

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2009-507309
(P2009-507309A)

(43) 公表日 平成21年2月19日(2009.2.19)

(51) Int.Cl.

G06F 19/00 (2006.01)

F 1

G06F 19/00

110

テーマコード(参考)

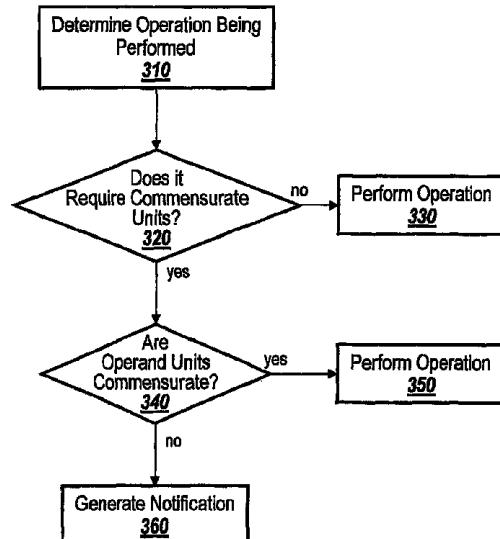
		審査請求 未請求 予備審査請求 有 (全 23 頁)	
(21) 出願番号	特願2008-530149 (P2008-530149)	(71) 出願人	505384841 ザ マスワークス, インク THE MATHWORKS, INC. アメリカ合衆国 マサチューセッツ州 01760 ナティック アップル ヒル ドライブ 3 3 Apple Hill Drive, Natick, MA 01760 (U.S.)
(86) (22) 出願日	平成18年9月6日 (2006.9.6)	(74) 代理人	100136630 弁理士 水野 佑啓
(85) 翻訳文提出日	平成20年4月10日 (2008.4.10)	(72) 発明者	キエルツエンカ, ヤセック アメリカ合衆国 マサチューセッツ州 01701 フラミングハム アン杰リカ ドライブ 61
(86) 國際出願番号	PCT/US2006/034625		
(87) 國際公開番号	W02007/030482		
(87) 國際公開日	平成19年3月15日 (2007.3.15)		
(31) 優先権主張番号	11/221,191		
(32) 優先日	平成17年9月6日 (2005.9.6)		
(33) 優先権主張国	米国(US)		

最終頁に続く

(54) 【発明の名称】 モデリング言語における物理計量単位

(57) 【要約】

本発明は、物理システムをモデル化するために固有の計量単位を提供する。すると、これら計量単位を用いて物理システムのモデル化が可能となる。従って、実際の単位は現実世界の測定値を反映しており、モデル化されたシステムで使用且つ伝播させることができ、計量単位を持つ結果を与えることができる。幾つかの実施形態では、計量単位を調べて、所与の演算に関して互いに適合しているかを判断してもよい。仮に複数の単位が互いに適合していなければ、ユーザには単位を互いに適合させる必要があることを通知してもよい。



【特許請求の範囲】**【請求項 1】**

物理モデリング環境における一方法であって、
前記物理モデリング環境で使用される固有の計量単位を与える段階と、
計量単位を持つ物理システムをモデル化する段階とを含む、方法。

【請求項 2】

前記物理システムの前記モデルの前記計量単位を調べて、該単位が所与の演算に関して互いに適合しているかを判断する段階を更に含む、請求項1に記載の方法。

【請求項 3】

前記計量単位が所与の演算に関して互いに適合していない場合は、通知を生成する段階を更に含む、請求項2に記載の方法。 10

【請求項 4】

前記不適合性への対処法に関してユーザ入力を要求する、請求項3に記載の方法。

【請求項 5】

前記計量単位は前記モデル化された物理システム内を伝播する、請求項1に記載の方法。
。

【請求項 6】

前記固有の計量単位はデータベースを介して与えられる、請求項1に記載の方法。

【請求項 7】

前記データベースは、固有の計量単位を追加するためユーザによるカスタマイズが可能である、請求項6に記載の方法。 20

【請求項 8】

前記データベースは、固有の計量単位を削除するためユーザによるカスタマイズが可能である、請求項6に記載の方法。

【請求項 9】

前記物理システムが、前記物理モデリング環境において少なくとも部分的にグラフィックモデル化される、請求項1に記載の方法。

【請求項 10】

前記計量単位は前記物理システムの前記グラフィックモデルに表示される、請求項9に記載の方法。 30

【請求項 11】

前記物理システムは、前記物理モデリング環境において少なくとも部分的にテキスト形式でモデル化される、請求項1に記載の方法。

【請求項 12】

モデル化している前記物理システムの前記計量単位を、同一の計量単位に変換する段階を更に含む、請求項1に記載の方法。

【請求項 13】

変換のための基本単位は、浮動小数点計算による丸めの誤差を最小化するよう選択される、請求項12に記載の方法。

【請求項 14】

前記物理モデリング環境はMATLAB物理モデリング環境である、請求項1に記載の方法。 40

【請求項 15】

前記物理モデリング環境はSIMULINK物理モデリング環境である、請求項14に記載の方法。
。

【請求項 16】

前記モデル化されたシステムからコードを生成する段階を更に含む、請求項1の方法。

【請求項 17】

一方法を実行するため計算装置により実行可能な命令を保持した、該計算装置と共に使用される媒体であって、該方法が、

固有の計量単位を与える段階と、 50

計量単位を持つ物理システムをモデリングする段階とを含む、方法。

【請求項 1 8】

前記物理システムの前記モデルの前記計量単位を調べて、該単位が所与の演算に関して互いに適合しているかを判断する段階を更に含む、請求項17に記載の方法。

【請求項 1 9】

前記計量単位が所与の演算に関して互いに適合していない場合は、通知を生成する段階を更に含む、請求項18に記載の方法。

【請求項 2 0】

同一基準でない単位への対処法に関してユーザ入力を要求する、請求項19に記載の方法。

10

【請求項 2 1】

前記計量単位は前記モデル化された物理システム内を伝播する、請求項17に記載の方法。

【請求項 2 2】

物理モデリング・アプリケーションを生成し且つ表示するシステムであって、データを前記物理モデリング・アプリケーションに入力するためのユーザ操作可能な入力手段と、

物理モデルを表示するための表示装置と、

コンピュータプログラム命令とデータを記憶するためのメモリ及び該記憶したコンピュータプログラム命令を実行するためのプロセッサを含むコンピュータ装置とを含み、該コンピュータプログラム命令は、固有の計量単位を与え、固有の計量単位を持つ物理システムをモデリングするための命令を含む、システム。

