

遺伝的アルゴリズムによるワークフローの最適化について : システムの現状と業務改善の課題

高木, 昇

<https://doi.org/10.15017/3000246>

出版情報 : 経済論究. 107, pp.69-76, 2000-07-31. Kyushu Daigaku Daigakuin Keizaigakukai
バージョン :
権利関係 :

遺伝的アルゴリズムによるワークフローの最適化について

——システムの現状と業務改善の課題——

高 木 昇

述べ例題を与えている。

1. まえがき

Windowsを搭載した低価格なPCやUNIXは、開発・生産分野のみならず事務的な管理業務にまで急速に浸透している。管理業務の電子化については、業務を部門間の流れとしてとらえ、管理と効率化について論じる「ワークフロー(workflow)理論」が注目を集めており、その理論を応用した管理システムはすでに大規模な企業において不可欠なインフラとして稼働中である。また、今後は帳票の保管方法や企業間の決済方法等にネットワーク時代に合わせた大きな変化が予想されることから、管理業務におけるデジタル・データの直接的な活用が促進されることとなり、それに伴って業務管理システムについての議論が高まると考えられる。

本論文では、現在稼働中のワークフローによる業務管理システムを概観し、その性格・機能について調査することによって、業務管理システムの現状を把握する。次に、問題点を述べ、ワークフロー理論の概念を拡張して自己組織的に進化する業務管理システムのプロトタイプについて述べる。特に、この場合、GA (Genetic Algorithm, 遺伝的アルゴリズム)を用いたワークフローの最適化について方法論を展開する。以下、2.ではワークフロー理論の定義と分野について整理し、3.ではワークフロー管理システムの現状について述べる。4.ではワークフローの改善課題とそれによる最適化の方法について

2. ワークフロー理論の定義と適用対象

ワークフロー理論とは、業務(ワーク)を部門間の流れ(フロー)としてとらえ、効率のよい経路について議論することで、業務の管理と効率化に言及する理論であると定義される。この理論に基づいた管理システムが登場したのは1980年代前半の英国であったが、現在までに多くのシステムが開発・販売されている。ワークフロー理論の対象となる業務には、電子的に取り扱われるという制約から、業務処理の経路と順序が明確であるという共通点がある。また、処理の担当者について、その役割と処理上の規則や条件についても明確である。

通常の企業業務は一般にこのような点が満たされているので、すでにワークフロー管理システムによって業務管理の効率化が図られた実績は枚挙に暇がないが、その一例として、表1に示す。今後もインターネットによる情報交換、ネットワークによる企業間決済が進展するにつれて、その適用対象も大きく増加すると考えら

表1 ワークフロー管理による業務改善の例

流通業	製造業	保険業	金融業
受注発注 運送追跡	審査業務 クレーム 処理 予算処理	査定業務 契約業務 返済業務 支払業務	査定業務 契約業務 返済業務 支払業務 クレジット

れる。

3. ワークフロー管理システムの現状

ワークフロー理論の適用対象が多岐に渡っていることから、実際に稼動しているワークフロー管理システムは対象となる業務にそれぞれ適応したものとして稼動しているが、システムの規模と性格から3つに大別することができる。ここでは、種類別に代表的な製品を幾つか取り上げ、その目的や特徴についてWWW及び試用版を用いて調査し[12]-[15]、ワークフロー管理システムの動向について考察する。

3.1 スタンド・アロン型

ワークフロー管理専用に開発されたソフトウェアであり、処理案件の経路の設計・管理を柱に、電子メールや回覧版の配信、個々人のスケジューリング、及び他のソフトウェア等と連動して管理者が部署間の業務の流れを監督する機能を持つ。開発は限られた内外の大手情報産業によって行われるが、販売はソリューションと呼ばれ、単なるソフトウェアの販売ではなく、代行企業・事業部と導入クライアントとの綿密なコンサルティングによる導入計画から導入後のコンサルティングまで一貫したサービスの一部としての性格を持つ。そのため、導入に要する時間は約数ヶ月、価格は数千万が相場であるとされている。表2-1、2-2に主な製品について、その開発元、動作環境、特徴についてまとめる。可視環境とは、図1に示すような図案化された操作画面をマウスでコントロールすることで、ワークフローの設計から日常の管理までの操作が行えるものである。現在のPC環境に適したインターフェースを取り入れることで、操作の習得に要するコスト（講習会等）も削減

