 表示領域、 • V (仮想プラント) 表示領域 の各ウインドウの組み合わせで表出することとし、これらの表示領域の各々の中に各サブシステムからの出力を意味に応じて分類し、その各々をペインとしてはめ込むタイリング方式とする。

		これらをあわせた統合表示システムのタイリング方式を、ここでは「TOPA-V ペイン方式」と呼ぶ。 (3) 今後の説明では、例えば下記の例に示す表記法を用いる。 (なお、サフィックスの b、m、s、n は各々 big、medium、small、normal を表す。) 例： $TP5::=Tn(1) \cdot Ob(2) \cdot Ps(3) \cdot An(1) \cdot Vn(1)$ • TP5：タイリング・パターン 5 • Tn(1)：タイトル表示領域全体は原則 normal サイズで、1つのペインのみで構成される。 • Ob(2)：操作・メッセージ領域全体は big ウィンドウで、2つのペインから構成される。 • Ps(3)：予測・状態領域全体は small ウィンドウで、3つのペインから構成される。 • An(1)：警報メッセージ領域全体は通常 normal ウィンドウで、1つのペインから構成される。 • Vn(1)：仮想プラント領域全体は normal ウィンドウで、1つのペインから構成される。
2	各ウィンドウならびに各ペインの出現・縮小・消滅に関するタイリング制御の基本原則	(1) まず基本的に、到着順、画面滞在時間、許容ペイン数によって下記のルールに従ったタイリング制御を行う。 • LIBA (Last_In_Big_Appearence) • LIUA (Last_In_Upper_Appearence) • PESA (Pre_Existence_Small_Appearence) • LEFO (Long_Existence_(over_limited_number)_First_Out) (2) エリア内 (ウィンドウ内) における許容ペイン数は仮に 3 とする。 大画面全体としてのペイン数は、7 プラス・マイナス 2 の原則に従うことにする。 (3) 警報メッセージは、常に最新到着のものを表示し、オペレータの操作により、過去のメッセージが表示できるようにしておく。 (4) なお、各サブシステムからの出力情報に対して重要性・信頼性・緊急性の属性を設定できる場合には、下記のルールに従ったタイリング制御を行う。 • IIUA (Importance_In_Upper_Appearence) • RIUA (Reliable_In_Upper_Appearence) • PEBA (Pre_Existence_Big_Appearence) • LEOO (Long_Existence_(Over_limited_number)_Operator_drives_Out)
3	フェーズ 3 で考えられるタイリング・パターン TP	基本タイリング・パターンは下記の 5 つ： • $TP0::=Tn(1) \cdot Os(0) \cdot Pb(1) \cdot An(1) \cdot Vn(1)$ • $TP1::=Tn(1) \cdot Os(0) \cdot Pb(2) \cdot An(1) \cdot Vn(1)$ • $TP2::=Tn(1) \cdot Os(0) \cdot Pb(3) \cdot An(1) \cdot Vn(1)$ • $TP3::=Tn(1) \cdot Om(1) \cdot Pm(3) \cdot An(1) \cdot Vn(1)$ • $TP4::=Tn(1) \cdot Ob(2) \cdot Ps(3) \cdot An(1) \cdot Vn(1)$ • $TP5::=Tn(1) \cdot Ob(3) \cdot Ps(3) \cdot An(1) \cdot Vn(1)$
4	各表示領域ウィンドウの中で	O (操作・メッセージ表示) 領域と P (予測・状態表示) 領域の中

のペインのタイリング・パターン	のペインのタイリングパターンとしては、フェーズ3では、表1の(2)と(3)にある下記の4種類とする。 • TP_O_j (j=1~4) • TP_P_k (k=1~4)
-----------------	---

2.8.2.1.2. 応用指針

この応用指針では、

- ・ オペレータの表示介入
- ・ 順次・複数異常通知

があった場合に対するタイリング制御指針を表 2.8-3示す。

表 2.8-3タイリング制御の指針（応用指針）

No.	項目	指針原案
1	オペレータの表示介入に対するタイリング制御	(1) いくつかのステップまではシナリオ通りに動作している状態で、オペレータ要求があった場合にはその表示を優先 big_pane 表示をする。 • ORBA (Operator_Request_Big_Appearance) (例：第6番目のステップの異常原因表示の状態で、第4番目のステップの近未来状態表示をオペレータが要求する。) (2) 前述の基本指針に従った上で、下記のルールに従ったタイリング制御を行う。 • LISO (Last_In_Same_Out)
2	順次・複数異常通知発生に対するタイリング制御	(1) フェーズ3では、複数箇所の故障は想定しないが、異常通知が次々に発生し、現在状態提示の要求が複数起こる場合を想定する。 (例：一つのシナリオのステップが進んでいる段階で、シナリオ上は以前のステップである異常通知と現在状態提示が複数起こり、画面内に同時表示することが困難となる状況。) (2) 上述までの事象への対応と同様に、下記のルールに従ったタイリング制御を行う。 • LIBA (Last_In_Big_Appearance) • PESA (Pre_Existence_Small_Appearance) • LEFO (Long_Existence_(over_limited_number)_First_Out) • LISO (Last_In_Same_Out) (3) 各ペインの基本的扱いを以下にします。 • トレンドグラフは、1つの図の中に新たな曲線を追加することを原則とする。(ただし、フェーズ3における処理法については、別途検討する。) • 近未来グラフは、1つの図の中には1つの曲線しかないので、複数の近未来グラフが出現することが想定される。 • プラント系統図は同時に現れるものは1つとする。

2.8.2.1.3. 表示優先度の評価項目と処理方式の指針

表示優先度の評価に関わる事項と処理方式の指針を表 2.8-4に示す。

表 2.8-4表示優先度の評価項目と処理方式の指針

評価項目	内容	処理方式概要	指針
新鮮度 (F)	時間的な 新しさ	(1) 新鮮度の順にペインの場所は上(左)→下(右)に配置する。 (2) 新鮮度の順にペインのサイズは大→小にステップ的に変化させる。 (3) すでに同種のペインが存在する場合には、前のものを削除する。 (4) 表示限界数を越えた場合には、長く滞在するものから消去する。	(1) LIUA (2) LIBA PESA (3) LISO (4) LEFO
緊急度 (U)	処理の制 約時間	(1) 基本は新鮮度の処理方式に準じる。 (2) 次のペインが入力されても、該当ペインのサイズは小さくしない。 (3) 該当ペインの削除はオペレータ操作に従う。 (4) フェーズ3では、緊急度は扱わない。	(1) --- (2) PEBA (3) LEDO
重要度 (I)	障害の影 響度合い	(1) 基本は新鮮度の処理方式に準じる。 (2) 次のペインが入力されても、該当ペインのサイズは小さくしない。 (3) フェーズ3では、重要度は一定値とする。	(1) --- (2) PEBA
信頼度 (R)	提供情報 の信頼性	(1) 基本は新鮮度の処理方式に準じる。 (2) 信頼度の順にペインの場所は上(左)→下(右)に配置する。	(1) --- (2) RIUA

総合優先度は、

・ 新鮮度(F)>緊急度(U)>重要度(I)>信頼度(R)