20

【請求項 2 3】

グラフィカルモデリング・アプリケーションを生成し且つ表示するシステムであって、クライアント装置に物理システムをモデリングするための固有の計量単位を与える配布サーバと、

該配布サーバと通信するクライアント装置とを含む、システム。

【請求項 2 4】

前記クライアント装置は、計量単位を持つ物理システムをモデリングするよう動作可能である、請求項23に記載のシステム。

30

【請求項 2 5】

前記クライアント装置が、コンピュータプログラム命令とデータを記憶するためのメモリ及び該記憶したコンピュータプログラム命令を実行するためのプロセッサを含む計算装置であり、該コンピュータプログラム命令は、固有の計量単位を与え、固有の計量単位を持つ物理システムをモデリングするための命令を含む、請求項23に記載のシステム。

【請求項 2 6】

物理モデリング環境を実行するサーバと、該サーバと通信するクライアント装置とを備えたネットワークにおいて、一方法は、

前記サーバで固有の計量単位を与える段階と、

計量単位を持つ入力を前記サーバで前記クライアントから受け取る段階と、該計量単位を持つ物理システムをモデリングする段階とを含む、方法。

40

【請求項 2 7】

物理モデリング環境における方法であって、

前記物理モデリング環境において使用される固有の計量単位を与える段階と、

計量単位を持つ物理システムのモデルを構築する段階と、

前記物理システムの前記モデルを介して計量単位を伝播させる段階とを含む、方法。

【請求項 2 8】

物理モデリング環境における方法であって、

前記物理モデリング環境において使用される固有の計量単位を与える段階と、

計量単位を持つ物理システムのモデルを構築する段階と、

50

前記物理システムの前記モデルの前記計量単位を調べて、該単位が所与の演算に関して互いに適合しているかを判断する段階とを含む、方法。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願

本願は、その内容を引用して本明細書に援用する2005年9月6日付けの米国特許出願第11/221,191号の利益を主張する。

【0002】

本発明は、物理システムをモデル化するためのモデリング環境に関する。より詳細には、本発明は、物理システムをモデル化するために固有の計量単位(原語: *intrinsic units of measure*)を提供することに関する。

【背景技術】

【0003】

多くの組織が、その生産工程においてモデルベース開発の理論的枠組を採用している。「モデルベース開発」とは、一組のブロックダイヤグラム及び関連付けられたオブジェクトからなる共通の「モデル」を用いてシステムを規定し、分析し、且つ実装する行為を指す。システム実装は、典型的に、モデルの複数の部分とりわけシステムの制御アルゴリズムに対応した部分のコードを自動的に生成することからなる。

【0004】

グラフィカルモデリング環境はソフトウェア・アプリケーションの一例であり、ブロックダイヤグラムなどのグラフィックモデルを用いたユーザによる動的システム(すなわち、出力が時間と共に変化するシステム)のモデル化を可能とする。グラフィカルモデリング環境によってはモデルのシミュレーション及び分析が可能なものもある。動的モデルのグラフィカルモデリング環境でのシミュレーションは、典型的には二段階からなる処理である。先ず、ユーザは、シミュレーションすべきシステムのブロックダイヤグラムのようなグラフィカルモデルを作成する。グラフィカルモデルは、グラフィカルモデル・エディタのようなグラフィカル・ユーザインターフェースを用いて作成できる。グラフィカルモデルは、システム入力と、状態と、パラメータと、出力との間の時間に基づく関係を表現する。グラフィカルモデルの作成後、指定された時間にわたる動的システムの挙動を、グラフィカルモデルに入力された情報を用いてシミュレーションする。この段階では、グラフィカルモデルを用いて動的システムの出力の時間的進展を計算し且つ追跡し(グラフィカルモデルの実行)、更に、モデル全体又は一部の挙動を模倣する導入可能なソフトウェアシステム又はハードウェアシステムの記述を自動生成する(コード生成)。

【0005】

ブロックダイヤグラムは、動的システムをモデル化するためのグラフィカルモデリング環境内で作成される「実行可能な意味」を持ったグラフィカル実体であり、一般的に1つ又はそれ以上のグラフィカルオブジェクトを含む。例えば、動的システムのブロックダイヤグラム・モデルは、一般に線として図示されるグラフィックオブジェクトの集まりによって相互連結されたノードのようなグラフィックオブジェクトの別の集まりとして模式的に表される。ここで、これらの線は後者の集まりのグラフィカルオブジェクト間の論理結合を表す。ブロックダイヤグラミングに関するほとんどの理論的枠組では、ノードは「ブロック」と呼ばれ、幾何学的オブジェクト(例えば、円、長方形など)の形式を使って描かれる。線分はしばしば「信号」と呼ばれる。信号は、各線結合により表現される時間変化量に対応し、各瞬間ににおいて値を持つと仮定される。各ノードは基本動的システムを表すことができ、信号と状態変数との間の関係は、ノードにより表される式の組により定義される。信号と状態変数との間の関係の定義にはパラメータの概念が本質的に含まれており、パラメータは式の係数である。これら式は、入力信号と、出力信号と、状態と、時間と

10

20

30

40

50

の関係を定義するので、それぞれの線は、関連付けられた基本動的システムの入力及び／又は出力を表す。1つのノードから出て別のノードで終結する線は、最初のノードの出力が2番目のノードの入力となることを示す。任意のノードにおける個別の各入力又は出力はポートと呼ばれる。信号のソースノードは、そのシステムの式が解かれた所与の時刻に信号に書き込みを行う。この信号の宛先ノードは、このシステムの式が解かれている時にその信号から読み出しを行う。これら結合は因果的ではない場合もある。すなわち、第1ノードを第2ノードに結合する線は、出力が第1ノードから第2ノードに入力として与えられることを必ずしも意味しない。非因果的な結合の一例は、計算の方向性が存在しない電気回路に見つかる。物理システムでは、非因果的結合は、一組の変数が等しく(例えば電気回路の電圧)、他の組の変数は合計してゼロになる(例えば電気回路の電流)ことをしばしば示す。当業者であれば、「ノード」という用語は基本動的システムのみを排他的に示すのではなく、ブロックダイヤグラムの可読性及びモジュール性を向上させる他のモデリング要素を含みうることは理解するはずである。

