

サーバベースコンピューティング におけるCOBOL活用

NECシステムテクノロジー

ビジネスソリューション事業本部

Webビジネス事業部 第一Webビジネス部

岡本 成一

－目次－

Server-based Computing の概要

COBOL開発・実行環境

システム構築のポイント

導入事例

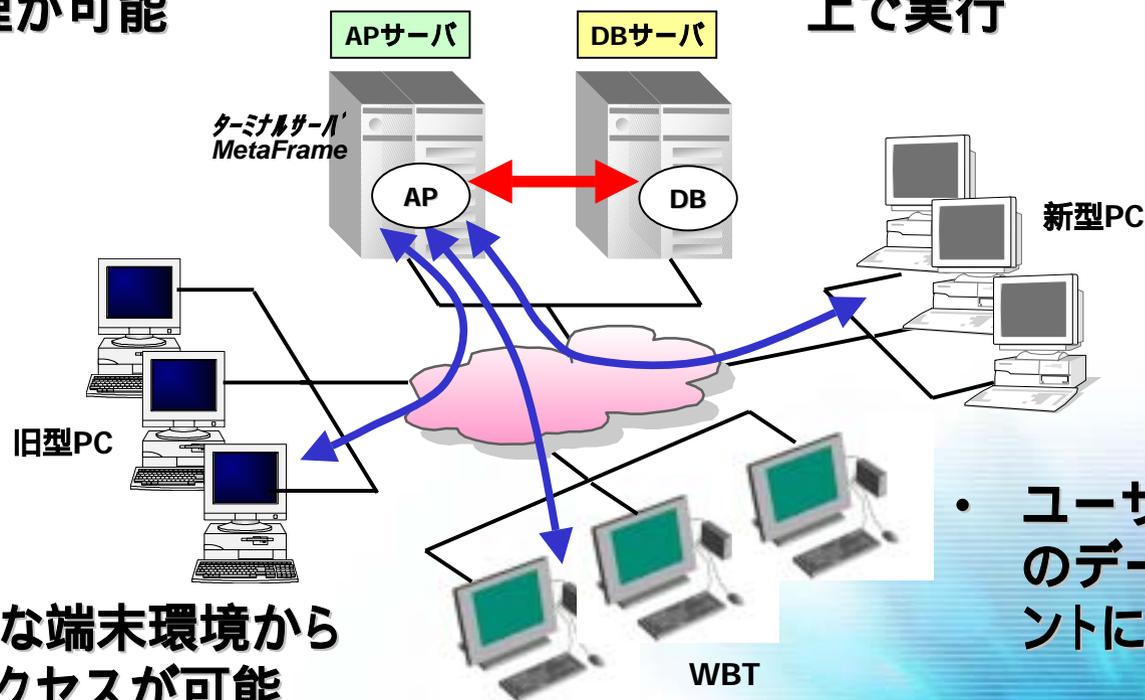
まとめ

Server-based Computing の概要

Server-based Computing とは？

クライアントのリソース(アプリケーション, データ)をサーバで実行, 管理を行うコンセプト

- シングルポイントでの制御・管理が可能
- アプリケーションは100%サーバ上で実行



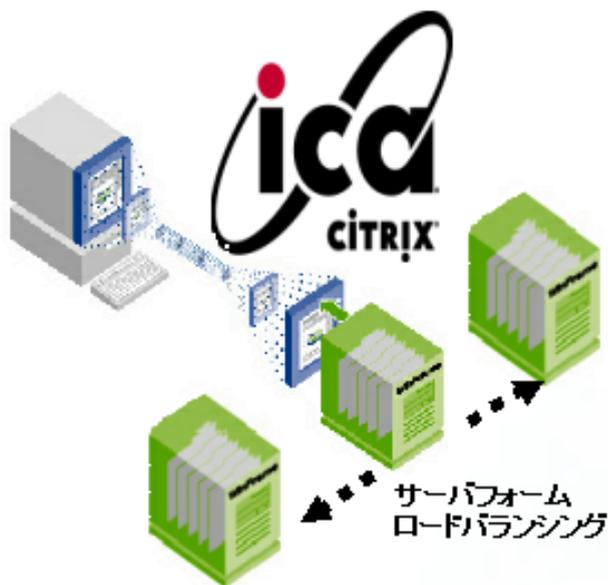
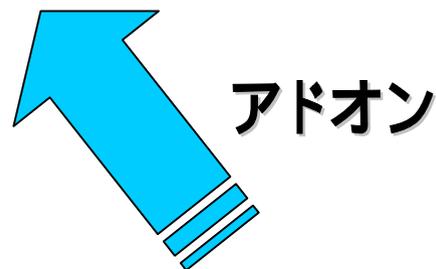
- 多様な端末環境からのアクセスが可能