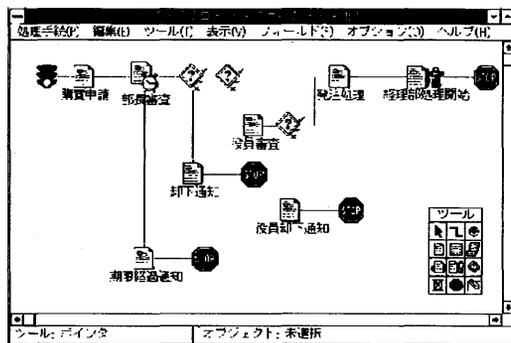


図1 可視環境の一例

できたと考えられる。

逐次処理追跡とは、処理案件が現在の担当者に到達しているかを常時表示することで、進行状況をリアルタイムに把握できるようにするものである。処理案件を受け取った担当者はその処理を行うに当たって他のアプリケーションソフトを使用することがあるが、そうした場合に適したアプリケーションを自動的に設定済みで発ち上げる機能が外部アプリ連携機能である。シミュレーション機能とは、管理者がワークフローの設計を行うに当たって、設計案の動作確認を行う場合に使用する。設計案を仮想的に動かすことで、非論理的なワークフローやデッドロック等の不都合を発見することができる。動作環境については、いずれも標準的なOSの複数をサポートしており、事実上プラットフォーム独立であるといつて差し支えない。尚、全ての製品でインターネット経由でのアクセスが可能である。携帯型情報端末等を利用して外部からワークフロー管理ソフトウェアにアクセスし、処理の進捗に関する各種情報を更新することができる。この機能は、後述するアド・オン型、グループウェアにも認められる。

表 2-1 主な特徴

開発・販売	主な特徴
IBM	可視環境, 逐次処理追跡, 外部アプリ連携, シミュレーション
Olibetti	可視環境, 逐次処理追跡, 外部アプリ連携, シミュレーション
NEC	可視環境, 逐次処理追跡, 外部アプリ連携, シミュレーション
富士電気	可視環境, 逐次処理追跡, 外部アプリ連携
日本UNISYS	可視環境, 逐次処理追跡, 外部アプリ連携

表 2-2 動作環境

開発・販売	主な特徴
IBM	Windows3.1, 95, 98, NT, Mac, UNIX
Olibetti	Windows3.1, 95, 98, NT, UNIX
NEC	Windows3.1, 95, 98, NT
富士電気	Windows3.1, 95, 98, NT
日本UNISYS	Windows3.1, 95, 98, NT, Mac, UNIX

3.2 アド・オン型

MicrosoftやLotus等が販売しているオフィスワーク用ソフトウェアの機能を強化する製品として販売されている。母体ソフトウェアの束縛を受けることから、機能的にはスタンド・アロン型に及ばず、メールの自動配信や印鑑イメージ等の枝葉的な機能強化にとどまっている。その分開発コストも低くすみ、導入コストも独立型より安価である。母体ソフトウェアがすでに稼働中のクライアントにとっては、欲しい機能が入手でき、かつ操作環境が大きく変わらないという利点もある。

しかしながら、母体ソフトウェアの改良による陳腐化・非効率化が予想され、専用に開発された大規模な独立型ソフトウェアを退けて主流の地位を築けるかには疑問が残る。なお、印鑑

表 3-1 主な特徴

開発・販売	主な特徴
CIS	可視環境, 自動メール配信, 逐次追跡, 自動メール配信
IBM	可視環境, 印鑑イメージ, 自動メール配信
Eastman	可視環境, 自動メール配信
京セラ	可視環境, 印鑑イメージ, 代理人決済

表 3-2 動作環境

開発・販売	動作環境
CIS	Windows95, 98, Mac
IBM	Windows95, 98, NT
Eastman	Windows95, 98, NT
京セラ	Windows95, 98, NT, Mac