の順番とする。

処理方式の概要を図 2.8-1に示す。

		a. オペレータの注意を引く b. 問題解決プロセスの説明 c. オペレータに操作を促す に厳選して提供する。 d. 内容の説明 に関し、複数の類似内容のものを説明する場合には、 ・最初のものみの説明 ・もしくは、まとめた表現での説明 を行う。 (3) 提示タイミングに関し、既に提示中の音声メッセージ情報に対する ・「続行」(C : continue) ・「中止」(S : stop) ・「中断」(I : interrupt) ・「再開」(R : restart) の各処理の選択判断は、原則として問題解決プロセスにおける有用性によりおこう。 (4) 複数情報に対する ・「同期提示」(SP : synchronous presentation) の処理の判断は、原則として問題解決プロセスにおける有用性によりおこなう。 (5) 音声メッセージで提示される情報の内容は、原則として大画面に表示される情報の内容にも含まれていることを前提とする。
2	「続行」処理の原則	(1) 異常発生通知をしているときには、該当通知のメッセージが終了するまで処理を「続行」する。 (2) 異常発生通知メッセージの最大継続時間を規定しておく。
3	「中止」処理の原則	(1) システム側からの情報による音声メッセージを提示中に、緊急度が高いメッセージを受け付けたときには、提示中の該当メッセージの提示を「中止」し、緊急度の高いメッセージに切り替えて提示する。
4	「中断」処理の原則	(1) オペレータ要求に対する音声メッセージを提示中に、緊急度が高いメッセージを受け付けたときには、提示中の該当メッセージの提示を「中断」し、緊急度の高いメッセージに切り替えて提示する。
5	「再開」処理の原則	(1) 「中断」処理されているメッセージを「再開」処理をするときには、中断された該当メッセージの最初から提示する。
6	「同期提示」の原則	(1) 「プラント状態表示領域」と「VP 表示領域」で、同種の提示がある場合には、「同期提示」を原則に表示する。

2.8.3.大画面統合表示インタフェースのコンセプトモデルの策定

ヒューマンメディア研究に資するための、次世代プラント用の大画面統合表示インタフェースのコンセプトモデル：「Operator/Co-operator モデル」を以下に示す。

2.8.3.1. 「Operator/Co-operator モデル」の構成

次世代プラントにおけるヒューマンインタフェース設計の基本コンセプトである「Operator/Co-operator モデル」の構成を以下に示す。

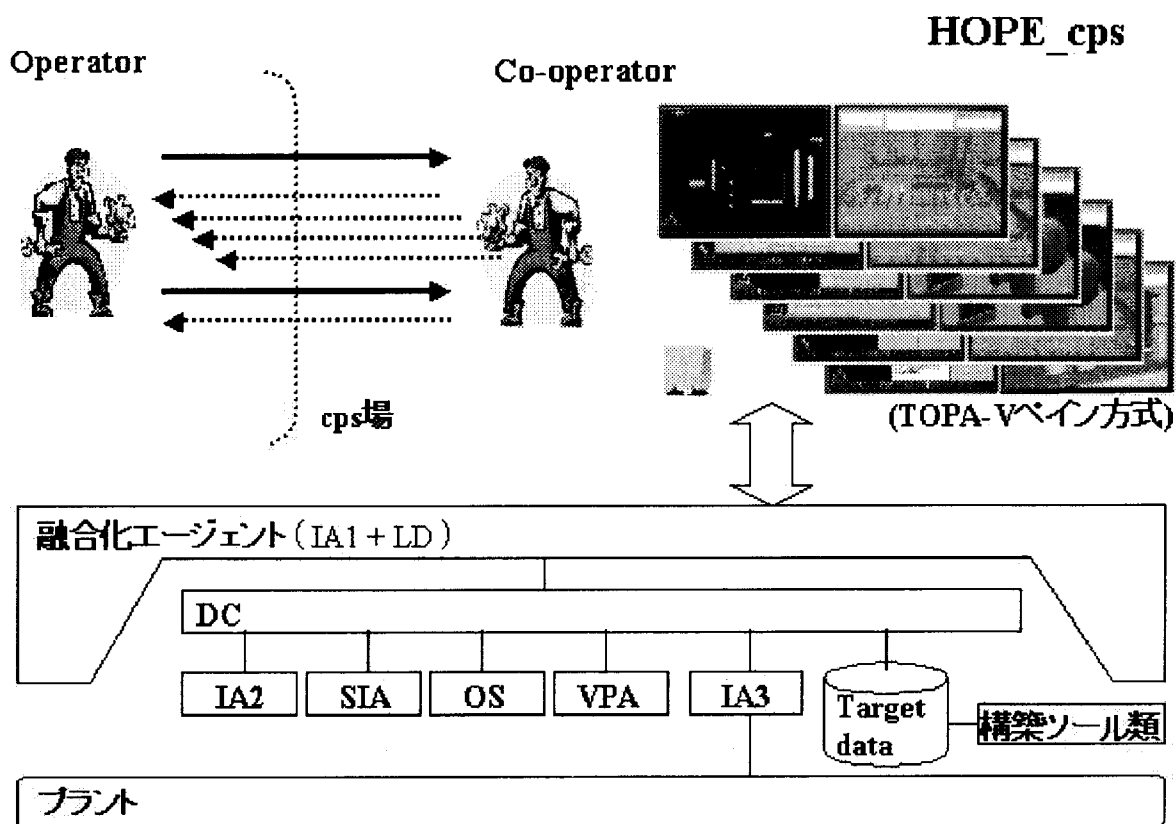


図 2.8-2 「Operator/Co-operator モデル」のイメージ図

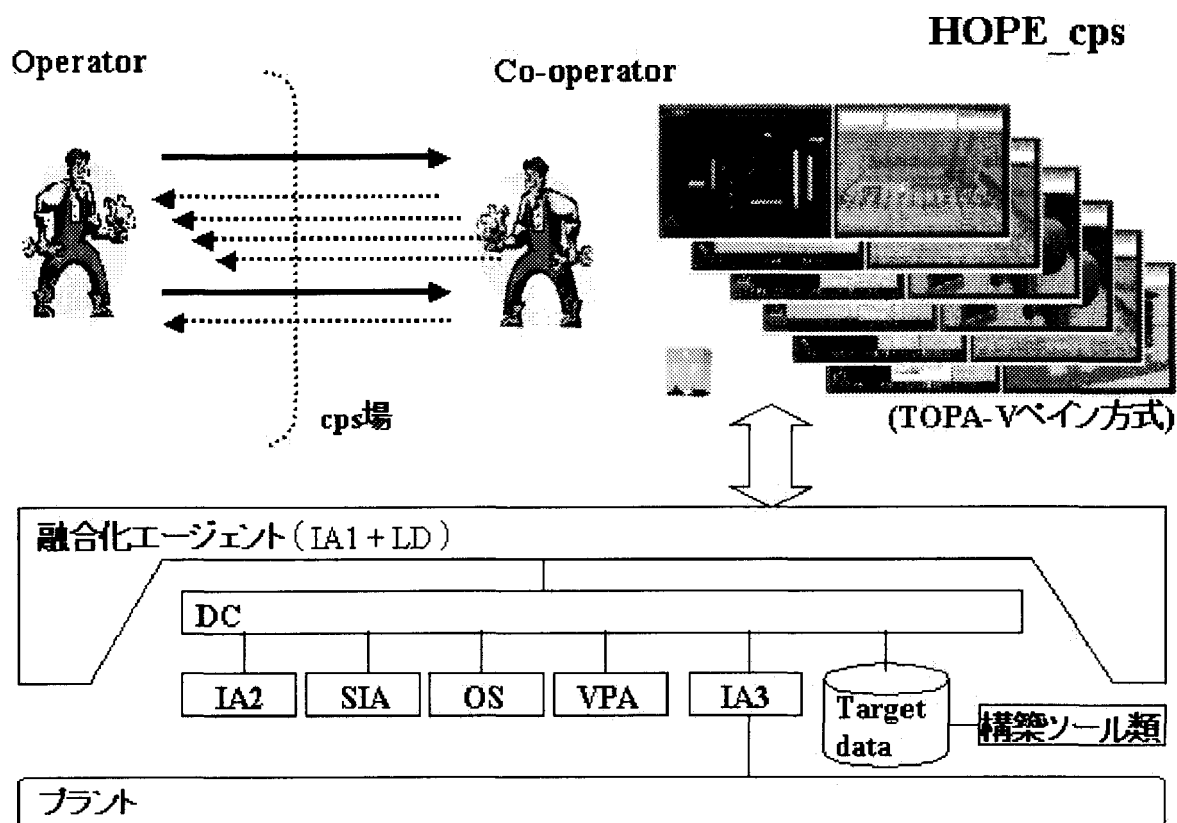


図 2.8-2の中の略称の説明、「Operator/Co-operator モデル」の構成とその概要を表 2.8-6に示す。

表 2.8-6 「Operator/Co-operator モデル」の構成とその概要

No.	名称	内容概要
1	Operator / Co-operator モデル	人である Operator と、機械システムである Co-operator が協調して問題解決をおこなうモデル。
2	cps 場	cps とは collaborative problem Solving (協調問題解決) の略で、cps 場とは、人と機械システムが協調して問題を解決していく場。
3	HOPE_cps	HOPE_cps とは、Human-centred Oriented Process Environment for collaborative problem solving (協調問題解決のための人間中心指向プロセス環境) の略で、人と機械システムが協調して問題を解決していくために提供される、人を中心に考える設計を指向してデザインされた、プロセスを共有できる環境。