【0006】

物理システムは、長さ、質量、時間、弧長、温度、及び電荷などの基本次元を備えている。物理システムにおいて次元に意味を与えるのは、これら次元の値に関連付けられている単位である。そして、例えば、長さ(1メートル)、時間(1秒)、速度(1メートル毎秒)などのように、次元の値に関連付けられた量と、それらの相互作用を定義することにより意味が与えられる。計量単位が完全に妥当性検査され且つシステム全体を伝播する物理モデリングシステムにおいて、固有の計量単位は現時点では実装されていない。すなわち、現時点では、物理システムは使用される値に固有に関連付けられた計量単位なしでモデル化されている。複数の入力値が整合性のない計量単位を持つことがあるため、これは問題を引き起こしかねない。例えば、ある値が例えば5メートルのようなメートル単位であり、他の値が例えば50cmのようなセンチメートル単位となっていることがありうる。関連付けられた計量単位(mとcm)がなければ、これらは単なる値(5と50)にすぎない。従って、関連付けられた計量単位がない状態でこうした値を加算すると答えは55になってしまふ。5メートルと50センチメートルを加算すれば5.5メートルか550センチメートルとなるはずなので、これは正答ではない。このように、システムを正確にモデル化する上で計量単位は非常に重要となりうる。

【0007】

発明の概要

本発明は、物理システムをモデル化するために固有の計量単位を提供する。すると、これら計量単位を用いて物理システムのモデル化が可能となる。従って、実際の単位は現実世界の測定値を反映しており、モデル化されたシステムで使用且つ伝播させることができ、適切な計量単位を持つ結果を与えることができる。

【0008】

第1の様態によれば、物理モデリング環境において一方法が提供される。前記方法は、固有の計量単位を与える段階と、計量単位を持つ物理システムをモデル化する段階とを含む。

【0009】

別の様態によれば、命令を保持し且つ計算装置と共に使用される媒体が提供され、該命令は一方法を実行するため該計算装置により実行可能である。前記方法は、固有の計量単位を与える段階と、計量単位を持つ物理システムをモデル化する段階とを含む。

【0010】

第1の様態によれば、物理モデリング・アプリケーションを生成し且つ表示するシステムが提供される。前記システムは、データを前記物理モデリング・アプリケーションに入力するためのユーザ操作可能な入力手段と；物理モデルを表示するための表示装置と；コンピュータプログラム命令とデータを記憶するためのメモリ及び該記憶したコンピュータプログラム命令を実行するためのプロセッサを含むコンピュータ装置とを含み、該コンピ

ユータプログラム命令は、固有の計量単位を与え、固有の計量単位を持つ物理システムをモデリングするための命令を含む。

【0011】

第1の様態によれば、物理モデリング・アプリケーションを生成し且つ表示するシステムが提供される。前記システムは、クライアント装置に物理システムをモデリングするための固有の計量単位を与える配布サーバと、該配布サーバと通信するクライアント装置とを含む。

【0012】

別の様態によれば、物理モデリング環境を実行するサーバと、該サーバと通信するクライアント装置とを備えたネットワークにおいて、一方が提供される。前記方法は、前記サーバで固有の計量単位を与える段階と、計量単位を持つ入力を前記サーバで前記クライアントから受け取る段階と、該計量単位を持つ物理システムをモデリングする段階とを含む。

【発明を実施するための最良の形態】

【0013】

本発明の例示的な実施形態は、物理モデリング環境において固有の計量単位を組み込むことに関する。例示的な実施形態に関連して本発明を説明する。当業者であれば、本発明は、多数の異なる応用例及び実施形態で実現可能であり、本明細書に記載した特定の実施形態に特に限定されることは理解するはずである。

【0014】

物理システムをモデル化するための固有の計量単位が与えられる。すると、これら計量単位を用いて物理システムのモデル化が可能となる。従って、実際の単位は現実世界の測定値を反映しており、モデル化されたシステムで使用且つ伝播させることができ、計量単位を持つ結果を与えることができる。

【0015】

図1は、本発明の例示的な実施形態を実施するのに適した環境を示す。計算装置2は、本発明の一実施形態によるソフトウェアが記憶されたメモリ4と、このメモリ内に記憶されたソフトウェアを実行するプロセッサ(CPU)7と、システムハードウェアを制御する他のプログラムとを含む。典型的には、計算装置2と人間のユーザとの対話は、グラフィカル・ユーザインターフェース(GUI)などの入出力(I/O)装置8を介して行われる。I/O装置8は表示装置8a(例えばモニタ)及び入力装置(例えば、マウス8b、キーボード8c、及び他の適切な従来のI/O周辺機器)を含むことができる。

【0016】

例えば、メモリ4は、システムモデルの電子バージョンを作成且つシミュレーション可能なモデリング・アプリケーション6を保持している。こうしたシステムモデルの電子バージョンには、ブロックダイヤグラム、状態ダイアグラム、信号ダイアグラム、フローチャート・ダイアグラム、シーケンス図、UMLダイアグラム、データフローダイアグラム、回路図、ラダーロジック・ダイアグラム、運動学的要素ダイアグラム、又はその他のモデルがあり、何れかを表示装置8aを介してユーザ10に表示できる。例示的なこの実施形態では、モデリング・アプリケーション6は、Simulink(登録商標)などのMATLABモデリング環境又は他の適切なモデリング環境を含む。本明細書では「モデリング環境」と及び「物理モデリング環境」という用語は、物理システムのモデルなどのモデルが作成され、実行可能命令に翻訳されるアプリケーションを指す。適切なモデリング・アプリケーションの例としては、The MathWorks社のSimulink、SimMechanics、及びSimDrivelineを含むMATLAB、National Instruments社から入手可能なLabVIEW、DasyLab、及びDiaDem、Agilent社から入手可能なVEE、Measurement

Computing社から入手可能なSoftWIRE、Visual Solutions社から入手可能なVisSim、Elanix社から入手可能なSystemVIEW、Coreco社から入手可能なWiT、PPT Vision社から入手可能なVision Program Manager、Khoros Research社から入手可能なKhoros、並びにその他多数が含まれるが、それらに限定されない。メモリ4は、例えば、CD-ROM、フロッピー(登

10

20

30

40

50

録商標)ディスク、テープ装置などの任意適切なインストール媒体や、DRAM、SRAM、EDO RAM、ラムバスRAMなどのコンピュータ・システムメモリ又はランダムアクセスメモリや、ハードドライブ又は光学記憶装置を含む磁気媒体などの不揮発性記憶装置を含むことができる。このメモリは、他の種類のメモリ又はそれらの組合せを含むことができる。