イメージとは、担当者が処理を官僚した場合に処理案件に対して印鑑の画像を電子的に印刷するものであり、日本の企業風土の要請によって生まれたものである。

3.3 グループウェア

ワークフロー管理システムと似た概念を持つ業務管理システムがグループウェアである。ワークフローが担当者間の業務の流れを管理することを目標としているのに対し、グループウェアでは複数での共同作業の管理を目標としている。業務の効率化という目標をワークフロー管理ソフトウェアと共有していることから、現在では多くのグループウェアでワークフロー管理が可能となっている。従って、グループウェアの製品としての規模も巨大であり、e-ビジネス等の登場と相まって、ワークフロー管理システムとの明確な住み分けは意味を成さなくなりつつある。

3.4 調査の総括

表 4-1 主な特徴

開発・販売	主な特徴
NEC	可視環境, ワークフロー関連, 外部アプリ連携, 検索エンジン
富士通	可視環境, ワークフロー関連, 外部アプリ連携
日立	可視環境, ワークフロー関連, 外部アプリ連携, 検索エンジン

表 4-2 動作環境

開発・販売	動作環境
NEC	Windows95, 98, NT, UNIX
富士通	Windows95, 98, NT, UNIX
日立	Windows95, 98, NT, UNIX

以上, 3種類のソフトウェアの代表例の機能を調査したが, いずれの製品も非常に類似していることが分かった。すなわち「可視環境下における, 場所に依存しない処理とシミュレーション機能」がワークフロー管理方法としてスタンダード化し熟成を迎えると考えられると同時に, 一つの限界に達しているとも考えられる。すなわち, メーリング等の自動化や可視化, WWW経由のアクセスにより管理方法が向上する一方で, より効率のよいワークフロー設計は依然として管理者の手作業によっている。シミュレーション機能はその場合に管理者を支援する事もあるが, 現在のところはデッドロック等の事前検証としての使用が想定されており, 効率化に直接繋がる指針を与えるものとはなっていない。

ワークフロー管理システムの目的は管理業務の効率化と自動化にあるので, ワークフローの最適化に能動的に機能するシステムの構築が, 今後の業務管理システムの進むべき方向の一つと考えられる。

4. GAによるワークフローの最適化

4.1 ワークフローにおける最適化問題

現在のワークフロー管理ソフトウェアでは, 処理状況を逐次追跡することができ, シミュレーション機能設計も持つことから, 設計案を仮想的に実行させ, 各処理作業における各種データが測定可能であることを示している。ワークフローの最適化に関する研究はすでに行われている [7]-[9]。

本論文では, 測定されたデータに対してGA (Genetic Algorithm: 遺伝的アルゴリズム) を適用し, システムに学習機能を持たせることで, ワークフローの自動最適化を行う方法について述べる。

まず, ワークフロー管理システムによって収集可能なデータとしては,

- 一回の処理, または処理全体に要する時間
- 処理が完了するまでの待ち時間
- スループット
- 処理上の不都合の発生, 信頼性
- 処理に要するコスト

が挙げられる。これらのデータは設計されたワークフローに固有のものであり, 数値パラメータとして表現される。すなわち, これらのパラメータを一組の集合体とし, その集合体の連鎖によって一つのワークフローが定義される。例えば, 融資の受け付け・審査・実際の融資に至るワークフローを考える場合, 受け付け担当, 審査担当, 融資担当のそれぞれにおいて, 前述の一組のパラメータがそれぞれに設定できる。審査担当における待ち時間を大きな値としてとれば, 顧客にとっては不都合であり, ワークフロー全体の処理時間としても不利となることがある。しかし管理者にとっては信頼性の面

で有利であり、ワークフロー全体としてのパフォーマンスにも有利に働くことが考えられる。逆に審査時間を短くとればその逆の現象がおこる。すなわちワークフロー全体の能力は個々のパラメータどうしの直接比較ではなく、ワークフロー全体としてのパフォーマンスを表す適合度を計算できると仮定する。すなわち、個々のパラメータを改善することが目的ではなく、ワークフロー全体のパフォーマンスを向上させるためにパラメータをどのように選択するか、という問題に帰着する。このような問題は一般にパラメータの数の増大に伴い複雑な非線形最適化問題となるので、陽に解くことはできない。そこで、本論文では、この問題をGAによって最適化する。

4.2 GAによる最適化の概念

GAにおいては、一組のパラメータの集合体を固体と呼び、ランダムに作成された固体全体の中から適合度に優れる複数の固体群に対して選択と交叉、突然変異を行うことでより高い適合度を持つ次世代の固体群を作り出す。この作業を多数回繰り返すことで、最終的には最高の適合度をもつ固体が作成される。この様子を図2に示し、具体的に解説する。