4	TOPA-V ペイン方式	プラントシステムにおいて、人と機械システムが協調して問題を解決していくためのプロセス共有の環境のひとつとして大画面ディスプレイによる環境を提供する。TOPA-V ペイン方式とは、この大画面ディスプレイのダイナミックなタイリング表示方式であり、T、O、P、A、V とは、各々； T： 問題解決プロセスのタイトルの表示領域 O： オペレーション・判断参考情報の表示領域 P： 予測・システム状態の表示領域 A： アラームメッセージの表示領域 V： 仮想プラントの表示領域 を表し、これらの表示領域を統合的に組合わせて、問題解決のためのダイナミックなプロセス環境として提供する。
5	融合化エージェント	以下の様々なエージェント類、インタフェース類、サーバ類等の機能を融合化・制御し、人と機械システムが協調して問題を解決していくためのプロセス共有の環境を提供するエージェント。 Co-operator を構成する要素のひとつであり、主に人の総合化・外界適合化の機能に相当する。
6	DC	分散協調通信サーバ。 Co-operator を構成する要素のひとつであり、主に人の情報収集・分配・伝達などの機能に相当する。
7	IA	インタフェース・エージェント。 インタフェース・エージェントは、以下の3つに分類される。 • 人とのインタフェース (IA1) • 過去の経験を含めた機械システムの中でのインタフェース (IA2) • プラント・システムとのインタフェース (IA3) Co-operator を構成する要素のひとつであり、主に人の様々なインタフェース機能に相当する。
8	SIA	意味表示インタフェース。 Co-operator を構成する要素のひとつであり、主に人の予測・表現変換の機能に相当する。
9	OS	オントロジー・サーバ。 Co-operator を構成する要素のひとつであり、主に人の言語・知識蓄積の機能に相当する。
10	VPA	仮想プラント・エージェント。 Co-operator を構成する要素のひとつであり、主に人の想像・イメージ化の機能に相当する。

2.8.3.2. 「Operator/Co-operator モデル」の基本となる認知的視点

2.8.3.2.1. 問題解決と問題点

(1) 問題解決

人間の認知過程をとらえる方法として、人間の情報処理機構に注目し、内容から独立した認知過程とその制約条件を明らかにしようとするアプローチに対して、

特定の内容課題を遂行する過程を問題解決としてとらえ、その過程を、

- ・「人がどういう問題として認識しているか」
- ・「どういう方法で解こうとするか」
- ・「そこに知識がどう影響するか」

などを区別して記述しモデル化するアプローチである。

問題解決のプロセスは、

- ・解釈・表現 (representing)
- ・プランニング (planning)
- ・実行 (executing)
- ・モニタリング (monitoring)

の4つの相互に関係したプロセスとして分析できる。

問題解決のプロセスは、問題解決者がその課題にどんな知識を持ち込んでくるかに依存する。

- ・問題に関するスキーマ的な知識
- ・解決法に関する方略的な知識
- ・条件の判断と行為を生じさせるルールに関する知識
- ・行為の完結に関する知識
- ・問題解決をどのように制御するかの知識
- ・問題に対する信念

などが影響する。

その影響は問題の解き方やその効率にとどまらず、投入する資源の量にも関係する。

(2) 問題解決の枠組みの問題点

人は最初に決めたプランを忠実に実行するとは限らない。状況に応じて、システムやその他のツールの利用可能性に応じて、プランも実行手順も、ときには目標さえも柔軟に変化させる。

外部から与えられ、問題解決者の頭の中に表現された問題と彼らの知識に閉じた形での問題解決の枠組みでは、人間が外部環境と相互作用しながら、環境に

遍在するリソースを適宜活用して問題を解くダイナミックな活動をとらえきれない。

こうした問題点から、例えば、ユーザの行動を従来の問題解決ではなく、ユーザの意図とシステムとの知的な対話として捉え、対話分析としてのインタフェース研究が展開されている。

特に、状況を重視し、ユーザとシステムの両者の状況・行為・意図に基づく分析が必要になる。これらは、ユーザが個人で問題に取り組む場合だけでなく、幾人かで共同で目標行動を行うときの分析にも有効である。

2.8.3.2.2. 目的的な活動における認知とインタフェースの相互作用

目的的な活動における認知とインタフェースの相互作用に関するモデルとして、以下の7つの段階からなる「Normanの7段階モデル」がある。

1. システムに向かい、何をするかの意図の形成
2. 具体的な下位課題を特定し、実行計画を立てる
3. 下位課題遂行の意図をもってシステムの画面に向かい、
その画面のどこをどうするかを理解する
4. 実行する
5. 実行の結果をシステムの状態変化を知覚することによって知る
6. システムの状態変化を解釈して、行為達成を判断する
7. 行為の結果を、全体的な意図と比べて順調かどうかを判断する

2.8.3.2.3. オブジェクト毎の認知的着眼点

画面の中のオブジェクト毎の認知的着眼点としては、下記のものあげられる。

- ・画面の中での探し易さ
(内的な計画とのマッピング)
- ・画面の中での分かり易さ
(操作のための理解、結果のための理解)
- ・操作し易さ
(操作自体のし易さ、操作フィードバックの知覚分かり易さ、
操作フィードバックの解釈し易さ)
- ・全体的なモニターのし易さ

2.8.3.2.4. 次世代プラント・ヒューマンインタフェースへの適用

上述の認知的視点を考慮し、まず、全体のコンセプトとして、人である Operator と、機械システムである Co-operator が協調して問題解決をおこな

う「Operator/Co-operator モデル」を基本にすることにした。

このモデルを実現するために、問題解決のプロセスに応じて大画面ディスプレイに必要な情報の提示をおこなうプロセス環境として「HOPE_cps」を、大画面のタイリング表示方式として「TOPA-V ペイン方式」を、それぞれ提供している。

2.8.3.3. ヒューマンメディアシステム参照モデルとの関係

ヒューマンメディアシステム参照モデルとの関係を融合化の視点より、とりまとめたものを表 2.8-7に示す。

表 2.8-7ヒューマンメディアにおける融合化の内容

HM システム参照モデル [要素技術の融合のレベル]	次世代プラント (HM の適用シーン)	人間中心の情報環境
	利用者とどんな関係を持つのか (surround)	利用者にどんな影響を与えるのか (affect)
論理世界、 物理世界、 心理世界	大画面の全体表示では、 - 現実の物理世界をシミュレートした仮想プラントの表示 - 論理世界であるシステムデータ、アラームデータ、事例ベース、推論結果の表示 - 心理的世界である運転員（オペレータ）の想像世界の予測・意味表示と、問題解決心理過程のプロセス表示 を行い、ヒューマンメディアの各要素技術を融合化した表示環境を提供している。 （「 TOPA-V ペイン方式 」と呼んでいる。）	運転員（オペレータ）の問題解決プロセスを支援する。 人である運転員（オペレータ）と、HM が融合されている PHM システムが運転員にとってのコ・オペレータとして協働して問題解決をおこなうというヒューマンインタフェースの基本コンセプトを「 Operator/Co-operator モデル 」と呼んでいる。 [参考] （1）既提出の各コンポーネント名の中で、インタフェースエージェントのコンポーネント名として新たに IA1 を追加提出すると、そ