- ユーザインターフェイスのデータのみをクライアントに送信

Server-based Computing を支える技術

• Windows 2000 Terminal Services

- マルチユーザ機能
- ユーザ個別のデスクトップ
- リモートサーバ管理



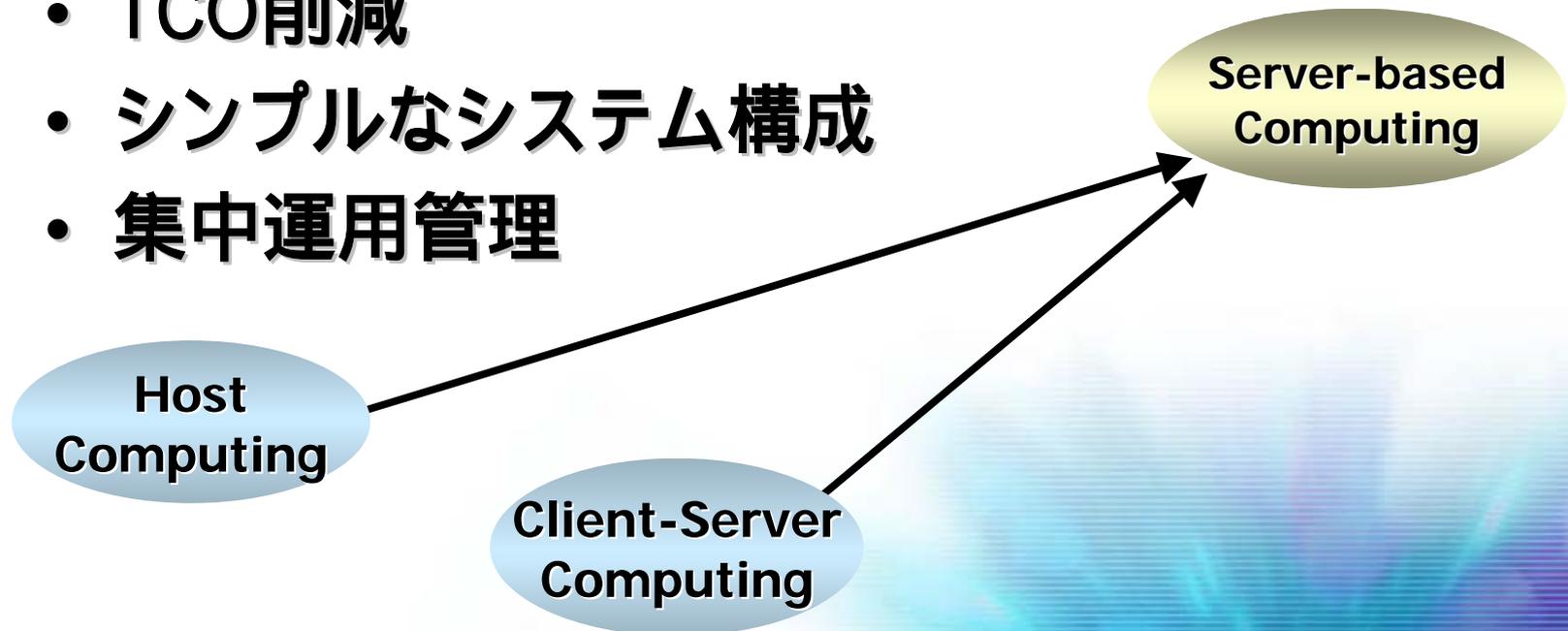
• MetaFrame

- 大規模システムのシステム管理
- 異機種混在システムの構築
- クライアントリソースの利用
- APのWeb対応によるポータル化

Server-based Computing のメリット

ホストコンピューティング、クライアントサーバコンピューティングのメリットを引き継ぎ、デメリットを解消

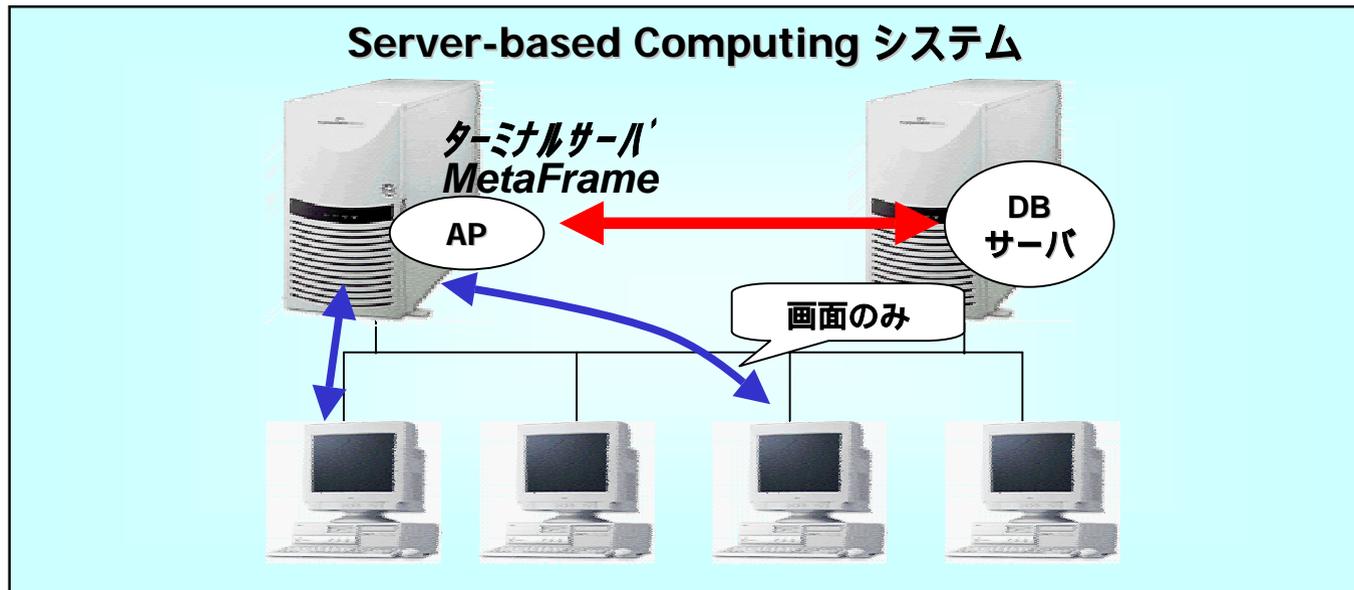
- 高速レスポンス
- TCO削減
- シンプルなシステム構成
- 集中運用管理



COBOL開発・実行環境

COBOL活用方法 (TS+MetaFrame)

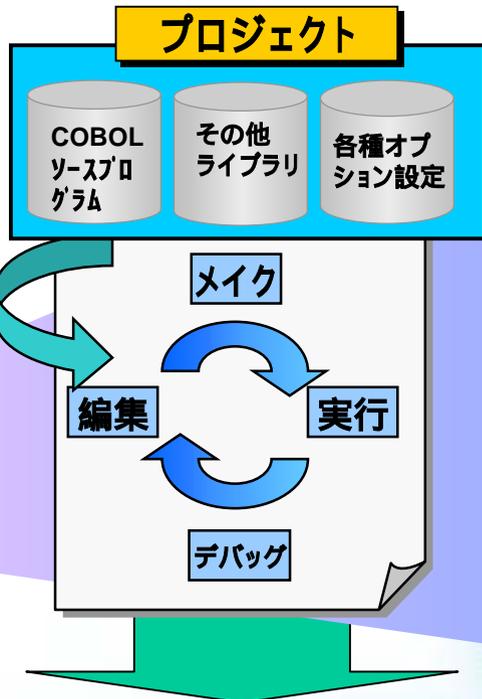
- 従来COBOL資産を活かし先進のオープン機能を追加
 - ホスト、オフコンの集中処理型でオープン化
 - クライアント・サーバ型システムのメリットを享受
 - TCO削減(開発コスト、初期コスト、運用コスト)



COBOL開発・実行環境

- ・オープン系COBOL統合開発環境 - Open COBOL Factory 21
- ・従来資産のオープン化を支援するミドル製品群