システムの状態はパラメータ $x_1 \sim x_n$ をもつ固体1～Nで表されている。これらを適合度順に並べ替え、適合度の高いグループと低いグループに二分する。適合度の高い固体32と固体3とで交叉処理を行い、新たに二つの固体をつくる。これを適合度の低い固体18、固体14と入れ替える。この作業を、適合度の低いグループが完全に更新されるまで行う。これが一回のGAの操作である。この操作を数十回繰り返し、解が局所的に収束しないように乱数で作成した固体を混入させたり、禁止条件（限度額以上の

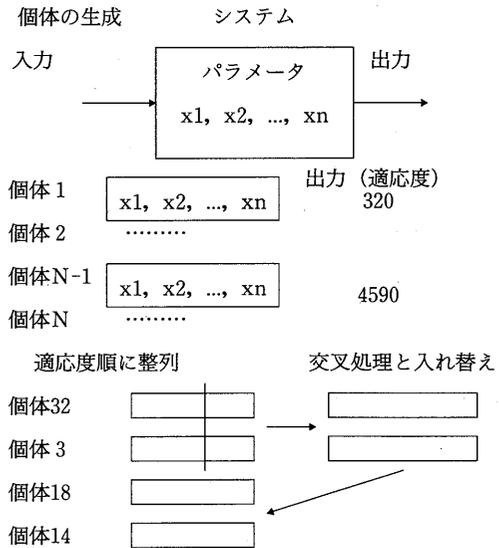


図2 GAによる最適化の概要

融資認可等)を与えている固体については適合度にペナルティを与える等の操作も行う。すなわち、GAは制約条件付きの非線形最適化問題にも対応できる。

4.3 GAによる最適化のシミュレーション (I)

一パラメータ最適化

ワークフローは、以下の4つのフローにより構成されていると考えられる。

- 順方向：一方通行の引き継ぎ型フロー
- 分岐：複数に分岐させるフロー
- フィードバック：

すでに通過した場所に返すフロー

- 同期：

平行するフローと連携して実行するフロー
このようなフローを全て含むワークフローの例として、図3に示すような組織を考える。この組織は直列型であり、フィードバックを含む定型的な申請、認可の業務等を1つの部門が行い、その部分の別の部門への依頼を含んでいるケースである。また、最終的には企画部門で結果が

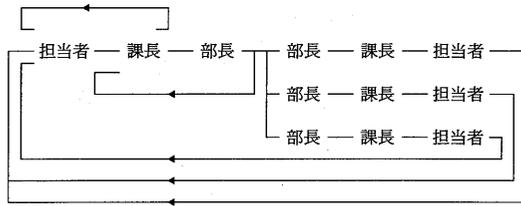


図3 ケースAのワークフローの説明

集約される。さらに、確認を慎重にするフィードバックも含んでいる。

このワークフローにおいて最適化すべきパラメータを次のようにする。

n_i : 各部門の処理能力

f_i : フィードバックの確率

GAでは、この二つを個体とする。すなわちこの二つは具体的に表現する数値の並び（ストリング）としてプールに格納される。このストリングは通常2進数で表現されるが、筆者らが実施した実験では10進数でもほとんど支障無く動作することが確認できている。従って、パラメータをそのまま10進数としてストリングに置き換えることにより、問題が簡単化できる。また、適合度には信頼性などを含めて拡張することも可能であるが、ここでは問題を簡単にするために処理時間と遅延時間のみを考慮する。図4にはGAの適用による適合度の変化を示してい