【0017】

代替的な実施形態では、更に、計算装置2がインターネットなどのネットワークに接続されている。当業者であれば、ダイヤグラミング・アプリケーション6が使用するダイヤグラムは計算装置2上にローカルに記憶させてもよいし、ネットワークを介して計算装置と接続したリモート位置9に記憶させてもよいことは理解するはずである。同様に、ダイヤグラミング・アプリケーション6はネットワークサーバ又はリモートピアに記憶してもよい。

10

【0018】

本発明の例示的な実施形態のモデリング・アプリケーション6は、多数の汎用構成要素を含む。本明細書の説明はマサチューセッツ州ナティック所在のThe MathWorks社のMATLABに絞ったものだが、当業者であれば、本発明は他のソフトウェア・アプリケーションにも適用できることは理解するはずである。例示的なモデリング・アプリケーション6の汎用構成要素は、動的システムのモデルを指定するモデルエディタ6aを含む。モデルエディタ6aにより、ユーザは、物理システムを表すモデルの構築、編集、表示、注釈付け、保存、及び印刷などの作業が可能となる。更に、例示的なダイヤグラミング・アプリケーション6は、モデルシステムに組み合わせ可能な要素を含む要素ライブラリ6bも含んでいる。グラフィカルモデリングを可能とするグラフィックアスペクト6cを備えることもできる。実行エンジン6dもこのアプリケーションに実装されており、モデルを処理してシミュレーションの結果を生成したり、このモデルを実行可能コードに変換したりするために用いられる。実行エンジン6dはモデルを実行可能実体に翻訳する。これら実行可能実体はコンパイルされ、コンピュータなどの計算装置で実行されて当該モデルが指定する機能を実装する。典型的には、コード生成により、生成コードの呼び出しひずのモデル階層が保存される。例えば、物理モデリング環境内のモデルの各サブシステムは、ユーザ指定関数及び生成コードにマッピングできる。マサチューセッツ州ナティック所在のThe Mathworks社から入手可能なReal-Time

20

Workshopは、コード生成に適した実行エンジン6dの一例である。

30

【0019】

例示的なこの実施形態では、モデリングプログラム6は、これもMathWorks社から入手可能なMATLABなどの技術計算プログラムの随伴プログラムとして実装されている。

【0020】

モデルエディタ6aは、物理システムを表すモデルのユーザによる作成及び修正を可能とする構成要素である。更に、モデルエディタ6aは、モデル要素6bに関連したデータのユーザによる作成及び記憶を可能とする。一組の命令とのテキスト形式インターフェースにより、モデルエディタとの対話が可能となる。ユーザは、このテキスト形式インターフェースを用いてモデルに自動編集処理を行う特殊スクリプトを書くことができる。一般に、ユーザは、モデルのキャンバスとして機能する一組のウィンドウと対話する。モデルには一般に複数のウィンドウがあるが、それはモデルがサブシステムを使用することで多数の階層レベルに分化できるからである。

40

【0021】

一旦、モデルがエディタ6aを使って構築されると、実行エンジン6dが、このモデルにより定義された式を解いて時間の関数としてシステム出力を追跡することで、このモデルをシミュレーションする。モデルの解はモデル実行と呼ぶことができ、ユーザが指定する時間枠で一組のユーザ指定入力に関して実行される。

【0022】

本発明の例示的な一実施形態によれば、固有の計量単位を持つ物理システムをモデル化できる。図2は、本発明による方法の代表的なフローチャート200を示す。この方法は、

50

固有の計量単位を与える段階210と、計量単位を持つ物理システムをモデル化する段階220とを含む。幾つかの実施形態では、この方法は、物理システムのモデルからコードを生成する段階230を更に含む。

【0023】

固有の計量単位を与える(段階210)ための1つの方法は、単位データベース又はレジストリの提供を含む。このデータベース又はレジストリは、モデリングプログラム6の一部としてメモリ4の内部又は外部記憶装置9内に設ければよい。この単位データベース又はレジストリは登録済み単位のリストである。このデータベース又はレジストリは基本単位の他にユーザ定義単位を含むことができる。このデータベースを用いれば、ユーザは次元とその基本単位とを登録できる。次元を登録するには次のような関数を使用できる。

10

adddimension ('length', 'm')

【0024】

ここで登録される次元は、基本単位であるメートル(記号「m」で表した)を持つ長さである。任意の数の次元をこの方法で登録できることに注目されたい。その他の登録可能な次元は、本開示を参照すれば当業者には明らかなはずである。

【0025】

同様に、ユーザはデータベースに単位を登録できる。単位を登録するには次のような関数を使用できる。

20

addunit ('cm', 0.01, 'm')

【0026】

ここで登録される単位はセンチメートル(「cm」)であり、0.01メートル(「m」)として定義されている。任意の数の単位をこの方法で登録できることに注目されたい。その他の登録可能な単位は本開示を参照すれば当業者には明らかなはずである。

30

【0027】

従って、これらの技法を用いれば、モデル化された物理システムの次元と単位が与えられる。この一例を次に示す。

Function pm_units

%PM·UNITS・物理モデリング単位のコンテキストで単位を登録する

%

% 基となる次元 SI

%

pm_adddimension ('length', 'm');

pm_adddimension ('mass', 'kg');

pm_adddimension ('time', 's');

pm_adddimension ('charge', 'C');

pm_adddimension ('temperature', 'K');

10

%

% SI 系長さの誘導単位

%

pm_addunit ('cm', 0.01, 'm');

pm_addunit ('mm', 1e-3, 'm');

pm_addunit ('km', 1000, 'm');

%

% エネルギー単位

20

%

pm_addunit ('J', 1, 'kg*m^2/s^2');

pm_addunit ('Btu', btu_2_J, 'J');

%

% 仕事率単位

%

pm_addunit ('W', 1, 'J/s');

pm_addunit ('HP', 735.4996249, 'W');

%

30

% 電気単位

%

pm_addunit ('V', 1, 'J/c');

pm_addunit ('A', 1, 'C/s');

pm_addunit ('F', 1, 'A*s/V');

pm_addunit ('H', 1, 'V*s/A');

pm_addunit ('Ohm', 1, 'V/A');

% [EOF] pm_units.m

40

【 0 0 2 8 】

上記の例では、長さ、質量、時間、電荷、及び温度という次元が、それぞれの基本単位であるメートル(m)、キログラム(kg)、秒(s)、クーロン(c)、及びケルビン(K)と共に登録される。これら次元と基本単位とを用いて、長さの誘導単位、エネルギーの単位、仕事率の単位、及び電気の単位を登録できる。上記の例から分かるように、ジュール(kg*m2/s2)のような複合単位もサポートされる。