 - AP実行環境、IFASPRO、DBリンクキット...etc

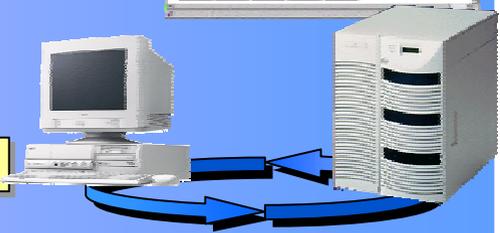
COBOL85 Pro



Web AP



分散 AP



GUI AP

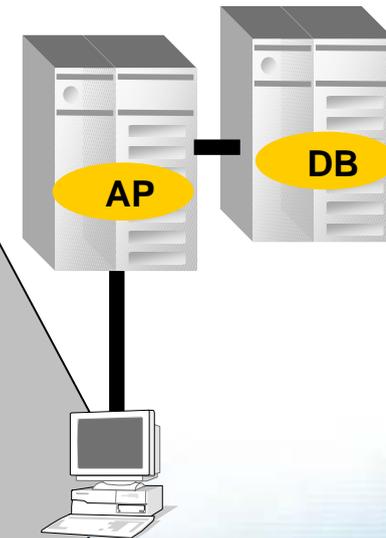


Windows 2000ターミナルサービス + MetaFrameを利用したサーバベース型のシステム構築が可能

COBOLアプリケーション実行例

- 従来資産を簡易GUI化機能でマイグレーション & TSクライアントでの実行例

型番	製品名	単価	発注数	合価	在庫数
N8100-591B	Express 5800/120 Md(Pen3X1.00G	06320000			
N8100-592B	Express 5800/140 Hb(Pen3X700	98800000			
N8100-594B	Express 5800/140 Hb(Pen3X700	12870000			
N8100-597B	Express 5800/140 Ra-4(Pen3X700	98800000			
N8100-598B	Express 5800/140 Ra-4(Pen3X700	12840000			
N8100-858A	Express 5800/120 Rc-2(Pen3 1.0	04320000			
N8100-861A	Express 5800/120 Ra-1	04600000			
合計			4	35040000	