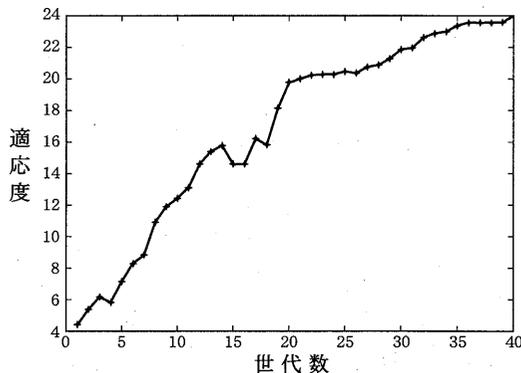


図4 GAによる適合度の変化

る。横軸は個体の世代数、すなわちGAの適用階数を表す。GAを約50回適用することにより、適合度が向上した状態に移行していることが分かる。

4.4 GAによる最適化の

シミュレーション (II) 一経路最適化

数段階の処理を通過させるフローの最適化を考えた場合、各段階での処理パラメータの最適化という観点のみならず、処理段階の簡略化や平行化といった方法でも最適化を行うことは可能である。すなわち、ワークフローの最適化は処理能力の最適化だけではなく、処理経路を最適化することによっても可能となる。しかしながら、最適化された経路が実現可能であるための制約条件が多岐にわたることから課題も多く、ここでは単純化されたモデルについて考察し、GAの経路最適化問題への適用の概要のみを記す。

フローは処理担当間を結ぶリンクと考えられることから、ワークフローの経路最適化問題はコンピュータネットワークにおける最短経路問題として考えることができる。すなわち、リンクの重みとして待ち時間やコスト等を処理担当に対して固有に与え、重みが最小となる経路を探索問題として考える。この問題に対してはDijkstraのアルゴリズムが有名な解法として知られているが、Dijkstraの解法ではフローが独立に最短経路を選択するので、ある時刻に待ち時間が少なかった経路に対して次の時刻にフローが集中してしまい、結局経路全体としての待ち時間が増大してしまうという問題がある。

そこで、以前に時刻の待ち時間を考慮して、GAにより「良好ではあるが最短でない」経路も作成しておく。作成に際しては処理パラメータの最適化の場合と同様に選択、突然変異、交叉

の各操作により求められるが、経路の場合は適合度として経路の長さをとる。突然変異は経路の始点と終点の間にある処理担当を単独で他の処理担当と交換する。交叉は、二つの経路に共通する処理担当を選び、その処理担当以降の経路を入れ替えることにより行われる。

経路最適化の一例として、以下のような問題を考える。処理を担当するノード（アクティビティ）の数を15と仮定し、それぞれのアクティビティには10個のフロー発生源が接続されているとする。その発生確率については省略する。

アクティビティから次のアクティビティへは3つ前後のリンクが出ているとする。1つのアクティビティから出るフローのリンクを平均が3である乱数であたえておく。発生するフローには最終的に消滅する場所が指定されており、流れは一方方向であると仮定する。この概要を図5に示す。

GAによる代替パスの生成とそれを用いたフロー分散の仕組みを図6に示している。図6では、最初にフローの発生ノードSからその終点ノードDに向かって最短パスを見出す。これをパスAとする。次に、このパスAの途中で任意に選択したノードIをとりこのノードIに隣接したノードJを1つ検出する。次に、これらのノードのペアS-J, J-Dの間で、それぞれ最短パスを求め、結合して1つのパスBとする。この場合、一般には最初の段階で全てのノードのペアに対して最短パスを計算しておくので、改めてS-J, J-Dの間の最短パスを計算する必要はない。

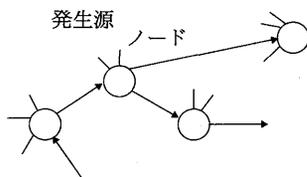


図5 ワークフローの発生とルーティングパス

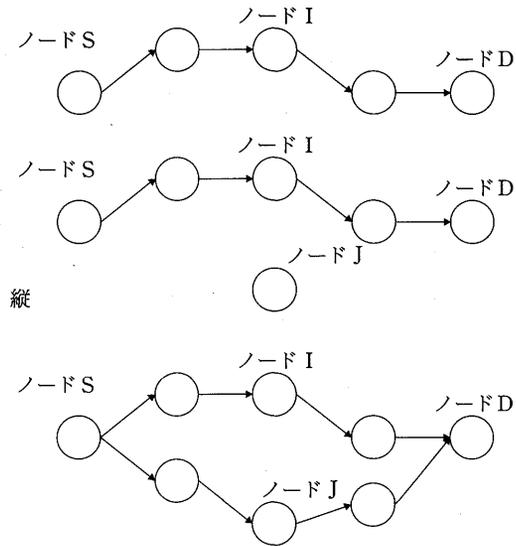


図6 GAによる代替パスの作成とフロー分散

これにより、もとの最短パスAと並んで代替パスBが見出せた。この2つのパスに対して、S-Dのフローの遅延時間に逆比例してフローを割り当てることにより、フローの集中化と遅延時間の抑制を、同時にはかることができる。