【 0 0 2 9 】

上述の例は「`adddimension`」及び「`addunit`」関数しか用いていないが、これらが唯一可能な関数と解釈すべきではない。幾つかの実施形態では、例えば、「`cunit`」及び「`unit`」関数などを使用してもよい。「`cunit`」関数は単位のない値を単位に結び付けるのに使

50

用できる。次に一例を挙げる。

a = cunit(4, 'm')

a =

4 : m

10

【 0 0 3 0 】

「unit」関数は、単位情報を、単位を持つ値に与えるのに使用できる。次に一例を挙げる。

a = unit (3, 'm');

[u, e] = unit(a)

20

u =

'm'

e =

1

30

【 0 0 3 1 】

他の使用可能な関数は本開示を参照すれば当業者には明らかなはずである。幾つかの実施形態では、アプリケーションプログラム・インターフェース(API)を用いて、単位データベース又はレジストリに単位を追加したり、削除したりできる。

【 0 0 3 2 】

上述の技法を用いれば、システム又は領域(原語: domains)を定義できる。例えば、電気領域を定義する関数を書くことができる。こうした関数は次のようになるはずである。

40

function electrical (root, domain)

%電気・電気領域を定義する

\$Author: batserve \$

v = domain.across_variable ('v');

v.dimension = [1 1];

v.unit = 'V';

10

i = domain.through_variable ('i');

i.dimension = [1 1];

i.unit = 'A';

20

end

% [EOF] electrical.m

30

【0033】

この例では、電圧(v)と電流(i)がその領域に関して定義される。電圧と電流は、それぞれボルト(V)とアンペア(A)の単位を持つ配列として定義される。上述の関数は多数の可能な関数のうちの1つにすぎず、指定可能な多数の領域の1つにすぎないことは明らかにはずである。他の使用可能な関数及び領域は本開示を参照すれば当業者には明らかにはずである。

【0034】

同様に、物理システムでモデル化する構成要素は固有の値を使って指定してもよい。例えば、抵抗器のような構成要素を固有の単位を使ってテキスト形式で指定できる。こうした例は次のようになるはずである。

40

```

function resistor (root, schema)
%抵抗器・線形抵抗器をモデル化する

p = schema.terminal ('p');
p.description = 'Positive Terminal';
p.domain      = root.electrical.electrical;

10
n = schema.terminal ('n');
n.description = 'Negative Terminal';
n.domain      = root.electrical.electrial;

r = schema.parameter ('r');
r.description = 'Resistance';
r.type        = ne_type ('real', [1 1], 'Ohm');

schema.setup(@setup);
20

end

function setup (root, resistor)
resistor.branch_across ( 'v', resistor.p.v, resistor.n.v);
resistor.branch_through ('i', resistor.p.i, resistor.n.i);

resistor.equation(@equation);
end
30

function e = equation (resistor)
e(1) = resistor.v - resistor.r * resistor.i;
end

% [EOF] resistor.m

```

40

【 0 0 3 5 】

この例では、抵抗器がテキスト形式で指定され、正端子(p)と、負端子(n)と、固有単位であるオームを持つ抵抗(r)とを備えている。この抵抗器の両端には電圧(v)が印加され、電流(i)が流れる。電圧と電流と抵抗との相互関係は、式 $e(1) = resistor.v - resistor.r * resistor.i$ で与えられる。この式で、変数は $e(1)$ はゼロに駆動される。言い換えれば、この抵抗器両端の電圧は、抵抗器を流れる電流をその抵抗に掛けたものに等しい($v = r * i$)。

【 0 0 3 6 】

上述の関数は、指定できる多数の可能な構成要素又は要素のうちの1つにすぎず、指定できる多数の可能な関数の1つにすぎないことは明らかにはずである。他の使用可能な関

50

数、構成要素、又は要素は本開示を参照すれば当業者には明らかなはずである。

【0037】

本発明の別の様態では、固有の単位を実装することによって固有の計量単位を検査してこれらが同一基準かどうかを判定できる。値は固有の計量単位を持ちうるので、演算が計量単位を持つ所与の値に関して実行できるかどうかを判断することが重要である。例えば、上述の関数pm_unitsは長さと時間の次元を、それぞれ固有の単位であるメートル(m)と秒(s)を持つものとして指定している。これらの次元は異なる固有の計量単位(メートルと秒)を備えており、従ってある種の演算では必ずしも互いに適合しない。これをどのように実行するかを示す代表的なフローチャート300を図3に示した。ここでは、実行する演算が特定される(310)。この演算は、加算、減算、乗算、又は除算などの単純な演算でもよいし、変換でもよい。続いて、与えられた演算が同一基準の単位を必要とするか判断する(320)。全ての演算が同一基準の単位を要件とするわけではないことは明らかなはずである。例えば、乗算及び除算は値の単位が同一であることを要件とはしない。幾つかの実施形態では、同一基準の単位を必要とする演算はデータベースのリストイングを調べれば判断できる。演算が同一基準の単位を必要としない場合は、この演算を実行できる(330)。演算が同一基準の単位を必要とする場合は、オペランドの単位を調べて同一基準かどうかを判断する(340)。例えば、時間の値と距離の値は同一基準ではないので、これらの値の加算はできない。幾つかの実施形態では、単位をチェックするにはデータベースを調べて互いに適合しているかどうかを判断する。例えばメートルとメートルを加算する場合のように単位が同一基準であれば、ここで演算を実行できる(350)。単位が同一基準でなければ、通知を生成できる(360)。動作時には、ユーザが次のように時間と距離との加算を試みたと想定する。

10

20

20

```
a = cunit(4, 'm');
b = cunit(5, 's');
c = b + a
```

30

【0038】

すると、ユーザは、これら単位が同一基準でないことを示すメッセージを受け取ることになる。

??? 単位「m」と「s」は同一基準ではない

40

【0039】

このメッセージは、単位を調べて所望の演算に関しこれら単位が同一基準となっているかを確認する必要があることをユーザに知らせる。幾つかの実施形態では、ユーザは同一基準でない単位の対処法に関する入力要求を受け取るようにもよい。上述のように、全ての演算が同一基準の単位を要件とするわけではない。例えば、距離(m)を時間(s)で割ると、メートル毎秒(m/s)単位の速度が得られる。