システム構築ポイント

サーバベース型システムの選択ポイント

クライアント管理コスト

- ・クライアント台数が多い、将来も増加予定のシステム
- ・APを段階的に強化したいシステム(AP修正、追加等)
- ・クライアントスペック不足が顕著になってきたシステム

WANレスポンス問題

- ・クライアント・サーバ間の通信量が多いシステム
- ・モバイル環境からの利用を想定したシステム

データ集中化

- ・各地に分散したデータを集中化し、一元管理を行いたいシステム
- ・集中化にともなうAP修正は最小限にしたいシステム

セキュリティ問題

- ・社内データの漏洩、外部持ち出し等の機密漏洩に厳しいシステム

システム構築のポイント

対象APの検討

- ・マルチユーザを前提にしているか
- ・作業用一時ファイルは使用されていないか
- ・クライアント固有情報を保持していないか
- ・排他制御処理の範囲に注意が必要

サーバスペックの検討

- ・稼働APの使用メモリ、同時接続台数がサーバスペックを見積もるためのファクター
- ・CPU数は、ハイエンドのCPU1個より、ローエンドのCPU2個の方を推奨
- ・サーバ台数は、冗長性を考慮するために複数台を推奨(4CPU × 1台より2CPU × 2台)
ロードバランシングサービスによる負荷分散も利用可能

ネットワークの検討

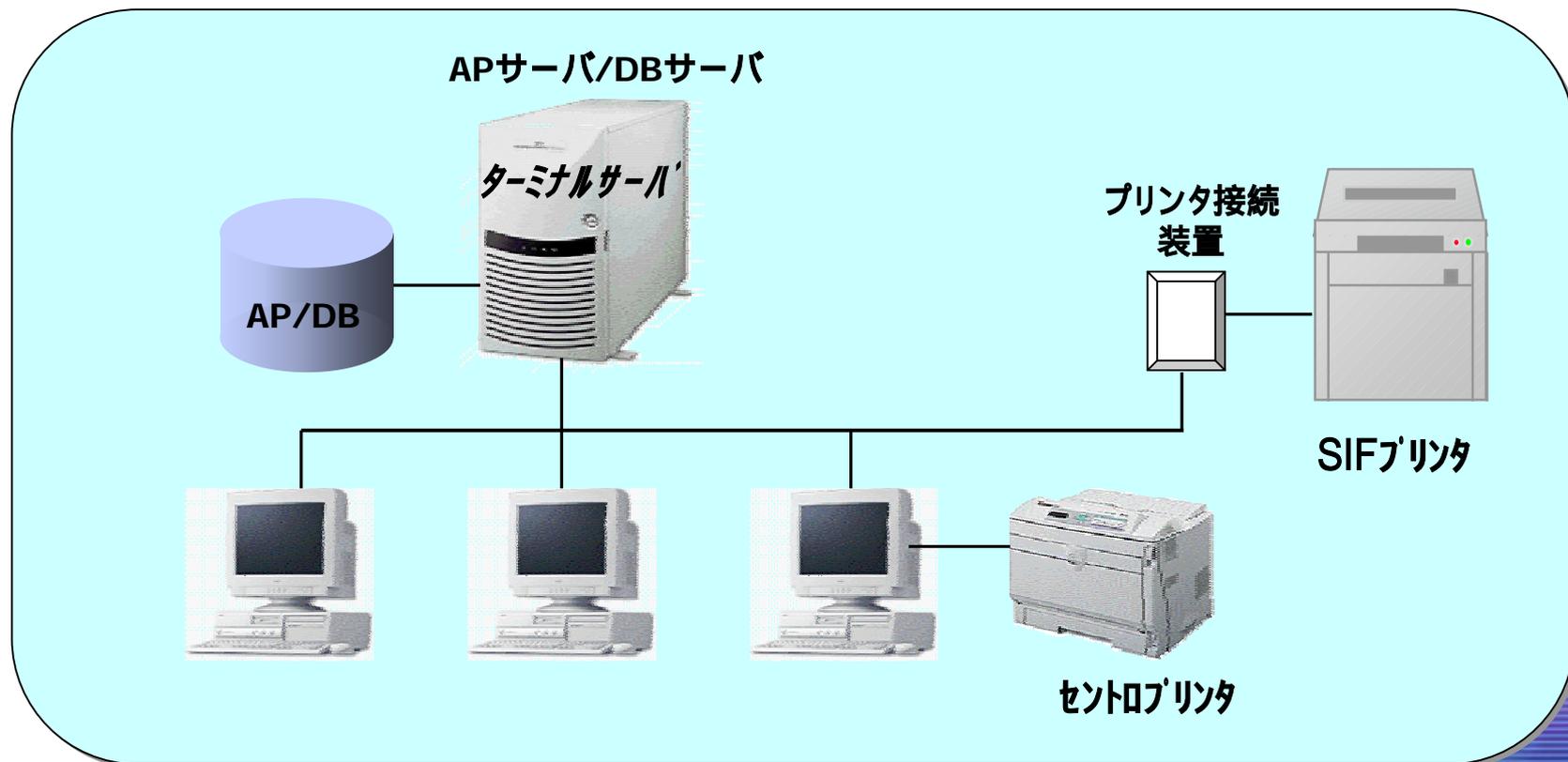
- ・目安として20kbps × セッション数
- ・必要であれば、帯域制御装置等の導入も必要