論理世界、 物理世界	大画面統合表示方式として、システムの論理世界を含めて、運転員（オペレータ）の問題解決の心理的過程に応じて、大画面表示構成・遷移をダイナミックに適応させるというプロセス環境を提供している。 （この HM の融合化されたプロセス環境を「HOPE_cps」と呼んでいる。）	れが融合化技術のコンポーネントになると考えられる。 これは、現在の参照モデルの図の中では、「モデル管理」の箱に相当すると思われる。 （２）PHM にとっての感性の例としては、VPA 中の「製品性状表示コンポーネント」が相当すると考えられる。 これは、「感性工房」でいう「感性」とは少しニュアンスが異なるが、３つのプロジェクトが対象とする実問題により様々な特徴ある「感性」が出てきて、その結果 HM の研究がより深まることになると思われる。
---------------	---	---

2.8.4. 大画面統合表示システム構築のためのガイドライン

本章では、これまでの章の内容をとりまとめて、前章で示した「Operator／Co-operator モデル」、「HOPE_cps」、「TOPA-V ペイン方式」を具現化し、具体的にインプリメントするときに必要な基本設計指針を示す。

具体的には、

- （１）基本コンセプト・基本設計指針
- （２）タイリング制御の一般的指針
- （３）タイミング制御の一般的指針

から構成されている。

2.8.4.1. 基本コンセプト・基本設計指針

基本コンセプト・基本設計指針としてとりまとめたものを以下に示す。

- （１） 次世代プラントの大画面統合表示システムにおけるヒューマンインタフェース設計のコンセプト・モデルとしては、人である Operator と機械システムである Co-Operator が協調して問題解決をおこなうモデルである「Operator／Co-operator モデル」をもちいる。
- （２） 上記 Operator と Co-operator が協調して問題を解決していくために提供される、人を中心に考える設計法を指向してデザインされた、プロセスを共有する環境を HOPE_cps (Human-centred

Oriented Process Environment for collaborative problem solving) と呼ぶ。大画面による統合表示のプロセスを設計するときには、人間の認知活動を参考にした問題解決プロセスに対応した HOPE_cps を設計する。

(3) 問題解決プロセスは以下の用語で表現されるステップものを基本とする。

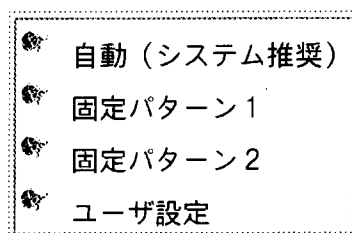
1. 正常運転中
2. 異常検知
3. 異常通知
4. 現在状態提示
5. 近未来状態提示
6. 異常原因候補提示
7. 異常原因提示
8. 対応操作候補提示
9. 操作手順表示
10. 確認項目提示
11. 異常リセット通知

(4) 上記各ステップに応じて大画面に統合表示される各種情報の表示パターン配置方式として、

- a. SAS (Static Allocation System) : 表示パターンをスタティックに固定的に配置して表示する方式。(この場合は、手動で大画面を切り替えないければ必要な情報を選択することはできない。)
- b. DAS (Dynamic Allocation System) : 表示パターンをダイナミックに配置して表示する方式。(この場合は、手動で大画面を切り替えることなく、自動的に、問題解決に必要とシステム側が選択提示する推奨画面を見ることができる。)

を選択できるように用意しておく。

具体的な選択法としては、例えば、画面設定ウインドウで下記のように選択設定ができるようにしておく。



(5) 上記 ダイナミックに配置表示する方式 DAS のひとつとして、ここでは TOPA-V ペイン方式を使用する。

TOPA-V ペイン方式とは、大画面ディスプレイのダイナミックなタイリング表示方式であり、T、O、P、A、V とは、各々：

- T : 問題解決プロセスのタイトルの表示領域
- O : オペレーション・判断参考情報の表示領域

P： 予測・システム状態の表示領域

A： アラームメッセージの表示領域

V： 仮想プラントの表示領域

を表し、これらの表示領域を統合的に組合わせて、問題解決のためのダイナミックなプロセス環境として提供する。

(6) ダイナミックに配置表示する方式として、TOPA-V ペイン方式を採用することにより、オペレータにとっては以下のメリットがある。

- オペレータにとっては、オペレーション・判断参考情報の表示領域が直接オペレーションに関係する領域であり、そのエリアの大きさの変化が問題解決のプロセスの進展に伴って自分に関係したタスクの重みとして現れていると気づき、オペレーターは問題解決の方向に注意を集中することができる。
- 画面構成の変化によって、問題解決に向けたシステムの進捗を感覚的に体感できる。
- 大量の情報を問題解決のプロセスに応じて、選択・整理して見ることができる。

(7) TOPA-V 方式におけるタイリング制御の一般指針とタイミング制御の一般指針を、各々以下の4.2節、4.3節に示す。

(8) 以下のタイリング制御、タイミング制御の基本設計方針として、複数異常発生時においても、短時間に大量の情報を提示することなく、問題解決プロセスに最低限必要な情報にフィルタリングして提示することを基本設計指針とする。

すなわち、オペレータに混乱を生じさせないように、あわてさせずに、安心感を与えることを基本設計指針とする。

2.8.4.2. タイリング制御の一般的指針

タイリング制御の一般的指針を2.8.2.1に示す。

2.8.4.3. タイミング制御の一般的指針

タイミング制御の一般的指針を2.8.2.2に示す。

3. 目標に照らした達成状況

3.1. 全体システム

本プロジェクトでは本来大画面表示システムは研究目標に照らせば研究範囲外であり、達成すべき目標は明記されていない。しかし、平成11年度研究の中で、大画面がないことにはヒューマンインタフェースとしての姿を見せることができず、融合化の効果も大きく失われることが予想されたため、研究開発チーム全体の努力、特にイメージ情報科学研究所の努力により、NTT-ATの指導のもと、IHIが大画面の実装を担当することとなり、当初の目標を上回る開発が実施され、融合化の成果を有効にデモンストレーションすることができた。

大画面統合表示システムを開発するにあたり、情報提示タイミング制御方法およびダイナミックウィンドウタイリング制御方法を考案し、これを実装することができた。ここで開発した方法はこれまでにないものであり、現在特許申請中である。実装にあたっては想定したシナリオのみならず、それ以外の異常に対しても動作できるよう汎用的に留意して開発した。想定外異常についてもオペレータへ情報提供することができる。

3.2. インタフェースエージェントの研究

3.2.1. 目標の達成度とその要因

「時系列データの学習機能と運転員とのコミュニケーション機能により、エージェントがプラントの挙動を監視しつつ運転員との対話を重ねることにより、個別のプラント状況に合わせてその能力を成長させ、適切な運転ができるようにする」という基本的目標は、事例に基づくプラント監視というこれまでにない新しい手法を提案実現し、その有効性を立証するという点では、本研究によって基本的に達成された。ユーザ評価に見られるように、本手法はこれまでにない斬新なプラント監視方法であり、特に石油業界では過去にエキスパートシステムの導入に失敗しているだけに、本手法の実用化への期待が高い。そもそもこのような斬新な方法を提案し得た理由には、提案者自身が過去に複数のプラント運転支援システムを構築し実用化してきた経験があり、次世代のプラント運転支援に望ましい要件を熟知していたことが挙げられる。