【0040】

本発明の別の様態では、固有の単位を実装することで固有の計量単位の変換も可能となる。これをどのように実行するかを示す代表的なフローチャート400を図4に示した。この処理は図3に示したステップ330又は350の一部として実行してもよい。先ず、単位の変換が必要かどうかを判断する(410)。変換が不要であれば、システムは所与の単位で処理を続行する(420)。単位の変換が必要な場合は、変換のための基本単位を特定する必要がある。するとこれら単位が変換され(440)、システムは変換した単位で処理を続行できる(45

50

0)。

【 0 0 4 1 】

幾つかの実施形態では、単位を尺度変更する必要がある場合もある。例えば、上述の関数pm_unitsは、長さの次元を固有の単位であるメートル(m)を持つものと指定している。関数pm_unitsは、センチメートル(cm)の単位を0.01メートルと指定している。メートルで与えられた値をcmで与えられた値に加算するには、これら単位の一致するように何れかの値を単位が尺度変更する必要がある。幾つかの実施形態では、この変換を演算の第1オペランドに基づいて行ってもよい。例えば、尺度変更を行う一実施形態では、例えばメートル(m)とセンチメートル(cm)を次のように加算する。

10

```
a = cunit (4, 'm');  
b = cunit (5, 'cm');  
c = a + b
```

すると次のような結果をメートル(m)で与える。

20

c =

4.05 : m

30

【 0 0 4 2 】

ここでは、単位は基本単位であるメートルに変換される。異なった単位だが同一基準の単位の複数オペランドが存在するときにも、この方法を適用できる。次に一例を挙げる。

```

a = cunit (4, 'm');
b = cunit (5, 'cm');
c = cunit (2, 'km');
d = a + b + c

```

d =

2004.05: m

【 0 0 4 3 】

単位の尺度変更は第1オペランドに基づく必要はない。幾つかの実施形態では、ユーザは、他の全ての単位がそれに合わせて尺度変更される基本単位を指定してもよい。他の実施形態では、単位の変換は1つの測定尺度から別の測定尺度への変換でよく、例えばポンド(lb)からキログラム(kg)へのようにイギリス単位系からメートル法への変換がある。例えば、ユーザは全ての単位をセンチメートル(cm)に尺度変更するよう指定できる。

【 0 0 4 4 】

他の実施形態では、単位の変換は1つの測定尺度から別の測定尺度への変換でよく、例えばポンド質量(lbm)からキログラム(kg)へのようにイギリス単位系からメートル法への変換がある。例えば、質量はキログラムで与えられる場合が最も多いので、ユーザは全ての重量計算がキログラムで与えられるように指定できる。(kg)従って、単位データベースにおいてポンド質量(lbm)がキログラム(kg)単位で指定されていると仮定すると、ポンド質量、グラム、キログラムの値を持つ質量を加算した場合、結果はキログラム(kg)で与えられることになる。

【 0 0 4 5 】

更に別の実施形態では、浮動小数点計算による丸めの誤差が減少するように単位の変換を最適に選択できる。従って、全ての単位は、実行されうる任意の浮動小数点計算による丸めの誤差が最小となる基本単位に変換される。

【 0 0 4 6 】

上述及び他の変換技法を互いに組み合わせて使用できることに注目されたい。幾つかの実現例では、特定の変換技法に他のものより高い優先順位を付けてもよい。例えば、所望の基本単位(例えばkg)への変換よりも、丸めの誤差の最小化を優先してもよい。

【 0 0 4 7 】

従って、このように固有の計量単位はシステム内を伝播できる。すなわち、値に計量単位が割り当てられると、その計量単位はその後に実行される全ての演算に反映される。これは、固有の値が物理システムのグラフィックモデルで使用されると特に明らかとなる。

【 0 0 4 8 】

固有の計量単位を持つ物理システムのグラフィックモデル500の一例を図5に示した。このグラフィックモデルは、素子A、B、及びC(510a、510b、510c)を備えている。素子A 510aは、電源520aと電圧センサ530aとに接続されている。素子B 510bは、電源520bと電圧センサ530bとに接続されている。素子A 510aに接続された電圧センサ530aは、ミリボルト(mV)単位の電圧を持つ信号540を出力する。素子B 510aに接続された電圧センサ530bは、ボ

ルト(V)単位の電圧を持つ信号550を出力する。信号550の電圧は微分器560で信号540の電圧から減算される。信号550の電圧を信号540の電圧から減算するに当たって、信号550のボルト(V)はミリボルト(mV)に変換される。信号540と信号550との電圧差は、ミリボルト(mV)単位の信号570で出力される。信号570のミリボルト(mV)は、ボルト(V)に変換され素子C 510cに電力供給する。

【0049】

この例からも分かるように、計量単位580をグラフィックモデルで示すことができる。ここでは、電圧がシステム内を伝播し、システムに含まれる素子の要件に応じて変換される。各素子は指定された基本単位580を備え、グラフィックモデル内で例えばミリボルト(mV)又はボルト(V)として表示できる。従って、素子が受け取る同一基準の単位は指定された基本単位に変換される。本図に示したように幾つかの実施形態では、自動変換が実行されることを示すグラフィック表示590を含めてもよい。本図では、自動変換は微分器560及び電源520cの入力ポートに590で示した。他の実施形態では、他の全ての同一基準単位の変換結果となる基本単位を、システム全体に関してグローバルに指定したり、参照モデル(原語: referenced models)又はサブシステムに関してサブグローバルに指定したりできる。

10

【0050】

与えられた又は指定された単位が同一基準ではない幾つかの実施形態では、システムは非同一基準単位の存在を示す通知を生成又は他の方法で与えることができる。例えば、図5で、電圧センサ530bがアンペア(A)単位の電流を持つ信号を出力する電流センサと取り替えられたとすると、ユーザは図6に示した通知600を受け取るようにしてよい。ここでは、通知600は、微分器560がアンペア(A)単位の電流をミリボルト(mV)単位の電圧から減算できないので、「mV」と「A」という単位は同一基準でないことを示すウィンドウである。幾つかの実施形態では、ユーザは同一基準でない単位の対処法に関する入力要求を受け取るようにしてよい。

20

【0051】

単位の伝播と、その後に行う単位が同一基準かのチェックとは、物理システムのモデルの実行に当たってエラーを取り除く助けとなるので有利である。値に計量単位が割り当てられていなければ、生成された結果がシステムにとって妥当でないか、またなぜ妥当でないかを知ることが極めて困難となりうる。