印刷処理の検討

- ・クライアントサーバ型でクライアント接続プリンタより大量印刷していたシステムをサーバ集中に移行する場合、APがサーバで動作するため、プリンタデータの流れ方が変わるので注意が必要

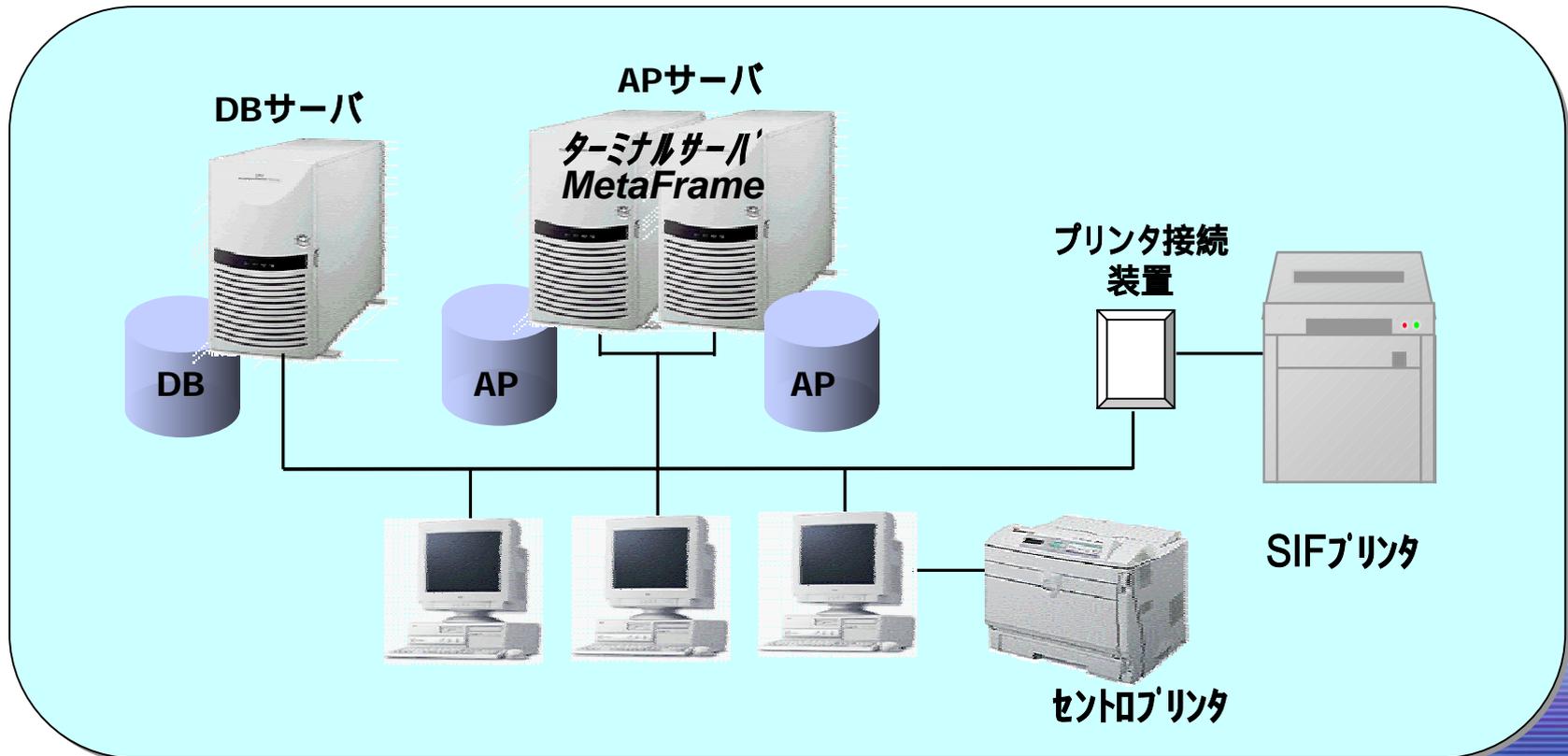
システム構成1:サーバ集中型

- 小規模システム向け
 - APサーバとDBサーバを一つにする形態



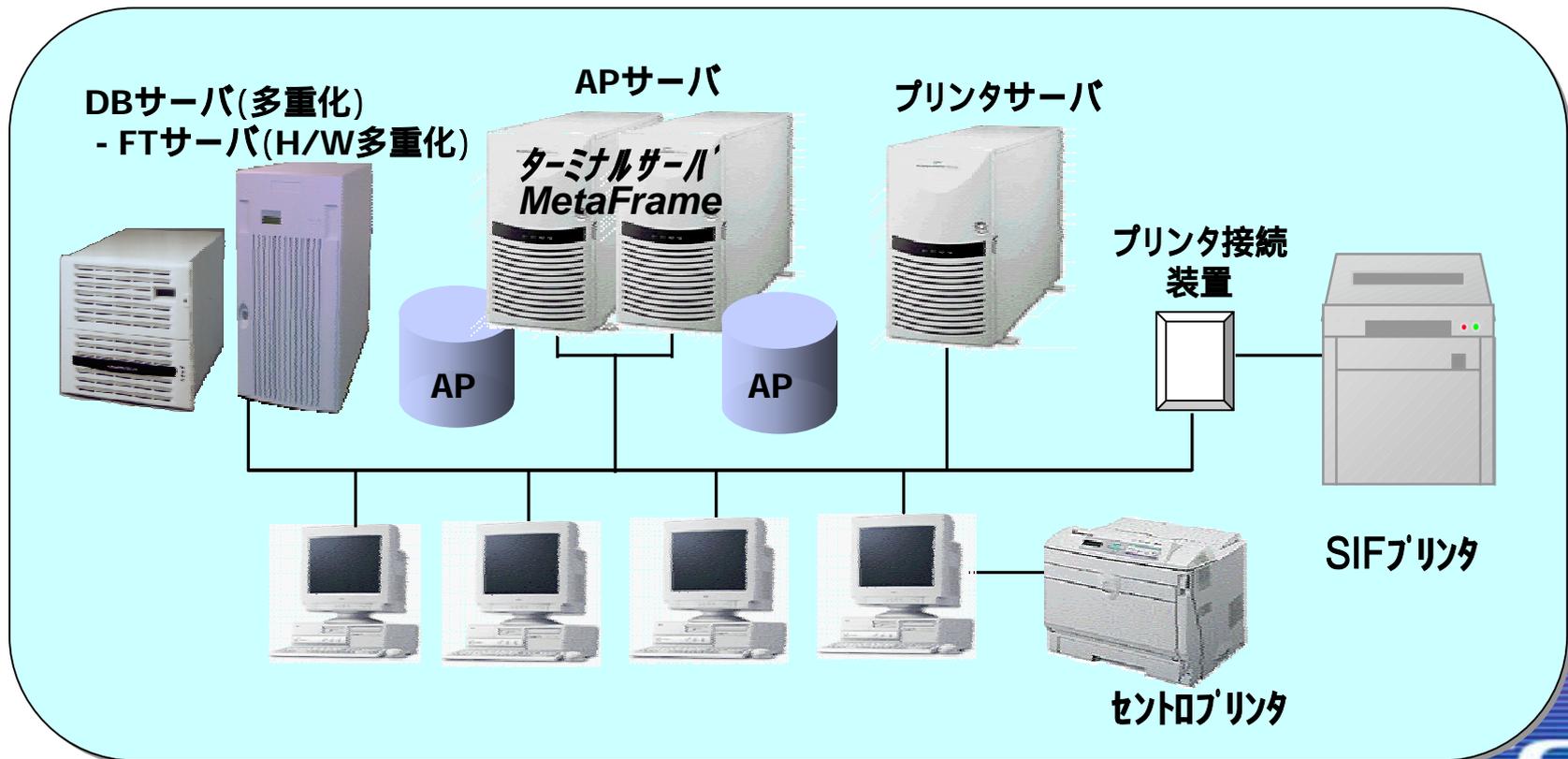
システム構成2:サーバ負荷分散型

- 中規模システム向け