このようなワークフロー管理システムにおいて、フローの最短パスを用いた場合と、GAによる複数のパスを用いてフロー制御を行った場合との比較を図7に示す。縦軸はフローの平均遅延時間を示し、横軸は入力フロー率、すなわち加えられるフローをネットワークの処理容量で割った比率である。実線がGA無し、点線がGA有りの結果である。この結果より分かるように、最短パスのみを用いた場合に比べて、平均遅延が減少しており、GAによるパス発生が有効であることが示される。

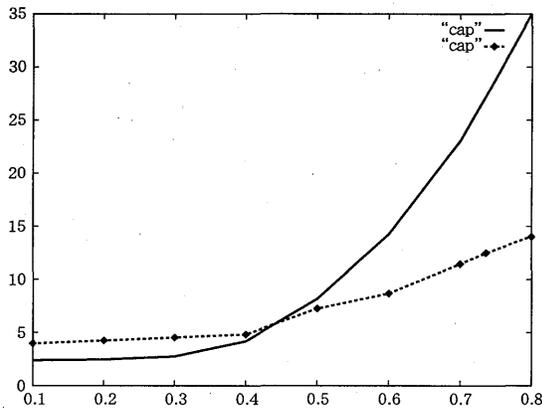


図7 GAによる代替パス利用による平均遅延時間の改善

6. まとめ

本論文の主旨は、ワークフロー管理システムの今後の課題の一つである自動最適化について、GAを用いることで達成できる可能性について言及することにあつた。

まず現在稼働中のワークフロー管理システムの現状を調査し、最適化に関して積極的な機能が搭載されていない、いずれも類似した製品であることを述べ、次にGAによる最適化の概念について述べた。さらに、ワークフロー最適化への適用方法としてパラメータ最適化と経路最適化の二通りの方法について述べた。

今後の課題としては、GUIをベースにしたワークフロー管理システムのプロトタイプを作成すること、実際の事例に適用して問題点を更に明確にすることがあり、検討を進めていく予定である。

参考文献

(1) 高木昇, 時永祥三: “ワークフロー管理システム

とBPR—現状と業務改善への応用”, OA学会誌, vol.20, No.3, pp.88-94, 1999

(2) 高木昇, 時永祥三: “ワークフロー管理システムとBPR—現状と業務改善への応用”, OA学会第40年全国大会予稿集pp.119-122, 1999

(3) 堀内正博, 飯島淳一: “ワークフロー管理システムの有効性について”, オペレーションズリサーチ, Vol.41, No.10, pp.559-568, 1996

(4) “ワークフローで仕事を変える” 日経コンピュータ1996.3.4号, 1996

(5) 横井康夫, 荒尾辰之: “ワークフローシステムによる新契約査定業務の改革”, 日立評論, Vol.78, No.5, pp.399-404, 1995

(6) D.E.Goldberg: Genetic Algorithm in Search, Optimization and Machine Learning, Los Altos, California, Morgan Kaufmann

(7) M.Richert and P.Dadam: “ADEPTflex-Supporting dynamic changes of workflows without losing control”, Journal of Intelligent Information Systems, Vol.10, pp.93-129, 1998

(8) B.Dellen, F.Maurer and G.Pews: “Knowledge-based techniques to increase the flexibility of workflow management”, Data & Knowledge Engineering, Vol.23, pp.269-295, 1997.

(9) F.Casati, S.Ceri B.Pernici and G.Pozi: “Workflow evolution”, Data & Knowledge Engineering, Vol.24, pp.211-238, 1998

(10) “Workflow tackles the productivity paradox”, Data Mation, 1995, Aug 15, pp.65-73 (1995)

(11) 大串葉子, 時永祥三: “ルール学習機能をもつワークフロー分析システムの構成”, オフィスオートメーション学会第37回全国大会講演論文集, pp.109-112 (1998).

(12) http://www.unisys.co.jp/product_info/staffware/

(13) <http://www.yamato.ibm.co.jp/manuals/web-book>

(14) <http://www.hitachi.co.jp/Prod/comp/soft1/gmax/index>

(15) <http://www.hitachi.co.jp/Prod/comp/soft2/teamstar/index-j.html>