しかし、実用性の観点からはまだ多くの解決すべき課題が残されている。平成8年度、9年度研究のごみ焼却プラントを対象とした研究において、ごみ焼却プラントの実操業データを対象とした解析においてノイズ処理の問題が重要であることがわかったが、平成10年度以降の石油プラントを対象とした研究ではシミュレータを用いた研究となったため、インタフェースエージェントが「プラントの挙動を監視しつつ運転員との対話を重ねることにより」プラント監視が可能になることは示すことができたが、ノイズ処理に対しては対処する必要がなく、研究は行われなかった。実用化のためには、ノイズを含む実プラントデータを用いて有効性を立証する必要がある。

監視すべきデータの個数や選択においても課題がある。本プロジェクトでは石油プラントシミュレータを用いて、実施できるすべての異常に対処することができたが、それは今回監視に用いた7個のデータに強く依存している。次の新しい異常が起こったとき、現在の7個のデータで十分健全な監視ができるという保証はどこにもない。シミュレータが与えられたとき、検出すべき新しい異常を一個追加した場合に、追加すべき新しい監視データを現在の延長として試行錯誤的に見出すことは可能と思われるが、たとえばシミュレータがない新設のプラントにおいて、本手法で監視すべき典型的な異常をいくつか抽出したとき、どのようなデータをどこまで監視範囲とすればよいのか、それを見極めるような一般的な手法を見出すにはまだあきらかに経験が足りない。

実用化にあたっては監視のリアルタイム性能と監視データ数の関係についても、留意すべき点が残されている。本年度研究にて、簡易バージョン空間を導入することにより、データベースを用いた本格的なシステムにおいても十分な

性能で監視できることを示したが、ここでは2秒毎に7個のデータを監視した。

一般に、石油プラントではそれほどリアルタイム性は必要としないが、では30秒毎の監視でいくつのデータ数が監視できるのかは、(試行すればすぐ明らかにすることではあるが)現在不明であり、さらにプラントごとに要求されるリアルタイム性能は異なることから、本手法の実用化にはプラント種類毎に異なる検討が必要である。

3.2.2. 成果が与えるインパクト

最近になって、エージェント技術が益々、色々な分野で益々注目され研究されるようになってきた。本プロジェクトでは、音声対話を含むインタフェースエージェント技術、プラントを自律的に監視するエージェント技術、互いに異なる処理能力を持つエージェントが協調してタスクを処理するマルチエージェント技術などが、研究され開発された。プラント業界においても、先端的な部分でエージェント技術が語られるようになってきたが、本プロジェクトほど本格的なものはない。シミュレータをプラントの代替としていえるとは言え、実際に動作する複数の大規模なシステムを業界に提示したインパクトは大きいものがあり、その影響はこれから益々明らかになってくると確信している。本プロジェクトで示された各種エージェント技術は、プラント業界においてこれから常に言及され基準となる参照モデルとなるであろう。

3.3. 仮想プラント表示インタフェースの研究

3.3.1. 実用化までの道のり

- (1)開発された仮想プラント表示の実用性は高いと考えている。日石三菱石油殿に対するデモ実施時において、プラント状態がわかりやすく、仮想プラント表示単体でも、現状のDCSと組み合わせた形で使えるとの判断をいただいている。
- (2)仮想プラントの構築は、専門家以外ではなかなか容易ではない。現在、プラントを初めとして建造物を設計する際に3次元CAD等の使用は珍しくなくなってきたはいるものの、まだ、一般的ではない。3次元CADによる設計データがあれば、仮想プラントの構築は3次元データのフォーマット変換で容易となる。
- (3)汎用グラフィクスAPIによるVRツールのミドルウェア化を進めていたが、OpenGLベースの汎用グラフィクスAPI開発の流れが、開発メーカの都合から頓挫している。このため、今後は、Windows NT版ではなく、より、実用性・汎用性の高いLinux版Graphics APIで進める必要がある。
- (4)プロセスデータ可視化については、モジュール単位で汎用性を有しており、対象モデルが変更されてもコンテンツの変更は不要である。

3.3.2. プロジェクト終了後の成果の各社での活用計画

当面は、日石三菱石油殿を中核とした石油連盟加盟の各社に使っていただくことを目指して努力するとともに、提供した仮想プラントのユーザ側での簡単な編集機能を盛り込んだVRツールをLinuxベースで完全なミドルウェア化を進めていき、石油プラントに限らず、様々な産業用プラントを対象としていく。

3.4. 分散協調処理技術の研究

分散協調処理技術は、分散環境下に置かれた各サブシステムを統合し、全体として円滑なインタフェース機能を実現ことを目標に研究開発を行った。実際に本プロジェクトで石油プラント向けプロトタイプシステムを構築した際に、プロジェクト各社、各大学からは、使いやすさや柔軟性について良い評価を頂いている。しかし、プロトタイプではない実システムでの利用を想定した場合には、次に挙げるような課題が残されていると考えている。

3.4.1. パフォーマンスの改善

現在、通信プロトコルとして TCP/IP のみを利用しているため、多数のサブシステムに一斉通知する際に、個別通信(ユニキャスト)で対応せざるを得ず、特に接続されるサブシステム数が増えると、遅延の増加や通達順序の入れ替わり等の問題が発生する。

本課題については、TCP/IP の代わりに同報(マルチキャスト)通信プロトコルを適用すれば解決可能だが、現状、ネットワーク構成などを問わず標準的に利用可能なマルチキャスト通信プロトコルは、信頼性を保証しない UDP/IP しかない。TCP/IP と同様の信頼性を保証するマルチキャスト通信プロトコルを適用するためには、今後、相応の研究開発が必要と思われる。

また、システム開発時の柔軟性向上や、標準化への対応などを重視して、協調処理機構とサブシステムを接続するための分散処理機構に Java、および Jini を適用した。しかし、socket 通信などに比べて 1 対 1 通信時の通信遅延が大きく、即時応答性が求められるようなシステムへの適用に対して、表 3.4-1 のデータが示す通り実用上の課題がある。

表 3.4-1 性能計測結果

	DirectDemandによるタスク実行	Demandによるタスク分担実行	Broadcastによるタスク一斉通知
(サブシステム数)2	0.41	1.77	0.87
3		1.77	1.11
4		1.75	1.30
5		1.75	1.51
6		1.75	1.75
7		1.76	1.94
8		1.75	2.16
9		1.76	2.35
10		1.79	2.58

※単位は秒

計測環境：PC1(協調処理サーバ)、PC2(協調処理クライアント)

PC1: Pentium II 266MHz、Windows NT

PC2: Pentium III 600MHz、 Windows 2000

計測内容：

DirectDemand によるタスク実行

PC2 上で 2 サブシステム(タスク依頼元、タスク実行先)を実行し、タスク依頼元が DirectDemand を発行してから ReceiveResult を受取るまでの時間を計測(タスクエージェント処理を行わない、サブシステム間直接通信処理を計測)

Demand によるタスク分担実行

PC2 上で 2~10 サブシステム(タスク依頼元が 1、タスク実行先が 1~9)を実行し、タスク依頼元が Demand を発行してから ReceiveResult を受取るまでの時間を計測(タスクエージェント処理を行う、協調処理を含んだサブシステム間通信処理を計測)

Broadcast によるタスク一斉通知

PC2 上で 2~10 サブシステム(タスク依頼元が 1、タスク実行先が 1~9)を実行し、タスク依頼元が Broadcast を発行し、自身も Announce によるタスク通知を受取るまでの時間を計測(全サブシステムに対する一斉通知が完了するまでの時間を計測)