30

【0052】

これまでの説明において、この点に関する例は、本発明のグラフィカルモデリング環境がローカル計算装置にあるシステムに主として焦点を当ててきた。本発明のグラフィカルモデリング環境は、図7に示したように、サーバ704とクライアント装置702とを備えたネットワーク700上でももちろん実装可能である。記憶装置706のような他の装置もこのネットワークに接続できる。

30

【0053】

こうした一実施形態では、グラフィカルモデリング・アプリケーションを生成し且つ表示するシステムは、物理システムをモデリングするための固有の計量単位をクライアント装置に与える配布サーバと、該配布サーバと通信するクライアント装置とを含む。ここでは、配布サーバは、上述の計算装置のようなクライアント装置に固有の計量単位を与える。これら固有の単位は、サーバ上でクライアントが利用可能なデータベースの一部としてもよい。これら固有の単位は、サーバ上でクライアントが利用可能なブロック又は他のグラフィック要素の一部としてもよい。こうしたブロック又はグラフィック要素は、固有の計量単位を持つ入力を受け取り且つ/又は固有の計量単位を持つ出力を与えるよう構成してもよい。すると、クライアントは物理システムをモデリングするためにそれら固有の単位を使用できる。

40

【0054】

別の実施形態では、サーバはこの物理モデリング環境を実行できる。するとユーザは、クライアント装置を介してサーバ上の物理モデリング環境と対話できる。こうしたシステ

50

ムの一例では、サーバとクライアント装置が提供される。サーバは物理モデリング環境を実行可能であり、この物理モデリング環境は物理システムをモデリングするための固有の計量単位を与える。クライアント装置はネットワークを介してサーバと通信する。次に、固有の計量単位を持つ値が、物理システムのモデルで使用するためクライアントからサーバに与えられる。こうしたシステムが次にモデル化され、結果がサーバからクライアント装置へ出力される。

【0055】

当業者であれば、これらネットワークの実施形態は代表的な例に過ぎず、その機能性をネットワークにおいて多くの方法で分割してよいことは理解するはずである。

【0056】

上述の例は電気領域の物理システムのモデルに主として焦点を当てたが、本発明が適した多数の領域、システム、分野、及び応用例が他にも存在することは当業者には明らかなはずである。これらの例としては、機械及び化学領域が含まれるがそれらに限定されない。

【0057】

ここまで例示的な実施形態に関連して本発明を説明してきた。本発明の範囲を逸脱することなく上記の構成に対して幾つかの変更を施すことが可能であるから、この説明に含まれ或いは添付の図面に示された全ては、例示的なものとして解釈されるべきであり、限定的な意味で解釈されるべきではない。

【0058】

更に、以下の特許請求の範囲は、ここに説明された本発明の全ての一般的特徴と具体的特徴とを網羅するものであり、また本発明の範囲に関する全ての説明をも網羅する。

【図面の簡単な説明】

【0059】

【図1】本発明の例示的な実施形態を実施するのに適した環境を示す。

【図2】物理モデリング環境において固有の計量単位を実現する処理の代表的なフローチャートである。

【図3】固有の計量単位が同一基準かどうかを調べる処理の代表的なフローチャートである。

【図4】固有の計量単位を変換する処理の代表的なフローチャートである。

【図5】固有の計量単位を持つ物理システムのグラフィックモデルの一例である。

【図6】オペランドが同一基準でない場合に生成される代表的な通知である。

【図7】本発明の例示的な実施形態を実施するのに適したクライアント・サーバ環境を示す。

10

20

30

【図1】

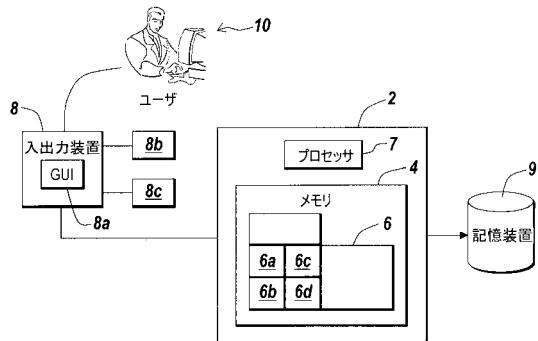


Fig. 1

【図2】

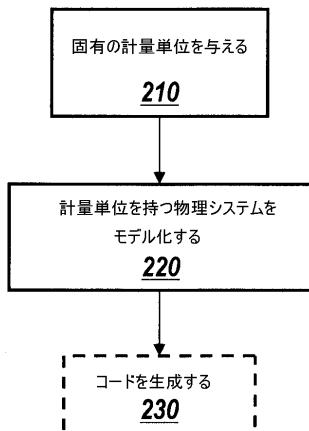


Fig. 2

【図3】

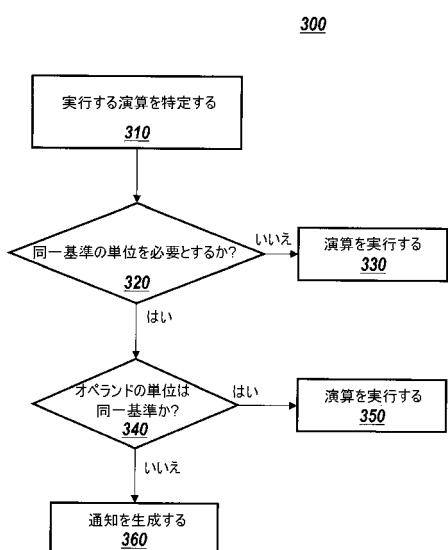


Fig. 3

【図4】

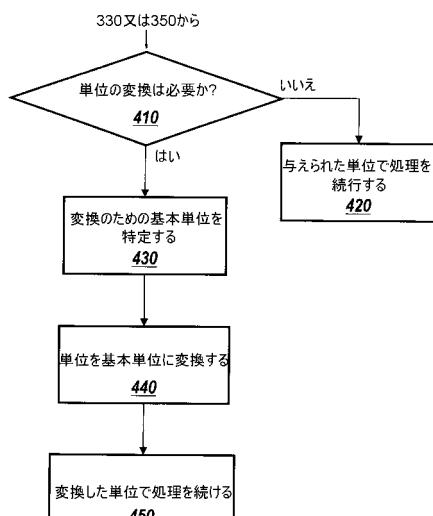
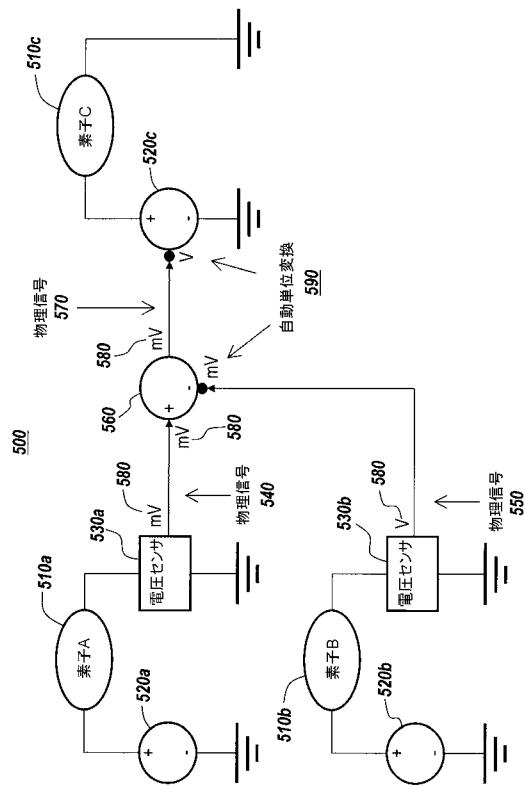