 - APサーバ、DBサーバに分け、負荷分散、高速化



システム構成3:サーバ負荷分散型

- 大規模システム向け
 - 各用途別にサーバを分け、多重化
 - 負荷分散、高速化、連続可用性、高信頼性



導入事例

導入事例：A製造業殿生産管理システム

導入目的：クライアント管理のTCO削減，WAN回線でのシステムパフォーマンスの向上

運用規模：MetaFrameサーバ5台、クライアント300台、DBサーバ1台

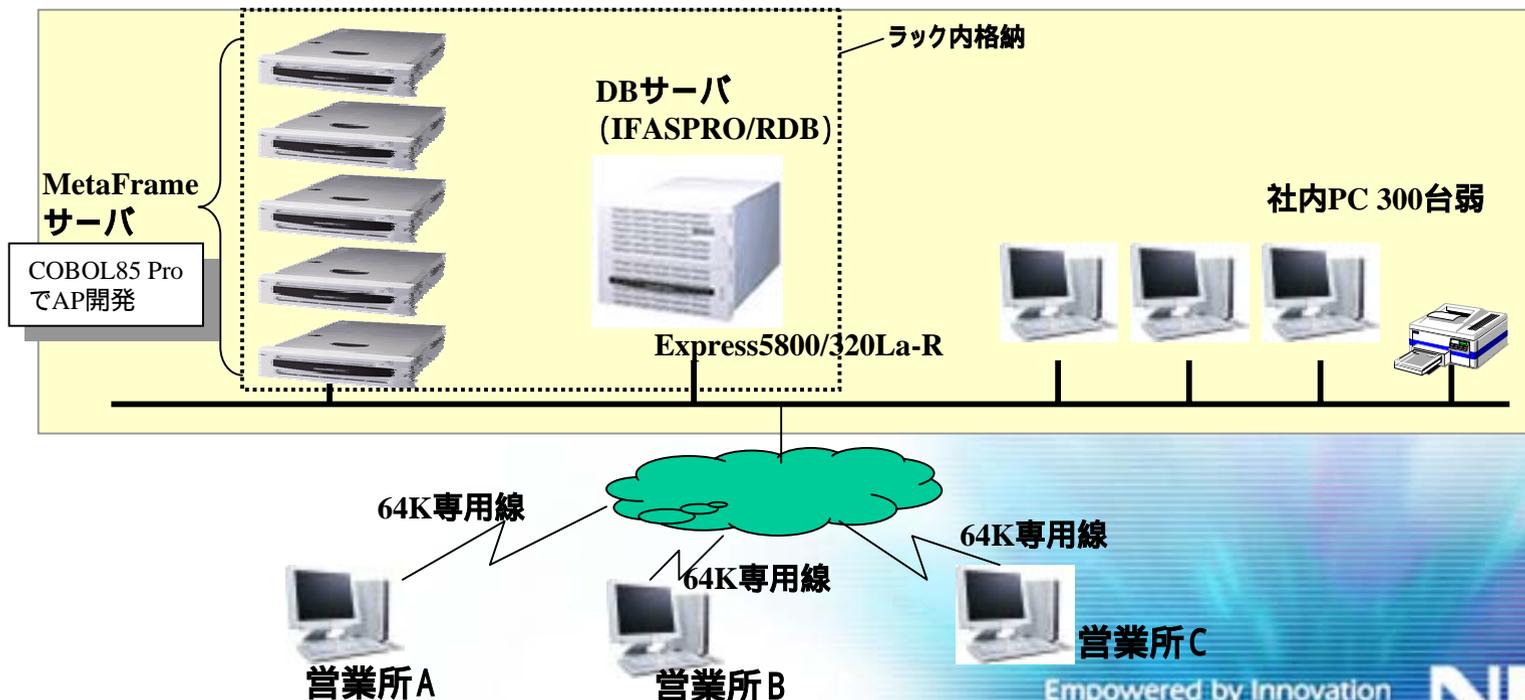
MetaFrameサーバ単位あたりのシステム

- ・サーバ：Express5800/Rc-2（2 CPU）、メモリ：4 GB、ディスク：9.1 GB × 3(RAID5)
- ・同時接続ユーザ数：200

使用アプリケーション：COBOL85 Proで開発

導入効果

- ・クライアントへのCOBOLプログラム展開が不要となりクライアントPCの管理コストを削減
- ・本社にデータ集中したにもかかわらずWAN接続先の営業所からLAN並みのパフォーマンスで業務を実現
- ・ロードバランス機能を使用し、最も軽負荷のアプリケーションサーバへ自動接続することで、サーバリソースを効率よく運用