Java、および Jini は未だ発展途上の技術であり、今後の改良も見込まれるが、その改良を先取りするような研究開発には相応の研究開発が必要である。一方、根本的な解決策とは言えないが、上位プログラム層でキャッシュ機構等を実装し、個々のケースについて改善策をとるといった対策も考えられるが、実施には約半年から 1 年間の期間が必要と思われる。

3.4.2. 安定性の向上

分散協調処理技術では、当初から特定の応用に依存しない、汎用性をもった分散協調プラットフォームとして機能検討し、設計開発を行っている。実装した各機能については、動作検証レベルの試験を実施しているが、予めあらゆる機能を見越すことは不可能であり、実装を通した実証的な試験を積み上げることが望まれる。

3.4.3. ミドルウェアの整備

分散協調処理機構では、各サブシステム固有の通信アドレスやインタフェース仕様、タスク情報などを外部定義データとし、実行時に参照する形式とすることで、特定の応用システムに依存せず、汎用性をもった分散協調プラットフォームとして開発した。当初の開発では、これらの外部定義データの各項目を意味的に整理し、利用者が設定を容易にする仕組みが無く、設定に対する煩わ

しさが解決できていなかった。

今年度の開発では、Jini を利用したことによりネットワークコンフィグレーションの自動化などが実現でき、設定項目を絞り込むことができたため、ある程度は使い勝手を改善することができたが、依然として決められた書式に従って利用者が設定作業を行う必要は残されており、その労力を改善するために、作成、および動作確認を支援するツールを開発する必要があると考える。

3.5. プラントモデルとインタフェースシステムの研究

昨年に引き続き、各要素技術が必要とする各種データ、情報を提供した。必要に応じて各要素技術への技術サポートを継続した。また、ユーザーの立場からのこのシステムを評価するため、石油連盟あるいは日本能率協会へのデモンストレーション検討、アンケート調査を行った。

3.5.1. シナリオ作成・検討

新たに2個シナリオを作成し、合計9個のシナリオを提供した。昨年まで、2個のシナリオをシミュレータで検証していたが、残り7個の検証を実施した。トレイフラッシングシナリオのシミュレータでの検証は難しく、デモンストレーションの候補から除外したが、プロジェクトの進捗状況から、実装されたシナリオは5個で、8個のシナリオで十分であった。

また、シナリオによるシステム稼動のみでは、機能的な評価を行うことが難しく、シミュレータでシナリオにならない異常を発生させても、システムが相応の対応をすることが必要であることを要求した。まだ十分ではないが、シナリオ以外の異常にも対応可能となっている。

3.5.2. 意味表示インタフェース用データ収集・評価

簡易高速予測シミュレータの計算結果の評価を実施した。異常発生用のシミュレータや精度の高い静的シミュレータの計算結果などとの比較を行った。主精留塔の計算では、当初、トレイの負荷にかなりの誤差があったが、塔のトレイを幾つかのブロックに分けていたセクションを更に細分化したり、塔頂温度の設定の考え方に間違いがあった等、入力データ、計算結果を詳細にチェックし、改善策を提示した。簡易モデルでは概略の方向性を提示することが目的であり、計算時間の短縮をはかるため精度はそれほど要求されない。何とか使用できる精度はあると思われるが、主精留塔モデルの実使用までの検証は出来ない。その他更に、加熱炉出口の原油温度の変化、塔頂温度の変化などの影響など、経験式を提供している。

今までは、異常原因候補はシナリオに登録した内容に既にあるものとして使用していた。しかし、これでは実際の異常診断の流れは、ここで途切れてしまい、実用化の評価ができない。ユーザーとしては必要な機能であり、何らかの対応を依頼したところ、意味表示インターフェースで異常原因候補を選出するプログラムを開発した。トッパーを機能ブロックに分割し、ブロックごとに開発が進んでいる。時間的な問題から、主精留など未完成部分がありまだ十分とは言えないが、機能的には一連の機能が繋がり、システムとしての形が整っ

た。

3.5.3. プラントオントロジー用データの収集・評価

昨年に引き続き、オントロジーに提供している石油精製プラント語彙言語データベースの精度アップを計っている。一通り、修正し、最近は特に不具合は発生していない。ほぼ問題ないレベルに達していると判断される。

石油プラントでは、原油や製品の配管が縦横に入り組んでおり、配管ルートは運転員にとって重要な情報である。オントロジーの機能は、語彙の作成と機器の繋がり情報を持つことにあるが、配管の繋がり情報を管理するよう依頼した。この結果、異常場所の確認、異常感知計器の位置など VR と絡めてより明確に把握できるようになった。

3.5.4. VR 用データの収集・評価

昨年に引き続き、VR の内容の充実を計った。原油、製品の配管ルートの詳細情報を提供した。プラントオントロジーとの連携で異常場所などの確認がより詳細、明確になった。

温度、圧力、流量などの運転データの可視化などアドバイスしたが、石油連盟等へのアンケート結果でもやはり馴染みがなく、イメージが湧かないようである。必要性は少ないと思われる。

3.5.5. 運転用インタフェースエージェント用データの収集・評価

提供した異常シナリオをシミュレータで検証し、運転データを提供した。運転変数による異常検知での閾値の設定値見直しを行った。デモ用にということで多少通常と異なる設定のものがああり、実状を説明し再試行をおこなうなど、チューニングを行った。異常事例データは 7 個の変数の動きのパターンでチェックしている。計測データは 130 個程度であるが、ある程度の異常認識は可能であるが、十分かどうかの判断はもっといろいろな異常を発生させて確認する必要がある。今回は音声入出力の機能を持っているが、機能的には十分である。ただし、オントロジー語彙作成機能とは別な機能なので、オントロジーの語彙作成機能を活用したシステムにすれば、応用が広がるとと思われる。

3.5.6. 大画面の設計・評価

オペレータ向けの大画面の開発において、必要とする情報を見易い形で提供できるよう、オペレータの行動様式についての情報と必要な機能についての情報を提供した。このシステムでは、表示する情報と表示方法は妥当との評価を得たが、既存の DCS との使い分けを考慮する必要があるようである。

3.6. プラントオントロジーの研究

3.6.1. 成功の要因

本研究課題で設定した主な目標は、

- 実用規模のプラントオントロジーとプラントオントロジーに基づく対象プラントモデルを開発し、それを核にマルチエージェントシステムを稼働させること
 - オントロジーを利用したメッセージ生成システムを開発すること
 - オントロジー構築の支援環境を開発すること
- であった。

現時点において、以上の3つは目標を達成したと言える。実際、現在稼働中の統合システムにおいて、オントロジーサーバーが対象プラントモデルを各エージェントに提供しており、それによって全エージェントが対象プラントに関する論理的な構造情報を共有することが可能となっている。更に、標準的な用語もオントロジーサーバーで管理されており、エージェント間の会話もスムーズに進行することが確認されている。

また、オントロジーを用いたメッセージ生成システムも、使用する単語選択もコンテキストに依存して柔軟に適応する機能も盛り込まれ、生成するメッセージは想定した全9つのシナリオにおいて適切であることが確認された。オントロジー構築支援環境も研究室内でフルに活用され実績を積んでいるだけでなく、日石三菱殿に詳細に見ていただいた結果も良好であり、成功していると言えよう。

上記3つ以外の目標として開発システムのミドルウェア化がある。これもオントロジーと対象モデル開発においては(3)の課題の達成によってほぼ達成されている。しかし、メッセージ生成システムに関してはそのメンテナンスには開発者レベルの知識が要求され、エンドユーザが独力でメンテナンスできるまでには至っていない。

これらの要因であるが、達成したことの要因としては研究を始める以前から知識ベースとオントロジーに関しては豊富な実績を持っており、周到的な準備がなされていたこと、プロジェクト内での協調がスムーズに行われたことなどが考えられる。メッセージ生成に関するミドルウェア化が十分でなかった点は、メッセージ生成の中でもコンテキストに依存して単語を選択する部分にパワーを集中し、自然言語文生成そのものを本格的には扱わないことにするというプロジェクト全体での合意に基づく決定が大きな要因である。