Fig. 4

【図 5】



【図 6】

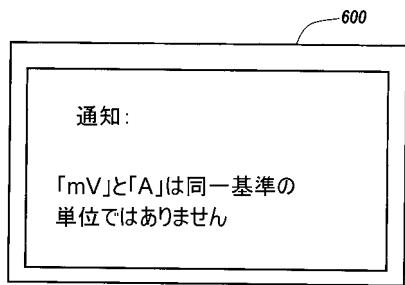


Fig. 5

Fig. 6

【図 7】

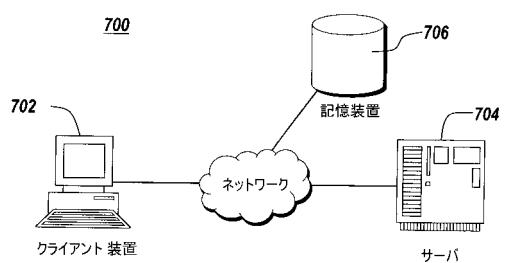


Fig. 7

【国際調査報告】

INTERNATIONAL SEARCH REPORT		International application No PCT/US2006/034625												
A. CLASSIFICATION OF SUBJECT MATTER INV. G06F17/50														
According to International Patent Classification (IPC) or to both national classification and IPC														
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) G06F														
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched														
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal														
C. DOCUMENTS CONSIDERED TO BE RELEVANT <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 2px;">Category*</th> <th style="text-align: left; padding: 2px;">Citation of document, with indication, where appropriate, of the relevant passages</th> <th style="text-align: left; padding: 2px;">Relevant to claim No.</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">X</td> <td style="padding: 2px;">DOMNISORU C: "Using MATHCAD in Teaching Power Engineering" IEEE TRANSACTIONS ON EDUCATION, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, Vol. 48, no. 1, February 2005 (2005-02), pages 157-161, XP011126796 ISSN: 0018-9359 in particular: Abstract and first paragraph, left column on page 158 the whole document</td> <td style="padding: 2px;">1-22, 27, 28</td> </tr> <tr> <td style="padding: 2px;">Y</td> <td style="padding: 2px;">----- -----</td> <td style="padding: 2px;">23-26</td> </tr> <tr> <td style="padding: 2px;"> </td> <td style="padding: 2px;">-----</td> <td style="padding: 2px;">-/-</td> </tr> </tbody> </table>			Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.	X	DOMNISORU C: "Using MATHCAD in Teaching Power Engineering" IEEE TRANSACTIONS ON EDUCATION, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, Vol. 48, no. 1, February 2005 (2005-02), pages 157-161, XP011126796 ISSN: 0018-9359 in particular: Abstract and first paragraph, left column on page 158 the whole document	1-22, 27, 28	Y	----- -----	23-26		-----	-/-
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.												
X	DOMNISORU C: "Using MATHCAD in Teaching Power Engineering" IEEE TRANSACTIONS ON EDUCATION, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, Vol. 48, no. 1, February 2005 (2005-02), pages 157-161, XP011126796 ISSN: 0018-9359 in particular: Abstract and first paragraph, left column on page 158 the whole document	1-22, 27, 28												
Y	----- -----	23-26												
	-----	-/-												
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C.		<input checked="" type="checkbox"/> See patent family annex.												
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier document but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed														
Date of the actual completion of the international search: 23 February 2007		Date of mailing of the international search report: 06/03/2007												
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016		Authorized officer: Lerbinger, Klaus												

INTERNATIONAL SEARCH REPORT

International application No PCT/US2006/034625

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	<p>HUMAR I ET AL: "Mathematical tools for supporting web-based education of electromagnetics"</p> <p>INFORMATION TECHNOLOGY BASED HIGHER EDUCATION AND TRAINING, 2004. ITNET 2004. PROCEEDINGS OF THE FIFTH INTERNATIONAL CONFERENCE ON ISANBUL, TURKEY 31 MAY - 2 JUNE 2004, PISCATAWAY, NJ, USA, IEEE, 31 May 2004 (2004-05-31), pages 111-116, XP010741665</p> <p>ISBN: 0-7803-8596-9</p> <p>In particular section II. B. The MATLAB Web Server the whole document</p>	23-26
A	<p>BACHMAN B ET AL: "Modelica - A Unified Object-Oriented Language for Physical Systems Modeling - Tutorial and Rationale"</p> <p>MODELICA - A UNIFIED OBJECT-ORIENTED LANGUAGE FOR PHYSICAL SYSTEMS MODELING. TUTORIAL AND RATIONALE, 15 December 1999 (1999-12-15), pages 1-49, XP002170770</p> <p>section '4.9 Units and Quantities'</p>	1-28
A	<p>US 2004/068528 A1 (ERICKSON RANEL EINAR [US]) 8 April 2004 (2004-04-08)</p> <p>the whole document</p>	1-28

INTERNATIONAL SEARCH REPORT			
Information on patent family members			International application No
Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004068528	A1 08-04-2004	NONE	

フロントページの続き

(81)指定国 AP(BW,GH,GM,KE,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HU,IE,IS,IT,LT,LU,LV,MC,NL,PL,PT,RO,SE,SI,SK,TR),OA(BF, BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BW,BY,BZ,CA,CH,CN,CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, L C, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG ,SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW

(72)発明者 プリュートン, ネイサン, イー.

アメリカ合衆国 マサチューセッツ州 01778 ウェイランド ローリング レーン 36