導入事例：B金融業殿システム

導入目的： MetaFrameサーバを利用することにより、エンドユーザに対してASP形態でのサービスを提供

運用規模： MetaFrameサーバ1台、DBサーバ1台

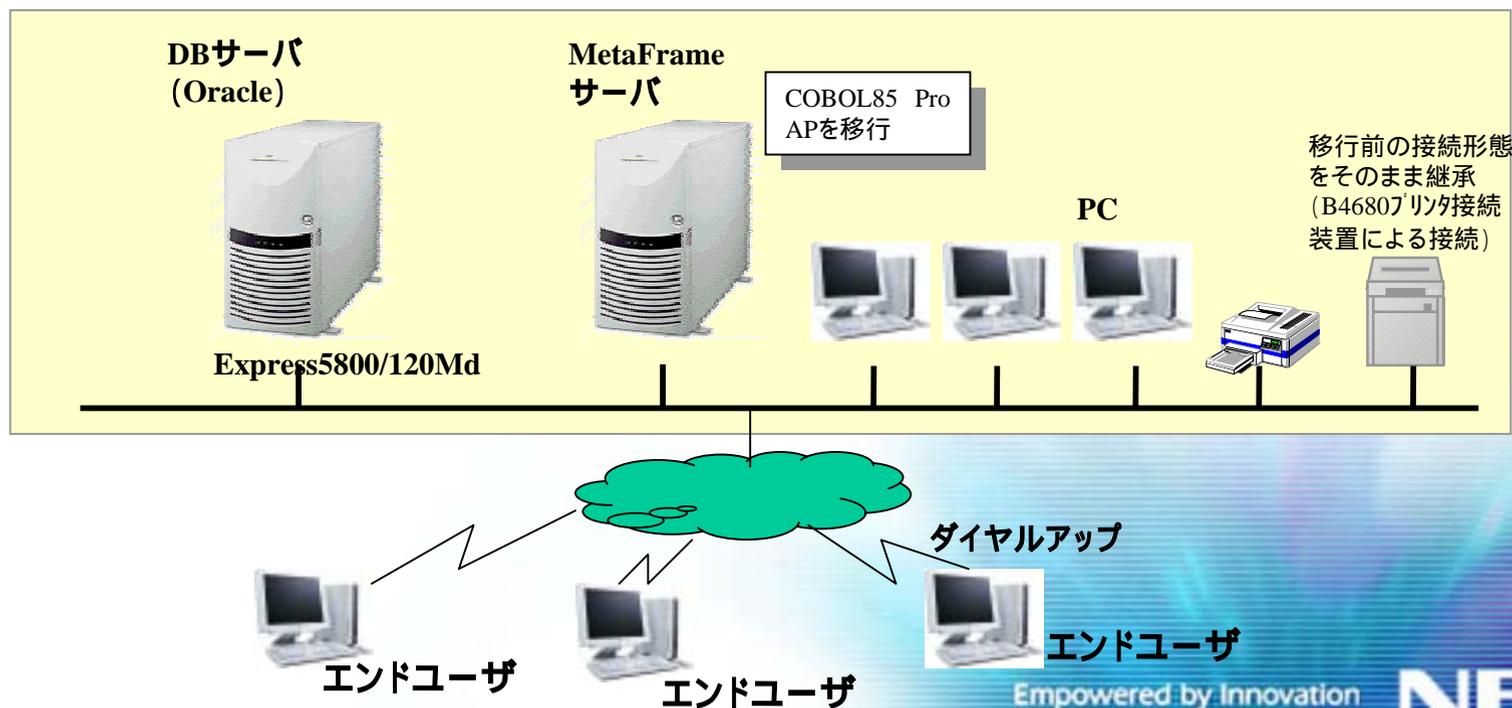
MetaFrameサーバ単位あたりのシステム

・サーバ: Express5800/120Md (2 CPU)、メモリ: 768MB、ディスク: 18.1 GB × 3(RAID5)

使用アプリケーション： COBOL85 Proで開発 (クライアント/サーバシステム(Windowsベース)からの移行)

導入効果

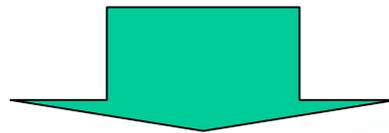
- ・クライアントへのCOBOLプログラム、クライアント製品の展開が不要となりクライアントPCの管理コストを削減
- ・COBOLソース、ファイル等は全てそのまま移行



ライセンスの考え方

NECの場合...

- COBOL製品(対応製品)は、APサーバヘインストール、クライアントは不要
- APサーバのCOBOL APを実行するクライアント毎にCOBOL実行ライセンスが必要
 - ex. APサーバ 2台、クライアント(PC) 10台



COBOL製品 × 2
実行ライセンス × 10

まとめ

まとめ

- **COBOLは既にサーバベースコンピューティングで使用されている言語**
- **サーバベースコンピューティングは、COBOL資産を早く・安く・正確にオープン化できるソリューション**
- **NECは、従来プラットフォーム、そして新たなプラットフォームのためのシステム構築に向けて、COBOL開発環境を提供し、今後も確かなサポートを続けていきます**

Empowered by Innovation

NEC