3.6.2. 成果が与えるインパクト

実規模の石油精製プラントを対象にしたオントロジー開発と対象モデルの構築、並びにそれらを用いた知的処理システムの開発で成功したこと自体、学界に与えるインパクトは大きい。オントロジーに関する興味は近年十分に高まってはいたが、具体的な成功例が非常に少なかったからである。

また、産業界に対しても大規模システムの開発においてはこのようにオントロジーをまず開発して、それに基づいてシステム開発をおこなうという方法論が定着すれば得るところは大きい。

3.7. 意味表示インタフェースの研究

当初、本年度の研究開発課題として設定した項目は、(i) 表示意図に基づく意味表示の実装、(ii) 高速シミュレータを利用した対応操作候補の評価、(iii) 最終プロトタイプシステムの開発、および、(iv) ミドルウェアとしての整備であった。

本年度の研究では、項目 (i) については機能—目標レイヤーの表示にプラント状況や診断結果に応じた詳細表示と簡略表示の切り替えを実装した。項目 (ii) 対しては、高速シミュレータを完成させ、対応操作評価手法を開発した。また、ユーザサイドからの強い希望により、モデルに基づく異常原因推定部を検討開発した。さらに、運転操作や経験的知識を利用した対応操作候補の絞り込み手法を開発した。これらの、サブシステムの開発を踏まえて、項目 (iii) の最終プロトタイプシステムを開発するにあたり、意味表示インタフェースを2つの分散エージェントに分割した。すなわち、異常原因推定、対応操作候補導出、対応操作候補絞り込み、対応操作評価を行う推論部 (SIA1) と種々のグラフを自律的あるいは他エージェントからの要求により表示するグラフ表示部 (SIA2) である。このように、項目 (i) から (iii) については、当初の計画以上の研究開発が行えた。これは、項目 (iv) の整備の一部を未実施とすることに計画変更したこともあるが、共同研究機関の日石三菱殿の多大なる協力により、プラントに関する情報が十分に提供されたことが大きな要因である。また、最終プロトタイプシステムの開発においては、分散協調システムで実装したことが、各研究機関で開発したシステムの統合における調整がスムーズにいき計画通りに研究開発が進んだ要因と思われる。

しかしながら、項目 (iv) については、プラントモデル構築ツールの整備は実施できたが、主に時間的な原因により HPT の Java 版を整備するには至らなかった。これは、プロジェクトの開始時期においては HPT を Smalltalk-80 で実装し、Java への転換が遅れたことも影響している。HPT の Java 版整備については、プロジェクト終了後1年以内に実施したい。

プラント異常時のような時間制約時における人間の認知特性の特徴は、最初に注目したシグナルのみに執着する視野狭窄であると言われる。意味表示インタフェースは、複数の視点（機能、挙動、構造、運転）からの情報を統合的に提供するため、視野狭窄を回避する手がかりを与えることができる。この点で、意味表示インタフェースは、時間的な制約のある状況への対処を支援するマンマシンインタフェースの開発へ応用できる。

運転員がプラントの異常時に対応する場合に最終的に必要となる情報は、対応操作の実施手順に関する情報である。意味表示インタフェースプロトタイプ

システムが導出する対応操作情報には操作手順までは含まれていない。操作手順は運転マニュアルに詳細に記述されているのが通例である。従って、意味表示インタフェースと電子化された運転マニュアルとの結合を図ることが必要である。運転マニュアルは近年電子化される傾向にあるので、意味表示インタフェースとの結合は容易である。

石油精製プラントでは、環境変化や生産計画により運転条件がしばしば変更される。本年度のユーザ評価でも、プラント起動時やシャットダウン時などの非定常時の異常事象への対応の期待が高かった。意味表示インタフェースプロトタイプシステムでは、運転条件が変更されることまでは考慮していない。運転条件の変更に対しては、特に、異常原因推定のための知識を拡張することにより対応できると考えている。

機械の智能化が進むにつれて、人間と機械との間に介在するインタフェースシステムはますます重要になるであろう。人間と機械が互いにその特長を生かして効率的に協調するためには、それぞれの行動の意図を明確にして互いのインタラクションを図ることが必要である。表示意図を明確化した適応表示の方法は、柔軟な協調機構の開発に貢献できる。

なお、来年度以降、原子力関係研究機関と共同で意味表示インタフェースのサブシステムのうち、対応操作候補の導出、絞り込み、評価部を、原子力プラントに適用することを計画している。

3.8. 大画面統合表示システム構築のためのガイドライン作成

「大画面統合表示システム構築のためのガイドライン作成」の目標は、各種シナリオに対する大画面統合表示パターンを検討し、大画面統合表示に関する合意形成のアルゴリズムを導出すると共に、大画面統合表示のためのガイドラインを作成し、次世代プラント用ヒューマンインタフェースのシステム構築の検討に資することであった。

本年度は以下の項目を実施すると共に、具体的にフェーズ3システムにおけるデモシステムの実証にその成果を反映させることができ、当初の目標を達成することができた。

(1) プラント運転における各種異常シナリオの調査・分析

既存の各種異常シナリオを調査・分析し、大画面統合表示に対する要求条件を明らかにし、大画面統合表示パターンを分析・分類することができた。

(2) 大画面統合表示パターンのロジックの策定

既存のシナリオ、一般的シナリオおよびシナリオ以外の事象に対して動作する大画面統合表示パターンの制御アルゴリズム／ロジックを策定することができた。具体的には、情報内容の提示に関するウインドウタイリング制御の一般的指針と、オペレータに提示するタイミング制御の一般的指針を策定することができた。

(3) 大画面統合表示インタフェースのコンセプトモデルの策定

次世代用の大画面統合表示インタフェースのコンセプトモデルとして「Operator/Co-operator モデル」を策定し、ヒューマンメディア研究に資するコンセプトモデルとすることができた。

(4) 大画面統合表示システム構築のためのガイドライン作成

以上の(1)～(3)をとりまとめて、ガイドラインとすることができた。

なお、ここで得られた「大画面統合表示システム構築のためのガイドライン」と「コンセプト・モデル：Operator/Co-operator モデル」は、複雑なオペレーション・システムで大画面を必要とする場合のシステム設計において必須となる技術であり、本プロジェクトだけではなく、広く産業界において利用可能な技術となりうるものである。

4. 海外の技術動向調査

4.1. ISRA'2000

用件：

2000 2nd ISRA International Symposium on Robotics and Automation(ISRA'2000)にて次世代プラントヒューマンインタフェースの仮想プラント表示インタフェースに関する講演発表

出張者：

石川島播磨重工業(株) 遠藤政利

出張先：

メキシコ合衆国モンテレイ

期日：

2000 年 11 月 10 日～12 日

内容：

ISRA は、International Symposium on Robotics and Automation の略であり、今年で 2 度目である。ISRA は、主としてメキシコの下記学協会および IEEE RAS(Robotics and Automation Society)が主催したものである。

Consejo Del Sistema Nacional De Educacion Technica (COSNET)
Direccion General De Institutos Technologicos
Instituto Tecnologico De Nuevo Leon

Academia Nacional De Investigacion En Robotica (ANIRob)

参加者は、各国およびメキシコから 200 数十名が参加していた。

今回の訪米は、ヒューマンメディア技術の紹介が主目的であり、会議以外には調査活動を行っていない。

報告者は、Virtual Reality & Agent のセッションにおいて、5 件中の 1 件として「Development of Virtual Plant Agent –An Attempt to Represent Process Plant Information Using Virtual Reality」というタイトルで、仮想プラント表示インタフェース技術の紹介を行った。

その外の講演では基調講演として、Fusion of human and machine intelligence, T. J. Tan, Washington University, USA、Embedded Robotics and MEMS, Suhkan Lee, Samsung Advanced Institute of Technology, KOREA、Towards autonomous robots working in daily life environment, 筑波大学機能工学系 油田信一教授のロボティクスについての講演があった。

5. 研究発表・講演、文献、特許等の状況

5.1. 研究発表・講演

- [1] Koide, S. and M. Kawamura, An Implementation of Case-based Memory of an Interface Agent by Lisp, Java, and C++, Proceedings of the Japan Lisp User Group, (2000), Franz.
- [2] 小出誠二, プラント運転用インタフェースエージェントの記号接地問題, 人工知能学会全国大会(第14回)論文集, pp.512-513, (2000).
- [3] 小出誠二, インタフェースエージェント, 日本原子力学会ヒューマンマシンシステム研究部会第11回夏季セミナー, pp.74-84, (2000).
- [4] Masatoshi Endo and S. Misono, Development of Virtual Plant Agent –An Attempt to Represent Process Plant Information Using Virtual Reality, Proceedings of the 2nd International Symposium on Robotics and Automation, pp.354-359, (2000), ISRA.
- [5] 御園昇平, 遠藤隆也, 溝口理一郎, 五福明夫, 渡部修介, 分散協調型次世代プラント用大画面ヒューマンインターフェース, 電気関係学会関西支部連合大会シンポジウム
- [6] 坂下善彦, 渡部修介, 徳本修一, 三石彰純, プラント運転支援向け分散協調機構, 情報処理学会マルチメディア通信と分散処理研究会, 2001.3
- [7] Foundation of Knowledge Systematization: Role of Ontological Engineering : Riichiro Mizoguchi and Yoshinobu Kitamura, Industrial Knowledge Management - A Micro Level Approach, Rajkumar Roy Ed., Chapter 1, pp.17-36, Springer-Verlag, London, 2000
- [8] Construction and Deployment of a Plant Ontology: Riichiro Mizoguchi, Kouji Kozaki, Toshinobu Sano and Yoshinobu Kitamura, Knowledge Engineering and Knowledge Management - Methods, Models and Tools -, The 12th International Conference, EKAW2000, (2000), 113-128
- [9] Development of an Environment for Building Ontologies which is based on a Fundamental Consideration of "Relationship" and "Role": Kouji Kozaki, Yoshinobu Kitamura, Mitsuru Ikeda, and Riichiro Mizoguchi, The Sixth Pacific Knowledge Acquisition Workshop, PKAW2000, pp.205-221 (2000)
- [10] 「ロール」および「関係」に関する基礎的考察に基づくオントロジー記述環境の開発: 古崎, 来村, 溝口, 人工知能学会誌 (投稿中)
- [11] オントロジー構築・利用環境「法造」の開発と利用ー実規模プラントのオ

ントロジーを例として：古崎，本松，佐野，来村，溝口，人工知能学会誌（投稿中）

- [12]五福明夫, 田中 豊, 機能モデルを用いて導出された対応操作候補の運転, 挙動知識による絞込み, 日本機械学会 2000 年度年次大会講演論文集, Vol. II, pp.491-492, (2000).
- [13]Akio Gofuku, Intention as a Basis of Communicating Information between Human and Machine, presented in Seminar on Advanced Human-Machine Cooperation (HMC) in Complex Dynamic Systems, (2000).
- [14]五福明夫, 田中 豊, 表示意図に基づく異常時プラント機能情報の適応表示, ヒューマンインタフェースシンポジウム 2000 論文集, pp.337-340, (2000).
- [15]伊井謙司, 五福明夫, 田中 豊, 石油精製プラントの運転支援のための意味表示インタフェースの開発, 日本機械学会第7回動力・エネルギー技術シンポジウム講演論文集, pp.398-401, (2000).
- [16]Akio Gofuku, Usage of Functional and Structural Information of Plants to Generate Dynamic Simulation Programs and Operator Support Information, Proc. 2000 The Seminar of Core University Program's Nuclear Instrumentation and Control Subtask, pp.74-91, (2000).
- [17]Akio Gofuku, Yutaka Tanaka, Selection of Plausible Counter Actions Derived from a Fundamental Model By Operational and Behavioral Knowledge, Int. Conference on Cognitive Systems Engineering in Process Control, pp.156-161, (2000).
- [18]五福明夫, 小出誠二, 次世代プラント用異常時運転支援情報の生成と表示, 平成12年電気関係学会関西支部連合大会講演論文集, p.S31, (2000).
- [19]御園昇平, 遠藤隆也, 溝口理一郎, 五福明夫, 渡部修介, 分散協調型次世代プラント用大画面ヒューマンインタフェース, 平成12年電気関係学会関西支部連合大会講演論文集, p.S32, (2000).
- [20]五福明夫, 次世代プラントにおける運転員支援ヒューマンインタフェース, 2000 計装制御技術会議, pp.1.2.1-1.2.19, (2000).
- [21]五福明夫, 犬塚 圭, 田中 豊, 機器接続情報を利用した探索範囲拡大型異常原因推定手法, 日本機械学会第10回設計工学・システム部門講演会講演論文集, pp.209-210, (2001).
- [22]Akio Gofuku, Kei Inuzuka, Yutaka Tanaka, An Anomaly Identification Technique Based on Functional Models of Components, to appear in Int. NAISO Congress on Information Science Innovations, (2001).

5.2. 文献

- [1] Riichiro Mizoguchi, Akio Gofuku et al., Human Media Interface System for the Next Generation Plant Operation, Conference Proceedings of IEEE SMC'99, pp.630-635(Vol2), 1999, IEEE
- [2] Masatoshi Endo et al., Development of Human-Machine Interface Composed of Virtual Reality, Conference Proceedings of IEEE SMC'99, pp.636-641(Vol2), 1999, IEEE
- [3] Gerhard Weiss, Multi-agent Systems, MIT Press, 1999
- [4] Jörg P. Müller, The Design of Intelligent Agent, Springer, 1996
- [5] オントロジー構築過程におけるタスク・ドメインロールの抽出・整理に関する考察 ～石油精製プラントを例にして～: 石川誠一、古崎晃司、来村徳信、溝口理一郎, 人工知能学会研究会資料 SIG-KBS-A003-4, , pp.-(2000 年 11 月)
- [6] 機能に関する知識記述と推論のためのオントロジー: 来村徳信, 溝口理一郎, 2000 年度人工知能学会全国大会(第 14 回), pp.518-521, 2000
- [7] オントロジー構築利用環境「法造」における「関係」および「ロール」の取り扱い: 古崎晃司, 来村徳信, 池田満, 溝口理一郎, 2000 年度人工知能学会全国大会(第 14 回), pp.522-525, 2000

5.3. 特許等

- [1] P2000-358346, プラント運転監視支援装置、石川島播磨重工業、御園昇平、小出誠二
- [2] P2001-081381, プラント運転監視装置および方法、石川島播磨重工業、速水努、御園昇平

5.4. その他の公表（プレス発表等）

- [1] 雑誌「映像情報」、2000 年 6 月号、小特集 映像技術による設計、製造レビュー ション「3 次元 VR 映像を利用したプラント運転用ヒューマンインタフェイス」として発表、pp.67-72

本報告書の内容を公表する際はあらかじめ
新エネルギー・産業技術総合開発機構
応用技術開発室の許可を受けてください。

電話：０３－３９８７－９３８８

ヒューマンメディアの研究開発
石油プラント高度情報化システム技術研究開発

平成１３年３月発行

発 行 財団法人イメージ情報科学研究所

〒559-0034

大阪市住之江区南港北１－１４－１６

ＷＴＣビル２１階

ＴＥＬ ０６－６６１５－６３５５

ＦＡＸ ０６－６６１５－６３５３

禁無断転載